Applied Acoustics 170 (2020) 107511

Contents lists available at ScienceDirect

Applied Acoustics

journal homepage: www.elsevier.com/locate/apacoust

FLGCNN: A novel fully convolutional neural network for end-to-end monaural speech enhancement with utterance-based objective functions

Yuanyuan Zhu [a,b], Xu Xu [a,b], Zhongfu Ye [a,b,*]

[a] Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei, Anhui 230027, China
[b] National Engineering Laboratory for Speech and Language Information Processing, China, Hefei, Anhui 230027, People's Republic of China

ARTICLE INFO

Article history:
Received 14 November 2019
Received in revised form 30 May 2020
Accepted 16 June 2020
Available online 21 July 2020

Keywords:
End-to-end speech enhancement
gated convolution
CSTFT/CISTFT layer
Utterance-based objective functions

ABSTRACT

This paper proposes a novel fully convolutional neural network (FCN) called FLGCNN to address the end-to-end speech enhancement in time domain. The proposed FLGCNN is mainly built on encoder and decoder, while the extra convolutional-based short-time Fourier transform (CSTFT) layer and inverse STFT (CISTFT) layer are added to emulate the forward and inverse STFT operations. These layers aim to integrate the frequency-domain knowledge into the proposed model since the underlying phonetic information of speech is presented more clearly by time–frequency (T-F) representations. In addition, the encoder and decoder are constructed by the gated convolutional layers so that the proposed model can better control the information passed on in the hierarchy. Besides, motivated by the popular temporal convolutional neural network (TCNN), the temporal convolutional module (TCM) which is efficient in modeling the long-term dependencies of speech signal is inserted between encoder and decoder. We also optimize the proposed model with different utterance-based objective functions to exploit the impact of loss functions on performance, because the entire framework can realize the end-to-end speech enhancement. Experimental results have demonstrated that the proposed model consistently gives better performance improvement than the other competitive methods of speech enhancement.

1. Introduction

Speech enhancement algorithms have attracted a great deal of interest for a long time with various applications including hearing aids, speaker/speech recognition, hands-free communications, etc. [1]. Since the clean speech signals are usually vulnerable to disturbances from background noise and interference, the enhancement of speech is normally required to improve the perceived overall speech quality and/or intelligibility without much distortion of desired speech signals. Conventional monaural speech enhancement algorithms include statistical methods [2–4] and sparse-model-based approaches [5–8]. But these algorithms usually depend on some explicit assumptions and easily introduce the additional artifacts in the enhanced speech.

In the past few years, supervised methods based on deep neural networks (DNNs) have been the mainstream for speech enhancement and separation [9]. DNNs are powerful models that can learn complicated nonlinear mappings from large amounts of data, so DNNs generally tend to outperform the traditional algorithms when provided with enough data. The most popular deep learning methods for speech enhancement are the masking-based and the mapping-based methods. These two categories usually apply short-time Fourier transform (STFT) to convert noisy signal to time–frequency (T-F) representations and the training target is also constructed from T-F representations. Some of the most frequently employed training targets are ideal ratio mask (IRM), ideal binary mask (IBM) [10] and log-power spectra of target speech [11].

Even though using the T-F representations is the most popular approach, it still has some limitations. Fist of all, it is unclear whether the STFT is the optimal transformation of the signal for speech enhancement (even assume the parameters it depends on are optimal, such as size and overlap of audio frames, window type and so on). More importantly, inconsistent spectrogram or invalid STFT problem usually arises in these methods. A STFT X is consistent only if it satisfies

[*] Corresponding author at: Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei, Anhui 230027, China.

E-mail addresses: zhu0209@mail.ustc.edu.cn (Y. Zhu), xxu@ustc.edu.cn (X. Xu), yezf@ustc.edu.cn (Z. Ye).

https://doi.org/10.1016/j.apacoust.2020.107511
0003-682X/© 2020 Elsevier Ltd. All rights reserved.

$$X = \zeta[\zeta^{-1}(X)] \tag{1}$$

where $X = \zeta[x(t)]$ and $x(t)$ is a real-valued time-domain signal. $\zeta[\bullet]$ and $\zeta^{-1}[\bullet]$ represent forward and inverse STFT operators. But in the frequency-domain speech enhancement, popular approaches including T-F masking [10] and spectral mapping [11] generally focus on processing the STFT magnitudes while ignoring the phase information and just use the STFT phase of noisy signal for the time-domain signal reconstruction. Therefore, the mismatch between the enhanced magnitude and the noisy phase will most likely lead to an invalid STFT and inconsistent spectrogram [12]. Obviously, such invalid STFT problem causes undesired artifacts and unpleasant signal distortions in synthesized signals.

As an approach to overcome the above problems, a few recent studies have explored deep learning for time-domain speech enhancement. For example, the generative adversarial network (GAN) [13,14] and WaveNet [15] were subsequently applied to speech enhancement task. But most of these methods take the time frames of noisy utterance as input but fail to perform speech enhancement in an utterance-wise manner. To address this problem, some researchers apply fully convolutional neural network (FCN) to perform speech enhancement [16,17] because a FCN model only consists of convolutional layers [18] and the filters in convolution operation can accept inputs with variable lengths. However, the underlying characteristics of speech signal are more distinguishable from the background noise in T-F domain than that in time domain. Therefore, we believe that integrating the frequency domain knowledge into the time domain neural networks could be conductive to the core task of speech enhancement. [19] also proves that replacing the time domain loss with a frequency domain loss can improve the enhancement performance in time domain.

Motivated by the above considerations, we propose a novel and effective FCN called Fourier Layers-based Gated Convolutional Neural Network (FLGCNN) for end-to-end monaural speech enhancement. The proposed model framework can be regarded as an extension of the temporal convolutional neural network (TCNN) [17], which mainly consists of encoder, decoder and temporal convolutional module (TCM) [20]. The encoder creates a low dimensional representation of input noisy signal while the decoder aims to reconstruct the enhanced speech signal. Between the encoder and decoder, TCM is inserted to help network better learning the long-range dependencies from the past. But different from TCNN, the convolutional-based STFT (CSTFT) layer and inverse STFT (CISTFT) layer are added to the encoder and decoder respectively in order to emulate STFT and ISTFT in the neural network. As a result, the frequency-domain knowledge can be integrated into a time domain framework while avoiding the invalid STFT problem. In addition, we apply the gated linear units (GLUs) to construct the gated convolutional and deconvolutional layers for encoder and decoder. Similar to the long short-term memory (LSTM) model, GLUs play the role of controlling the information passed on in the hierarchy and this special gating mechanism allows to effectively capture long-range context dependencies by deepening layers without the gradient vanishing problem.

