Gated Residual Networks With Dilated Convolutions for Monaural Speech Enhancement

Ke Tan, *Student Member, IEEE*, Jitong Chen, and DeLiang Wang, *Fellow, IEEE*

***Abstract*—For supervised speech enhancement, contextual information is important for accurate mask estimation or spectral mapping. However, commonly used deep neural networks (DNNs) are limited in capturing temporal contexts. To leverage long-term contexts for tracking a target speaker, we treat speech enhancement as a sequence-to-sequence mapping, and present a novel convolutional neural network (CNN) architecture for monaural speech enhancement. The key idea is to systematically aggregate contexts through dilated convolutions, which significantly expand receptive fields. The CNN model additionally incorporates gating mechanisms and residual learning. Our experimental results suggest that the proposed model generalizes well to untrained noises and untrained speakers. It consistently outperforms a DNN, a unidirectional long short-term memory (LSTM) model, and a bidirectional LSTM model in terms of objective speech intelligibility and quality metrics. Moreover, the proposed model has far fewer parameters than DNN and LSTM models.**

***Index Terms*—Dilated convolutions, residual learning, gated linear units, sequence-to-sequence mapping, speech enhancement.**

I. Introduction

MONAURAL speech separation is the task of separating target speech from a single-microphone recording, which may include nonspeech noise, interfering speech and room reverberation. It has a wide range of real-world applications such as robust automatic speech recognition and hearing aids design. In this study, we focus on monaural speech separation from background noise, which is also known as speech enhancement.

Monaural speech separation has been extensively studied in the speech processing community for decades. In recent years, speech separation has been formulated as supervised learning, inspired by the concept of time-frequency (T-F) masking in computational auditory scene analysis (CASA) [40]. The ideal binary mask (IBM) [39], which assigns 1 to a T-F unit if the target energy within the unit exceeds the interference energy and 0 otherwise, is the first training target used in supervised speech separation. More recent training targets include the ideal ratio mask (IRM) [43] and the phase-sensitive mask (PSM) [7], and mapping-based targets corresponding to the magnitude or power spectra of target speech [48].

Over the last several years, supervised speech separation has greatly benefited from the use of deep learning. Wang and Wang [44] first introduced deep neural networks to address speech separation, where DNNs are trained as binary classifiers to predict the IBM in order to remove background noise. A more recent study has demonstrated that ratio masking yields better speech quality than binary masking [43]. Subsequently, Xu *et al.* [48] employed a DNN to learn the mapping function from the log power spectrum of noisy speech to that of clean speech. Their experimental results indicate that the trained DNN leads to higher perceptual evaluation of speech quality (PESQ) [30] scores than a traditional enhancement method.

The last decade has witnessed the tremendous success of CNNs in the fields of computer vision and natural language processing. A typical CNN architecture comprises a cascade of convolutional layers, subsampling layers and fully connected layers. Although CNNs have been used for speech separation in recent years, none of them achieve substantial performance improvement over a DNN. In [19], a convolutional maxout neural network (CMNN) is employed to estimate the IRM for speech enhancement. Experimental results show that CMNN yields comparable PESQ gains compared to DNN-separated speech. Another study [26] uses a convolutional encoder-decoder network (CED) to learn a spectral mapping. CED exhibits similar denoising performance compared with a DNN and an RNN, but its model size is much smaller. Moreover, a similar encoder-decoder architecture is developed in [21]. Other studies [9], [38], [24], [1], [14], [15] using CNN for mask estimation or spectral mapping also achieve small performance improvements over a DNN. Recently, Fu *et al.* [11] have proposed a fully convolutional network (FCN) for raw waveform-based speech enhancement. In contrast to masking and mapping based approaches that reconstruct enhanced speech using noisy phase, FCN performs speech enhancement in an end-to-end manner, and allows for a straightforward mapping from a noisy waveform to the corresponding clean waveform. An extended study [10] follows the same framework to construct an utterance-based enhancement model and uses short-time

Manuscript received April 3, 2018; revised July 22, 2018, September 7, 2018, and October 7, 2018; accepted October 11, 2018. Date of publication October 15, 2018; date of current version October 29, 2018. This work was supported in part by two NIDCD Grants (R01 DC012048 and R01 DC015521), and in part by the Ohio Supercomputer Center. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Tuomas Virtanen. *(Corresponding author: Ke Tan.)*

K. Tan is with the Department of Computer Science and Engineering, Ohio State University, Columbus, OH 43210-1277 USA (e-mail: tan.650@osu.edu).

J. Chen was with the Department of Computer Science and Engineering, Ohio State University, Columbus, OH 43210-1277 USA. He is now with Silicon Valley AI Lab, Baidu Research, Sunnyvale, CA 94089 USA (e-mail: chen.2593@osu.edu).

D. L. Wang is with the Department of Computer Science and Engineering and the Center for Cognitive and Brain Sciences, Ohio State University, Columbus, OH 43210-1277 USA (e-mail: dwang@cse.ohio-state.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASLP.2018.2876171

objective intelligibility (STOI) [33] as the objective function during training. Their experimental results show 4% to 10% STOI gains over noisy speech. Another attempt is complex spectrogram enhancement using a CNN, i.e. estimating clean real and imaginary spectrograms from noisy ones [8].