Based on this processing structure, we further utilize three different utterance-based objective functions defined by mean square error (MSE), scale-invariant source-to-distortion ratio (SI-SDR) and the short-time objective intelligibility (STOI) to optimize the model. As we all know, the most popular loss function for enhancement algorithms is MSE loss, but there exactly exists a mismatch between the model optimization and evaluation criterions for deep learning-based speech enhancement systems. The reason accounting for not applying the evaluation metrics as objective functions may be that accomplishing the metrics generally needs the whole clean/enhanced speech utterance. However, the conventional frequency-domain-based approaches usually process the magnitude spectrogram and a great deal of pre-processing an post-processing such as framing, STFT, overlap-add method are necessary. Other waveform enhancement algorithms including GAN and WaveNet still process the noisy waveforms in a frame-based manner. On the contrary, the end-to-end enhancement approaches via FCN make it possible to optimize the evaluation metrics directly, so we also train the proposed model with the other two metric-based objective functions to explore their effects on enhancement performance.

The rest of this paper is organized as follows. We introduce the monaural speech enhancement problem in Section 2. In Section 3, we describe the related modules and then the proposed model is presented with details. The utterance-based training targets are given in Section 4. Experiments and comparisons are provided in Section 5. Finally, Section 6 concludes this paper.

2. Problem formulation

Given a single-microphone noisy signal $y(t)$, the goal of monaural speech enhancement is to estimate target speech $s(t)$. In this study, we focus on the condition where clean speech is corrupted by an additive background noise. Hence, a noisy mixture can be modeled as

$$y(t) = s(t) + n(t) \tag{2}$$

where t indexes a time sample and $n(t)$ denotes the background noise. For T-F representations-based methods, speech enhancement can be formulated as a process that maps from acoustic features of a noisy mixture $y(t)$ to a T-F mask or a spectral representation of target speech $s(t)$. The estimated acoustic features by the model are resynthesized with noisy phase to reconstruct the time-domain speech signal. But for end-to-end approach, we aim to directly estimate $s(t)$ from $y(t)$.

3. System description

In this work, we propose a fully convolutional neural network that is comprised of a series of gated convolutional layers and TCM to enhance speech in time domain. We first briefly review the TCM architecture and gated mechanisms. Further, we introduce the designed CSTFT layer and CISTFT layer and show the details of the proposed model FLGCNN.

3.1. Temporal convolutional module (TCM)

The temporal convolutional network (TCN) was first proposed as a replacement for recurrent neural networks (RNNs) in sequence modeling task [20], which was integrated with causal and dilated convolutional layers as Fig. 1 and residual connections [22]. The causal convolutions ensure that there is no leakage of information

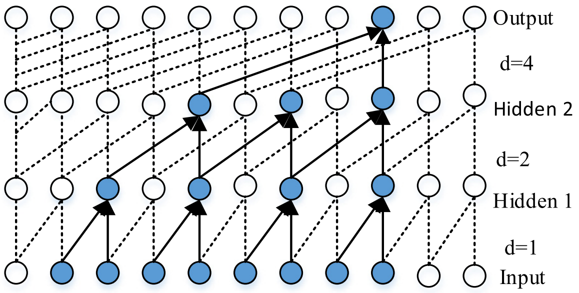


Fig. 1. An example of dilated causal convolution with a filter of size 2.

from the future to the past, while the dilated convolutions help to increase the receptive field.

Motivated by the TCN, [17] proposes the similar TCM that is comprised of stacked residual blocks as shown in Fig. 2. Each residual block consists of 3 convolutions: input 1×1 convolution, depthwise convolution, and output 1×1 convolution. The input convolution is used to double the number of incoming channels. The goal of output convolution is to get back to the original number of channels, which makes the addition of the inputs and outputs compatible. To further decrease the number of parameters, depthwise separable convolution involving depthwise convolution and pointwise 1×1 convolution is applied to replace the standard convolution. In addition, the input and the middle convolutions are followed by parametric ReLU [23] non-linearity and batch normalization [24].

3.2. Gated linear units

Gated mechanisms were first designed to facilitate the information flow over time in RNN [25], where an input gate and forget gate [26] were introduced to allow for long-term memory. Without these gates, it is easy that the information could vanish through the transformations of each timestep [27] and the vanishing or exploding gradient problem usually arises. Motivated by these, the output gates are considered in convolutional models to help control what information should be propagated through the hierarchy of layers. In [28], Oord et al. have shown the effectiveness of an LSTM-style mechanism for convolutional modeling of images:

$$\begin{aligned}\mathbf{y} &= \tanh(\mathbf{x} * \mathbf{W}_1 + \mathbf{b}_1) \odot \sigma(\mathbf{x} * \mathbf{W}_2 + \mathbf{b}_2) \\ &= \tanh(\mathbf{v}_1) \odot \sigma(\mathbf{v}_2)\end{aligned} \tag{3}$$

where $\mathbf{v}_1 = \mathbf{x} * \mathbf{W}_1 + \mathbf{b}_1$ and $\mathbf{v}_2 = \mathbf{x} * \mathbf{W}_2 + \mathbf{b}_2$. $*$ means a convolution operator. σ represents sigmoid function, and $\odot$ denotes element-wise multiplication. The gradient of gated tanh unit (GTU) as shown in Eq. (3) is

$$\begin{aligned}\triangledown[\tanh(\mathbf{v}_1) \odot \sigma(\mathbf{v}_2)] = \tanh'(\mathbf{v}_1) \triangledown \mathbf{v}_1 \odot \sigma(\mathbf{v}_2) \\ + \sigma'(\mathbf{v}_2) \triangledown \mathbf{v}_2 \odot \tanh(\mathbf{v}_1)\end{aligned} \tag{4}$$

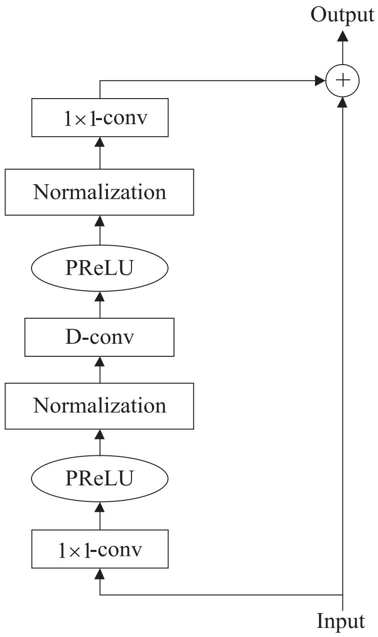


Fig. 2. The architecture of residual temporal convolution module.

where $\tanh'(\mathbf{v}_1), \sigma'(\mathbf{v}_2) \in (0, 1)$, and the prime symbol denotes differentiation. However, gradient gradually vanishes as we stack layers due to the downscaling factors $\tanh'(\mathbf{v}_1)$ and $\sigma(\mathbf{v}_2)$. To track this problem, a simplified gating mechanism named GLUs is proposed as

$$\begin{aligned}\mathbf{y} &= (\mathbf{x} * \mathbf{W}_1 + \mathbf{b}_1) \odot \sigma(\mathbf{x} * \mathbf{W}_2 + \mathbf{b}_2) \\ &= \mathbf{v}_1 \odot \sigma(\mathbf{v}_2)\end{aligned} \tag{5}$$

while the gradient of the GLUs includes a path $\triangledown \mathbf{v}_1 \odot \sigma(\mathbf{v}_2)$ without downscaling:

$$\triangledown [\mathbf{v}_1 \odot \sigma(\mathbf{v}_2)] = \triangledown \mathbf{v}_1 \odot \sigma(\mathbf{v}_2) + \sigma'(\mathbf{v}_2) \triangledown \mathbf{v}_2 \odot \mathbf{v}_1 \tag{6}$$

The architecture of gated convolutional layer constructed by GLU is shown in Fig. 3, which is mainly different from the plain convolutional layer in terms of output gates $\sigma(\mathbf{v}_2)$ to control the information passed on. The gated convolutional neural network (GCNN) by stacking gated convolutional layers has been proved to be efficient in building a hierarchical representation and capturing the long-range dependencies [21]. In this work, we adopt 2-dimensional gated convolutional layers (GConv2d) to build the main framework of encoder and decoder.

3.3. Convolutional-based STFT/ISTFT layer

As described in Section 1, we design the CSTFT/CISTFT layer to perform the pseudo-STFT and pseudo-ISTFT, so that the frequency information can be introduced into time-domain-based enhancement algorithms. Actually, the STFT and ISTFT operations are linear transforms by multiplying the framed signal with a complex-valued discrete Fourier transform (DFT) matrix **D** as following:

$$\mathbf{x}_f = \mathbf{D}\mathbf{x}_t \tag{7}$$

where $\mathbf{x}_f$ is the DFT of the framed signal $\mathbf{x}_t$. Since $\mathbf{x}_t$ is real-valued, the relation in Eq. (7) can be rewritten as

$$\mathbf{x}_f = (\mathbf{D}_r + j\mathbf{D}_i)\mathbf{x}_t = \mathbf{D}_r\mathbf{x}_t + j\mathbf{D}_i\mathbf{x}_t \tag{8}$$

where $\mathbf{D}_r$ and $\mathbf{D}_i$ are real-valued matrices formed by taking the real and imaginary part of **D** and j denotes the imaginary unit. Motivated by Eq. (8), the CSTFT layer is implemented by two 1-dimensional convolutions, each weights of which are initialized with real and imaginary part of STFT kernels respectively, and the CISTFT layer is similar to this one. These modules are constructed on normal convolutional layers and thus it is easy to integrate these modules into neural network. But it should be noted that these layers are not equivalent to the STFT/ISTFT operations. Actually, these layers are superior to STFT/ISTFT because the weights are learnable with back-propagation while the parameters of STFT/ISTFT are fixed.

Since the CSTFT/CISTFT layers are both constructed on convolutional layers, our model can be optimized with the task of mapping from the frequency domain to the time domain. The target is still the clean speech signal in time domain, so no invalid STFT problem arises.

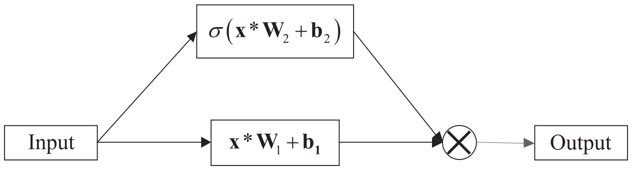

$\sigma(\mathbf{x}*\mathbf{W}_2+\mathbf{b}_2)$
Input
$\mathbf{x}*\mathbf{W}_1+\mathbf{b}_1$
Output

Fig. 3. The architecture of gated convolutional layer.

3.4. Proposed model

Following the aforementioned methodology and principle, we have designed an end-to-end speech enhancement model which consists of encoder, decoder, TCM and CSTFT/CISTFT layers as shown in Fig. 4. First of all, the input to CSTFT layer is the sequence of noisy signal and its dimension is $1 \times L$, where L is the number of sample points. The CSTFT layer outputs the pseudo-STFT representations with size of $2 \times T \times 257$ as the input of encoder, where T is the number of frames.

The first layer in the encoder increases the number of channels from 2 to 16, and the output dimension after the first layer is $16 \times T \times 257$. The next stacked six gated convolutional layers successively reduce the size along the frame dimension using convolutions and the final output of the encoder is of dimension $64 \times T \times 4$. None of the layers in the network modifies the size along the time dimension so that the output has the same number of frames as in the input. Each layer in the encoder is followed by batch normalization and parametric ReLU non-linearity.

The TCM operates on the reshaped output of encoder with size $256 \times T$ and produces an output of the same size. The TCM has three dilation blocks stacked together. A dilation block is formed by stacking six residual blocks having exponentially increasing dilation rates. In a dilation block, the successive dilation rates in the residual blocks are 1, 2, 4, 8, 16 and 32.

The decoder is a mirror-image of the encoder and consists of seven stacked gated transposed deconvolutional layers. Different from plain convolution layer, gated convolutional layer has two data flows. So unlike the skip connections in [17], our autoencoder has two kinds of skip connections: one is ordinary connection same as in [17], and the other is gated skip connection. At the training time, a dropout rate of 0.2 is applied at every 2 layers. Each layer in the decoder is followed by batch normalization and parametric ReLU non-linearity as well.

A more detailed description of the network parameters is given in Table 1. The hyparameters for encoder and decoder are specified in *filterHeight* × *filterWidth, (stride along frame, stride along frequency)*. For TCM, the hyparameters are in the format *filterHeight, dilationRate, outputChanels* and the entries enclosed by the small braces represent a residual block.

Compared to TCNN, the advantages of the proposed model are twofold. First of all, the frequency domain information is introduced by CSTFT/CISTFT layer to help neural network better enhance speech, while TCNN only utilizes the time-domain samples. As we all know, clearer structures of speech signal are usually presented by STFT spectrogram instead of sample points in time domain. And the energy of corresponding frequency components can be reflected by the values of frequency bins as well. Secondly, the stacked gated convolutional layers and two data flows in encoder and decoder can better control the information passed on in the hierarchy and learn the long-range context dependencies, but the encoder and decoder of TCNN are constructed by only plain convolution layers.

Table 1
The architecture of the proposed model.

Layer name	Input size	Layer hyparameters	Output size
CSTFT	$1 \times L$	$(512, 1, 257)$	$2 \times 257 \times T$
reshape_1	$2 \times 257 \times T$	–	$2 \times T \times 257$
GConv2d_1	$2 \times T \times 257$	$3 \times 5, (1, 1)$	$16 \times T \times 257$
GConv2d_2	$16 \times T \times 257$	$3 \times 5, (1, 2)$	$16 \times T \times 128$
GConv2d_3	$16 \times T \times 128$	$3 \times 5, (1, 2)$	$16 \times T \times 64$
GConv2d_4	$16 \times T \times 64$	$3 \times 5, (1, 2)$	$32 \times T \times 32$
GConv2d_5	$32 \times T \times 32$	$3 \times 5, (1, 2)$	$32 \times T \times 16$
GConv2d_6	$32 \times T \times 16$	$3 \times 5, (1, 2)$	$64 \times T \times 8$
GConv2d_7	$64 \times T \times 8$	$3 \times 5, (1, 2)$	$64 \times T \times 4$
reshape_2	$64 \times T \times 4$	–	$256 \times T$
TCM	$256 \times T$	$\left.\begin{matrix}\begin{pmatrix}1,1,512\\3,1,512\\1,1,256\end{pmatrix}\\\begin{pmatrix}1,1,512\\3,2,512\\1,1,256\end{pmatrix}\\\begin{pmatrix}1,1,512\\3,4,512\\1,1,256\end{pmatrix}\\\begin{pmatrix}1,1,512\\3,8,512\\1,1,256\end{pmatrix}\\\begin{pmatrix}1,1,512\\3,16,512\\1,1,256\end{pmatrix}\\\begin{pmatrix}1,1,512\\3,32,512\\1,1,256\end{pmatrix}\end{matrix}\right\}\times 3$	$256 \times T$
reshape_3	$256 \times T$	–	$64 \times T \times 4$
GDConv2d_7	$256 \times T \times 4$	$3 \times 5, (1, 2)$	$64 \times T \times 8$
GDConv2d_6	$256 \times T \times 8$	$3 \times 5, (1, 2)$	$32 \times T \times 16$
GDConv2d_5	$128 \times T \times 16$	$3 \times 5, (1, 2)$	$32 \times T \times 32$
GDConv2d_4	$128 \times T \times 32$	$3 \times 5, (1, 2)$	$16 \times T \times 64$
GDConv2d_3	$64 \times T \times 64$	$3 \times 5, (1, 2)$	$16 \times T \times 128$
GDConv2d_2	$64 \times T \times 128$	$3 \times 5, (1, 2)$	$16 \times T \times 257$
GDConv2d_1	$64 \times T \times 257$	$3 \times 5, (1, 2)$	$2 \times T \times 257$
CISTFT	$2 \times T \times 257$	(512, 1, 257)	$1 \times L$

4. Utterance-based training targets

One benefit of end-to-end algorithms is the ability to optimize the model in an utterance-wise manner. As a consequence, the objective functions can be designed for whole utterances while loss functions in many STFT-based methods are not directly applicable. In this section, we introduce three utterance-based training targets. One is the MSE loss, and the others are the metric-based loss functions.

4.1. MSE loss

The common goal of model-based speech enhancement methods can be regarded as minimizing the following objective function:

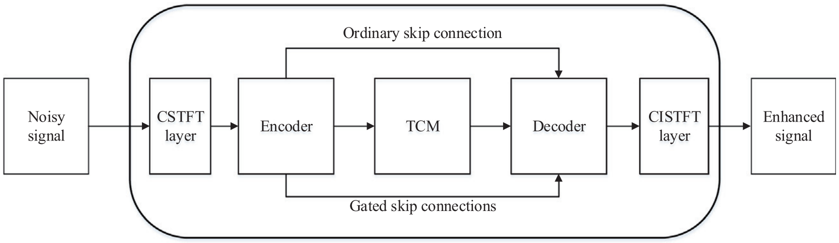


Fig. 4. The block diagram of the proposed framework of the proposed end-to-end model for speech enhancement.

$$O = \|\hat{\mathbf{s}} - \mathbf{s}\|^{\beta} \tag{9}$$

where β is a tunable parameter to scale the distance. One of such loss is defined as the average of the MSE losses between the clean and enhanced speech:

$$O = \frac{1}{L}\|\hat{\mathbf{s}} - \mathbf{s}\|^{2} \tag{10}$$

where $\hat{\mathbf{s}} \in \mathbb{R}^{1\times L}$ and $\mathbf{s} \in \mathbb{R}^{1\times L}$ are the estimated and original clean sources, respectively.

However, on fact is that MSE-based objective functions just simply compare the similarity between the clean and enhanced speech. In fact, there is an inconsistency between MSE and evaluation criterion for enhanced speech. For example, the relationship between the MSE value and human auditory perception is still not a monotonic function. The MSE between the original signal, its negative version, and its amplitude shifted version is very large, but these three signals are almost same for human by listening. Therefore, it is necessary to find loss functions that have stronger relationship with the performance evaluation of speech signal.

4.2. SI-SDR loss

SI-SDR is one of the objective functions developed from SDR metric which demonstrates the overall distortion of enhanced speech [29]. The higher SDR value indicates that the enhanced speech has less distorted components. SI-SDR is defined as:

$$\begin{cases} \mathbf{s}_{target} := \frac{\langle \hat{\mathbf{s}}, \mathbf{s}\rangle \mathbf{s}}{\|\mathbf{s}\|^{2}} \\ \mathbf{e}_{noise} := \hat{\mathbf{s}} - \mathbf{s}_{target} \\ \mathrm{SI-SDR} := 10\log_{10}\frac{\|\mathbf{s}_{target}\|^{2}}{\|\mathbf{e}_{noise}\|^{2}} \end{cases} \tag{11}$$

where $\|\mathbf{s}\|^{2} = \langle \mathbf{s}, \mathbf{s}\rangle$ denotes the signal power. Scale invariance is ensured by normalizing $\hat{\mathbf{s}}$ and $\mathbf{s}$ to zero-mean prior to the calculation.

4.3. Perception-based loss

The perceptual evaluation of speech quality (PESQ) and STOI scores are both prevalent measures used for predicting the intelligibility of noisy or processed speech [32,33]. A higher STOI value and PESQ value indicate better speech intelligibility and quality. But compared to PESQ, the most functions in STOI computations are continuous and STOI is more correlated with the improvement in word error rate than PESQ [30,31]. So in this paper, we focus on optimizing the STOI metric and briefly describe the 4 major steps in calculating STOI:

1) *T-F representations:* STFT is applied to both noisy and clean signals to obtain the corresponding T-F representations which are simplified internal representations resembling the transform properties of the auditory system. The signals are first segmented into 50% overlapping, Hanning-windowed frames with a length of 256 samples and then Fourier transformation is performed. Noted that the silent regions have been removed by excluding the frames where the clean speech energy is lower than 40 dB with respect to this maximum clean speech energy frame.

2) *One-third octave band analysis:* This is approximated by summing STFT coefficient energies. Let $S(k,m)$ denoted the kth T-F bin of the mth frame of the clean speech, then we get

$$S_j(m) = \sqrt{\sum_{k\in CB_j} |S(k,m)|^{2}}, \quad j = 1,2,\ldots,J \tag{12}$$

where j is the one-third octave index, and J is usually set 15. CB_j represents the index set of STFT coefficients related to the jth one-third octave frequency band. The lowest center frequency is set to 150 Hz and the highest center frequency is 4.3 kHz or so. By grouping the frequency bins, the short-time spectrogram vector can be obtained by

$$\mathbf{s}_{j,m} = [S_j(m-N+1), S_j(m-N+2), \ldots, S_j(m)]^{T} \tag{13}$$

Typically, the parameter N is set 30 which equals an analysis length of 384 ms. Similarly, $\hat{\mathbf{s}}_{j,m}$ denotes the short-time spectrogram vector of the enhanced speech.

3) *Normalization and clipping:* First, a normalization procedure is applied to $\hat{\mathbf{s}}_{j,m}$ in order to compensate for global level difference. Second, the normalized $\hat{\mathbf{s}}_{j,m}$ is clipped to lower bound the SDR. Let the $\check{\mathbf{s}}_{j,m}$ represent the normalized and clipped short-time spectrogram vector.

4) *Intelligibility measure:* The intermediate intelligibility index is defined as the spectral correlation coefficients between the two temporal envelopes:

$$d_{j,m} = \frac{(\mathbf{s}_{j,m} - \mu_{\mathbf{s}_{j,m}})^{T}(\check{\mathbf{s}}_{j,m} - \mu_{\check{\mathbf{s}}_{j,m}})}{\left\|\mathbf{s}_{j,m} - \mu_{\mathbf{s}_{j,m}}\right\|_2 \left\|\check{\mathbf{s}}_{j,m} - \mu_{\check{\mathbf{s}}_{j,m}}\right\|_2} \tag{14}$$

where $\|\cdot\|_2$ denoted the L_2 norm and $\mu_{(\cdot)}$ represents the sample mean of the corresponding vector. Finally, the eventual STOI value is simply given by the average of the intermediate intelligibility index over all frames and bands:

$$\mathrm{STOI} = \frac{1}{JM}\sum_{j,m} d_{j,m} \tag{15}$$

Although the calculation of STOI is somewhat complicated, most of the computation is differentiable and thus it can be employed as the objective function for our utterance optimization as represented by the following equation:

$$O = -\frac{1}{B}\sum_{b} stoi(\hat{\mathbf{s}}_b, \mathbf{s}_b) \tag{16}$$

where $\hat{\mathbf{s}}_b$ and $\mathbf{s}_b$ are the bth estimated and clean utterance in a batch respectively, and B is the batch size of training utterances. $stoi(\cdot)$ is the function that calculates the STOI value of the noisy/enhanced utterance given the clean one. In this work, we propose an objective function that incorporates both the SI-SDR and STOI, which can be represented by

$$O = -\frac{1}{B}\sum_{b}\left(\alpha \cdot 10\log_{10}\frac{\|\mathbf{s}_{target}\|^{2}}{\|\mathbf{e}_{noise}\|^{2}} - stoi(\hat{\mathbf{s}}_b, \mathbf{s}_b)\right) \tag{17}$$

where α is the weighting factor which is simply set to a small number to balance the scale of two targets. Since the contribution of α·SI-SDR value to the final loss function is much smaller than STOI value, we think training model with Eq. (17) still mainly aims to improve the STOI scores so we dub this loss function as *S-STOI*.

5. Experiment

In this section, some experimental results and discussions are presented. First, the data and performance metrics used in the experiments are considered. Second, we provide the comparison results for different enhanced models and objective functions. Additionally, the spectrograms and waveforms are presented for further analysis. Finally the listening test results are reported.

5.1. Data preparation and evaluation metric

In this work, all clean speech and noise data we use are selected from Wall Street Journal (WSJ0) corpus [34] and Musan dataset [35]. A training set consisting of 40 h of noisy signals and 3.3 h of validation data are generated by randomly selecting 7000 utterances and 2000 utterances from the WSJ0 tranining set si_tr_s. Noted that the training speakers are different from the validation speakers. Each training utterance is corrupted with 7 types of noise randomly selected from 800 types of noises in Musan at three levels of signal-to-noise ratio (SNR), i.e., −6 dB, −3 dB, 0 dB, 3 dB. Similarly, each validation utterance is mixed with 2 types of noise randomly selected from 90 noises in Musan. Another 1000 randomly selected utterance from the speakers in si_dt_05 and si_et_05 are used to construct the test set with one type from the other 39 noises in Musan dataset at three levels SNR. All the clean speech and noise waveforms are down-sampled to 16 kHz.

In our experiments, models are compared using STOI, PESQ and SDR scores, all of which represent the standard metrics for speech enhancement [29,32,33]. STOI has typical value range from 0 to 1, while PESQ values range from −0.5 to 4.5.

5.2. Training details

The initial learning rate is 1e^{-3} or 1e^{-4}, depending on the model configuration, and is halved if the accuracy of validation set is not improved in 3 consecutive epochs. Adam is used as the optimizer. A maximum of 60 training epochs is run to avoid over-adaptation. We use an early stopping criterion by stopping training if there is no loss improvement on the validation set for 10 epochs.

5.3. Experimental results

1) Comparison Between Different Network Architectures: To illustrate the efficiency of the proposed model, experiments compared our proposed model (FLGCNN) with original TCNN [17], TCNN with CSTFT layer and CISTFT layer (FLTCNN). These three time-domain models were compared for SI-SDR loss training. The corresponding results were shown in Table 2 (For convenience, we have left out the SDR unit *dB* in Table 2).

In addition, we also trained the TCNN with the frequency domain loss proposed in [19] (TCNN-MAE), where the loss was defined as the mean absolute error (MAE) between the estimated STFT magnitude and the clean STFT magnitude using L_2 norm. However, this approach is unable to work well in an end-to-end enhancement framework based on the results in Table 2, especially in relative low SNR level like −3 dB.

From Table 2, we can observe that there is a substantial performance gap between TCNN and FLTCNN in terms of all three metrics. As presented in [17], TCNN just directly process the sample points. But the characteristics of speech signal represented in the T-F domain are more distinguishable than in the time domain. The better results achieved by FLTCNN suggest that the frequency domain information introduced by CSTFT layers is really conducive to extracting the speech components from noisy signal. Next, we find that the results of FLGCNN are generally better than results of FLTCNN over three measurements. Since the key difference between FLGCNN and FLTCNN is the gated convolutional layers in encoder and decoder, this implies that gating mechanisms are really helpful for speech enhancement. Compared with the other networks, the proposed FLGCNN yields the most significant improvements over the test dataset in terms of SDR, STOI and PESQ, suggesting that the FLGCNN architecture is better than FLTCNN and TCNN for speech enhancement in time domain. In the −3 dB SNR case, for example, the FLGCNN improves the SDR score by 16.610, STOI score by 0.218 and PESQ score by 1.059 as compared to the unprocessed mixtures for the test set.

2) Comparison Between Different Objective Functions: In this part, we aim to conduct the experiments with the proposed model based on different objective functions including MSE loss, SI-SDR loss and STOI loss. But during training with STOI loss, we found that some unknown disturbing components arose in the enhanced speech and noise was not well suppressed. We analyzed that there were three reasons accounting for this phenomenon. First and foremost, the non-speech regions are not taken into consideration in the STOI score calculation. Secondly, the highest center frequency is about 4.3 kHz so optimizing STOI loss makes the model ignore the higher frequency region. Additionally, since the STOI is defined by the correlation coefficients between the clean and enhanced speech, the solution for maximizing the STOI value is not unique. So in this experiment, we used S-STOI loss function instead. The results shown in Table 3 were consistent with our analysis as well (For convenience, we left out the SDR unit *dB* in Table 3).

From Table 3, we observe that the proposed FLGCNN has higher SDR score with lower PESQ and STOI scores when trained with SI-SDR objective function. But when changing the objective function from SI-SDR to S-STOI, the STOI and PESQ values of the enhanced speech could be considerably improved with slightly decreased SDR score. As we all know, improving speech intelligibility is generally more challenging than enhancing quality at lower SNR condition, i.e. −6 dB, −3 dB SNR, but the results of the proposed model optimized with S-STOI are still encouraging: the PESQ score is 1.18 higher than noisy signal, the STOI score is 0.244 higher than noisy signal. Generally, the enhanced speeches with MSE loss function obtain the lowest measure scores in all SNR conditions, which demonstrates that metric-based loss functions have more significant impact on improving the measure scores than only minimizing MSE between clean and enhanced speech.

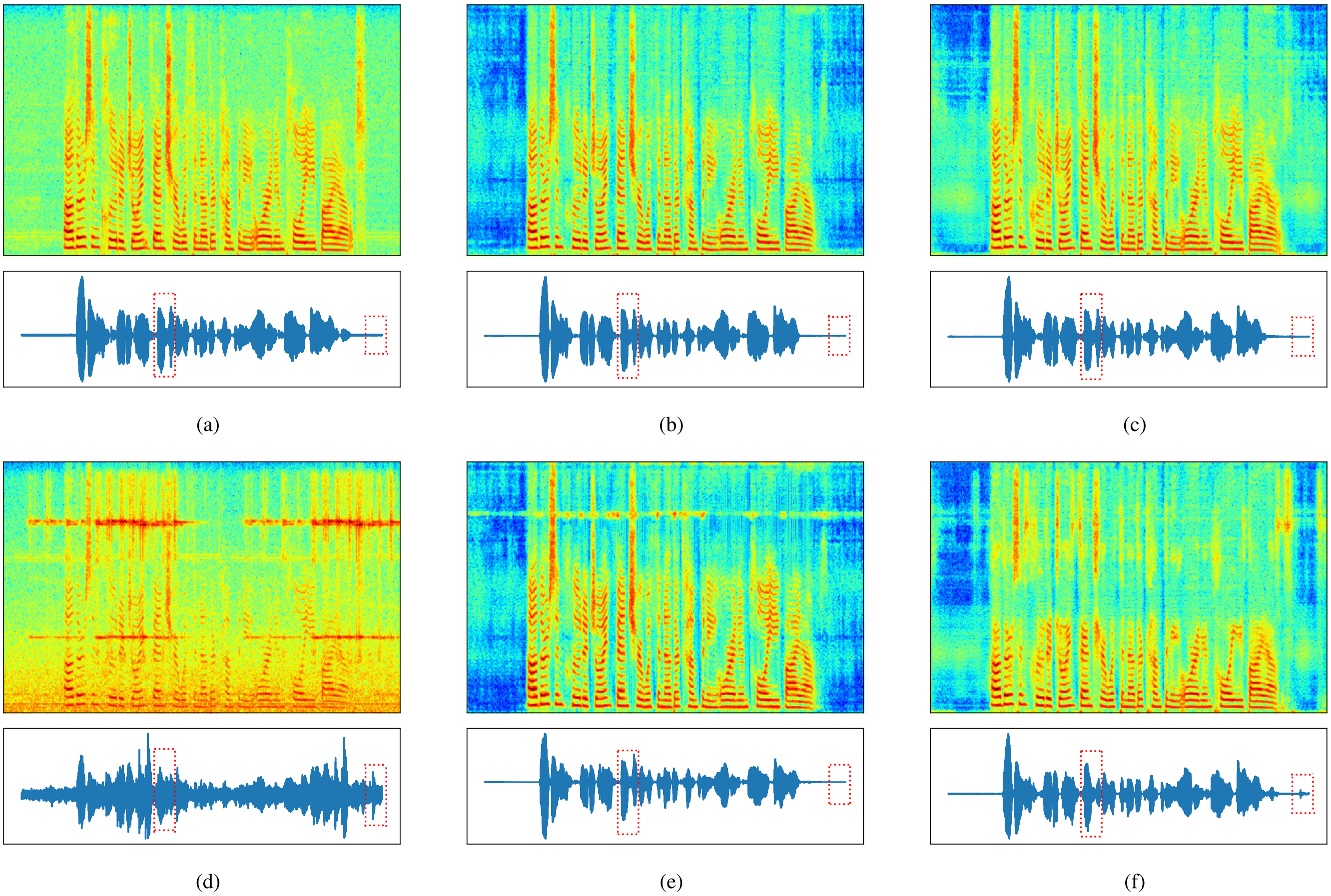
3) Spectrograms and Waveforms Comparison: For better illustration, we have plotted spectrograms and waveforms of a clean WSJ0 utterance and the same utterance corrupted by bell noise at −3 dB SNR as shown in Fig. 4 and Fig. 5(d). The spectrogram of enhanced speech by FLGCNN trained with SI-SDR loss has been presented in Fig. 5(b). Furthermore, in order to demonstrate the quasi STFT and quasi ISTFT via CSTFT and CISTFT layers are different from the normal STFT and ISTFT, we extracted the output of decoder in FLGCNN and apply the normal ISTFT to get the enhanced speech in time domain. The corresponding spectrogram are shown in Fig. 5(e). In other words, we processed the same decoder output by using

Table 2
Performance comparisons with various competitive network models.

SNR	−6 dB			−3 dB			0 dB			3 dB		
Metric	STOI	PESQ	SDR	STOI	PESQ	SDR	STOI	PESQ	SDR	STOI	PESQ	SDR
Noisy	0.537	1.772	−5.874	0.575	1.958	−2.922	0.635	2.143	0.052	0.709	2.331	3.039
TCNN−MAE	0.610	2.275	6.193	0.666	2.458	7.747	0.768	2.616	8.114	0.775	2.679	12.904
TCNN	0.719	2.687	11.218	0.766	2.850	12.669	0.798	2.985	14.163	0.846	3.157	15.770
FLTCNN	0.745	2.760	11.299	0.782	2.930	13.025	0.821	3.112	14.564	0.856	3.221	16.141
FLGCNN	**0.755**	**2.832**	**12.104**	**0.793**	**3.017**	**13.688**	**0.827**	**3.165**	**15.193**	**0.862**	**3.272**	**16.711**

Table 3
Performance comparisons with various loss functions.

SNR	−6 dB			−3 dB			0 dB			3 dB		
Metric	STOI	PESQ	SDR	STOI	PESQ	SDR	STOI	PESQ	SDR	STOI	PESQ	SDR
Noisy	0.537	1.772	−5.874	0.575	1.958	−2.922	0.635	2.143	0.052	0.709	2.331	3.039
MSE	0.741	2.808	10.133	0.775	2.946	12.102	0.820	3.108	13.970	0.842	3.180	15.434
SI-SDR	0.755	2.832	**12.104**	0.793	3.017	**13.688**	0.827	3.165	**15.193**	0.862	3.272	**16.711**
S-STOI	**0.787**	**2.985**	11.777	**0.819**	**3.138**	13.430	**0.850**	**3.299**	15.173	**0.886**	**3.382**	16.558


(a)

(b)

(c)

(d)

(e)

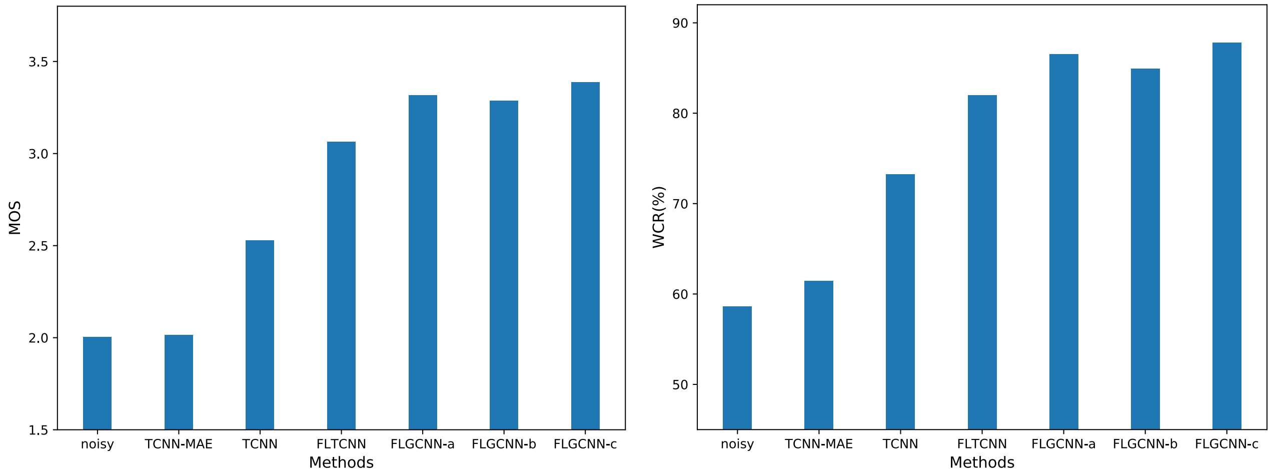
(f)

Fig. 5. Spectrograms and waveforms of a WSJ0 utterance: (a) clean speech, (d) noisy speech (STOI = 0.483, PESQ = 1.756, SDR=-2.991), (b) enhanced speech by FLGCNN trained with SI-SDR loss (STOI = 0.806, PESQ = 2.926, SDR = 13.768), (e) enhanced speech by using the decoder output of FLGCNN and the original ISTFT (STOI = 0.774, PESQ = 2.716, SDR = 7.805), (c) enhanced speech by FLGCNN trained with S-STOI loss (STOI = 0.820, PESQ = 3.068, SDR = 13.588), (f) enhanced speech by FLGCNN trained with only STOI loss (STOI = 0.737, PESQ = 2.921, SDR = 11.191).

two different methods: one was the quasi ISTFT via CISTFT layer and the other was original ISTFT. Comparing the Fig. 5(e) and Fig. 5(e), we observe that the CISTFT layer not only plays the role of ISTFT, but also removes the background noise effectively, which verifies that the weights of CISTFT layer are superior to the normal ISTFT kernels based on learning and backpropagation.

For perception-optimized speech, we utilized the original STOI and S-STOI as objective function to train our proposed model respectively. From the Fig. 5(e), and Fig. 5(f), we can easily find that the speech pattern in high frequency components is not identifiable compared to that in low and mid frequency components in Fig. 5(f)). This is because the highest center frequency of one-third octave band is about 4.3 kHz in STOI calculation, which makes the trained model pay no attention to the high frequency region and just moved the most of components. In addition, since the solution is not unique for correlation coefficients in STOI calculation, the obtained enhanced speech is usually not the one we want. In a consequence, the three metric scores for enhanced speech by FLGCNN trained with S-STOI are higher than that obtained by FLGCNN trained with only STOI loss function.

Compared to the Fig. 5(b), we can observe that much more residual noise exists in the silent regions in Fig. 5(c). The reason is that regions where show no speech activity has been removed before calculating STOI value. So during training, the model just ignores these regions where the noise actually exists. As a result, the enhanced speech shown in Fig. 5(c) has the lower SDR score than that shown in Fig. 5(b). The corresponding waveforms were plotted along with spectrograms in Fig. 5. Although the characteristics of signals are less distinguishable in waveforms, we can still find the differences among these waveforms. As marked by the first red block, the proposed technique (Fig. 5(b) and (c)) reconstructs better speech structures than others methods (Fig. 5(e)) and (f)). As marked by the second red blocks, there are still residual noise existing at the end of waveforms in Fig. 5(f). Therefore, these differences of waveforms in the temporal domain verify the advantages of the proposed FLGCNN as well.



(a) MOS

(b) WCR

Fig. 6. Average MOS and WCR scores of human subjects ("FLGCNN-a" represents FLGCNN(SI-SDR), "FLGCNN-b" represents FLGCNN(MSE), "FLGCNN-c" represents FLGCNN(S-STOI))).

4) Subjective Evaluation: In this part, subjective tests are conducted to evaluate the performance of the proposed FLGCNN for speech enhancement task. Fifteen normal hearing subjects (nine males and six females) participated in the listening tests. The experiments were conducted in a quiet environment and the stimuli were played to the subjects at a comfortable listening level. Each subject participated in a total 8 test conditions, and each condition contained 7 sentences: 1 SNR level × 1 noise type × 7 enhancement methods, i.e., noisy, TCNN-MAE, TCNN, FLTCNN, FLGCNN(SISNR), FLGCNN(MSE), FLGCNN(S-STOI). Since the human auditory system is robust to noise under 0 dB and 3 dB, we only selected lower SNRs (-6 dB and -3 dB). The order of these conditions was randomly selected for every subject. Listeners were allowed to repeat the stimuli twice and respond using letter and digit keys on computer keyboard. For evaluating speech intelligibility, the word correct rate (WCR) was calculated based on the ratio of the number of correctly identified words and the total number of words under each test condition. For evaluating speech quality, the subjects were asked to rate the quality of speech in a five-point Likert scale score (1:Bad, 2: Poor, 3: Fair, 4: Good, 5: Excellent).

Fig. 6 illustrates the average MOS scores and WCR results of listening test. We can observe that the quality and intelligibility of enhanced speech are improved greatly compared to the noisy one by the proposed FLGCNN. Furthermore, the FLGCNN optimized by S-STOI and SI-SDR obtains the higher MOS and WCR scores, which implies that these two measurements are indeed helpful for speech quality and intelligibility.

6. Conclusions

In this paper, we have proposed a novel model based on FCN architecture for end-to-end monaural speech enhancement. The proposed framework can be regarded as an improved version of TCNN but solve several problems simultaneously. 1) The frequency domain information can be integrated into an end-to-end model via the designed CSTFT and CISTFT layers, which helps model to better explore the speech characteristic from noisy signal. 2) The encoder and decoder were designed on gated convolutions in order to better control the information passed on in the hierarchy. Besides, the proposed encoder-decoder has one more gated skip connection compared to normal autoencoder. 3) The mismatch between objective function for optimization and the evaluation measure can be solved by the utterance-based speech enhancement model. Therefore, we trained the proposed model with utterance-based loss and made comparisons for three different objective functions. The experimental results shown in Table 2 have verified the effectiveness of CSTFT/CISTFT layers and gated convolutions, while the results in Table 3 have proved that measure-based loss functions could greatly improve the metric scores. A further research direction is to explore ways to apply the proposed model to real-time tasks.

CRediT authorship contribution statement

Yuanyuan Zhu: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing - original draft, Writing - review & editing. **Xu Xu:** Writing - original draft, Writing - review & editing, Formal analysis. **Zhongfu Ye:** Writing - original draft, Supervision, Project administration, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (No. 61671418).

References

[1] Loizou PC. Speech enhancement: theory and practice. Boca Raton, FL, USA: CRC; 2013.
[2] Boll SF. Suppression of acoustic noise in speech using spectral subtraction. IEEE Trans Acoust Speech Signal Process 1979;27(2):113–20.
[3] Lim JS, Oppenhenlim AV. "Enhancement and bandwidth compression of noisy speech. Proc IEEE 2005;67(12):1586–604.
[4] Ephraim Y, Malah D. Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator. IEEE Trans Acoust Speech Signal Process 1984;32(6):1109–21.
[5] Sigg CD, Dikk T, Buhmann JM. Speech enhancement using generative dictionary learning. IEEE Trans Audio Speech Lang Process 2012;20(6):1698–712.

[6] Luo Y, Bao G, Ye Z. Supervised monaural speech enhancement using complementary joint sparse representations. IEEE Signal Process Lett Feb. 2016;23(2):237–41.

[7] Fu J, Zhang L, Ye Z. Supervised monaural speech enhancement using two-level complementary joint sparse representations. Appl Acoust Mar. 2018;132:1–7.

[8] Kwon K, Shin JW, Kim NS. NMF-based speech enhancement using bases update. IEEE Signal Process Lett Apr. 2015;22(4):450–4.

[9] Wang D, Chen J. Supervised speech separation based on deep learning: an overview. IEEE/ACM Trans Audio Speech Lang Process 2018;26:1702–26.

[10] Wang Y, Narayanan A, Wang D. On training targets for supervised speech separation. IEEE/ACM Trans Audio Speech Lang Process 2014;22(12):1849–58.

[11] Xu Y, Du J, Dai L, Lee C. An experimental study on speech enhancement based on deep neural networks. IEEE Signal Process Lett Jan. 2014;21(1):65–8.

[12] Gerkmann T, Krawczyk-Becker M, Roux JL. Phase processing for single-channel speech enhancement: history and recent advances. IEEE Signal Process Mag Mar. 2015;32(2):55–66.

[13] Pascual S, Bonafonte A, Serra J. Segan: speech enhancement generative adversarial network, arXiv preprint arXiv: 1703. 09452, 2017..

[14] Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial networks, arXiv preprint arXiv:1406.2661, 2014..

[15] Rethage D, Pons J, Serra X. A Wavenet for speech denoising, arXiv preprint arXiv:1706.07162, 2018..

[16] Fu S, Tsao Y, Lu X, Kawai H. Raw waveform-based speech enhancement by fully convolutional networks. In: Proc Asia, Pac. signal inf. process. assoc. annu. summit conf.; 2017. p. 6–12..

[17] Pandey A, Wang D. TCNN: temporal convolutional neural network for real-time speech enhancement in the time domain. In: IEEE int. conf. acoustics speech and signal processing (ICASSP); 2019. Brighton, United Kingdom, 2019, pp. 6875–6879..

[18] Long J, Shelhamer E., Darrell T. Fully convolutional networks for semantic segmentation. In: Proc. IEEE conf. comput. vision pattern recogn; 2015. pp. 3431–3440..

[19] Pandey A, Wang D. A new framework for CNN-based speech enhancement in the time domain. IEEE/ACM Trans Audio Speech Lang Process 27:7;2019. 1179–1188.

[20] Bai S, Kolter JZ, Koltun V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling, arXiv preprint arXiv:1803.01271, 2018..

[21] Dauphin YN, Fan A, Auli M, Grangier D. Language modeling with gated convolutional networks, arXiv preprint arXiv:1612.08083, 2017..

[22] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition, arXiv preprint, arXiv:1512.03385, 2015..

[23] He K, Zhang X, Ren S, Sun J. Delving deep into rectifiers: surpassing human-level performance on Imagenet classification. In: Proc. int. conf. computer vision; 2015. p. 1026–1034.

[24] Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariance shift. In Int. conf. mach. learn.; 2015. p. 448-456.

[25] Horeiter S, Schmidhuber J. Long short-term memory. Neural Comput 1997;9(8):1735–80.

[26] Gers FA, Schmidhuber J, Cummins F. Learning to forget: continual prediction with LSTM. Neural Comput 12:10;2000. p. 2451–2471.

[27] Pascanu R, Mikolov T, Bengio Y. On the difficulty of training recurrent neural networks. In: Proc. int. conf. mach. learn.; 2013. p. 1310–1318.

[28] van den Oord A, et al. Conditional image generation with pixelcnn decoders. In: Proc. adv. neural inf. process. syst.; 2016. p. 4790–4798.

[29] Vincent E, Gribonval R, Fevotte C. "Performance measurement in blind audio source separation. IEEE Trans Audio Speech Lang Process 2006;14(4):1462–9.

[30] Moore A, Parada PP, Naylor P. Speech enhancement for robust automatic speech recognition: evaluation using a baseline system and instrumental measures. Comput Speech Lang Nov. 2016;46:574–84.

[31] Thomsen DA, Andersen CE. Speech enhancement and noise-robust automatic speech recognition. Aalborg, Denmark: Aalborg Univ; 2015.

[32] Perceptual evaluation for speech quality (PESQ), and objective method for end-to-end speech quality assessment of narrowband telephone networks and speech codesc, ITU, ITU-T Rec. P. 862, 2000.

[33] Taal CH, Hendriks RC, Heusdens R, Jensen J. An algorithm for intelligibility prediction of time-frequency weighted noisy speech. IEEE Trans Audio Speech Lang Process 2011;19(7):2125–36.

[34] Hershey JR, Chen Z, Le Roux J, Watanabe S. Deep clustering: discriminative embeddings for segmentation and separation. IEEE int. conf. acoustics speech and signal processing (ICASSP) 2016:31–5.

[35] Snyder D, Chen G, Povey D. Musan: a music, speech, and noise corpus, arXiv preprint arXiv:1510.08484, 2015..