Generalization to untrained conditions is crucial for any supervised learning task. In the case of speech enhancement, three important aspects of generalization are speaker, noise and signal-to-noise ratio (SNR). A simple yet effective method to deal with noise generalization and SNR generalization is to include many different noise types and SNR levels in a training set [3], [43]. Similarly, to tackle speaker generalization would be to train with many speakers. However, recent studies [2], [23] suggest that the capacity of a feedforward DNN in modeling a large number of speakers is limited. For a DNN, a window of consecutive time frames is typically utilized to provide temporal contexts for mask estimation or spectral mapping. Without the ability to leverage longer term information, a DNN tends to treat segments of training utterances as if they come from a single speaker [2]. When exposed to a large number of training speakers, DNN tends to mistake background noise segments for target speech, especially when background noise includes speech components (e.g., babble noise). As suggested in [2], it would be better to formulate speech enhancement as a sequence-to-sequence mapping in order to leverage long-term contexts. With such a formulation, Chen *et al.* [2] proposed a recurrent neural network (RNN) with LSTM layers to address speaker generalization. After training with many speakers and noises, the LSTM model works well on untrained speakers, and significantly outperforms a DNN based model in terms of STOI. Earlier works [46], [45] also showed that RNNs are more effective than DNNs for speech enhancement.

In a preliminary study, we recently developed a novel gated residual network (GRN) with dilated convolutions to address monaural speech enhancement [34]. The proposed GRN was inspired by recent success of dilated convolutions in image segmentation [4], [49], [50]. Compared with conventional convolutions, dilated convolutions expand receptive fields without loss of resolution while retaining the network depth and the kernel size. A receptive field is a region in the input space that affects a particular high-level feature. With the formulation of speech enhancement as a sequence-to-sequence mapping, large receptive fields of the GRN amount to long-term contexts. Motivated by recent works [6], [36] on gated convolutional networks, gated linear units (GLUs) are additionally incorporated into the proposed network. Compared with the LSTM model in [2], the GRN shows better generalization capability for untrained speakers at different SNR levels [34]. In this study, we further develop the GRN architecture to elevate the enhancement performance. The present work mainly makes the following four changes in the approach.

First, the outputs of all the residual blocks are summated to yield high-level features which are then fed into a prediction module to produce an estimate. Such skip connections preserve and integrate the knowledge learned by all the stacked residual blocks. Second, we redesign the frequency-dilated module to learn local spatial patterns in the T-F representation of speech along both time and frequency directions, rather than only along the frequency direction in [34]. Third, we replace rectified linear units (ReLUs) [13] by exponential linear units (ELUs) [5], which have been demonstrated to lead to not only faster convergence but also better generalization. Fourth, we evaluate the GRN with different training targets. Our experimental results suggest that the GRN achieves better performance with a mapping-based target than with a masking-based target.

Our experiments compare the proposed GRN with a DNN, a unidirectional LSTM model and a bidirectional LSTM (BLSTM) model. All the models are evaluated on the WSJ0 SI-84 dataset [28]. We find that the proposed GRN generalizes very well to untrained noises and untrained speakers, and it produces consistently higher STOI and PESQ scores than the DNN and the RNNs. Moreover, the number of learnable parameters of the GRN is one order of magnitude lower than that of the DNN and the RNNs.

The rest of this paper is organized as follows. We introduce the monaural speech enhancement problem in Section II. In Section III, we describe our proposed model in detail. Experimental setup is provided in Section IV. In Section V, we present and discuss experimental results. Section VI concludes this paper.

II. Monaural Speech Enhancement

A. Problem Formulation

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

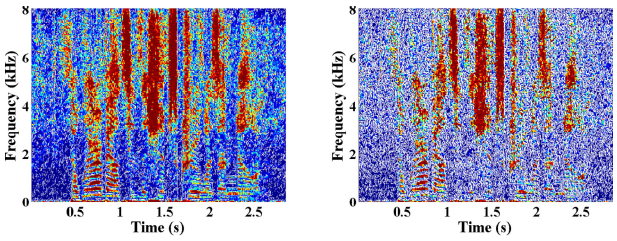
$$y(t) = s(t) + n(t) \tag{1}$$

where t indexes a time sample and $n(t)$ denotes the background noise. Supervised speech enhancement can be formulated as the process that maps from acoustic features of a noisy mixture $y(t)$ to a T-F mask or a spectral representation of target speech $s(t)$. Specifically, the input acoustic features and the corresponding desired outputs are passed into a learning machine for training. During inference, the estimated outputs and noisy mixture phases are fed into a resynthesizer to reconstruct the time-domain speech waveform.

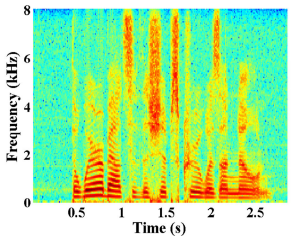
B. Training Targets

In this study, we assume that all signals are sampled at 16 kHz. A 20-ms Hamming window is employed to segment a signal into a set of time frames, where adjacent time frames are overlapped by 50%. We use 161-dimensional short-time Fourier transform (STFT) magnitude spectra as input features, which are calculated from a 320-point STFT (16 kHz $\times$ 20 ms). To demonstrate the effectiveness of the proposed model, we use three representative training targets, i.e. two masking-based targets and a mapping-based target.

1) Ideal Ratio Mask: The ideal ratio mask (IRM) is a widely used training target in supervised speech separation, which can



(a) IRM (b) PSM



(c) TMS

Fig. 1. (Color Online). Illustration of the IRM, the PSM and the TMS for a WSJ0 utterance mixed with a babble noise at -5 dB SNR.

be regarded as a soft version of the IBM [43]:

$$IRM(m,f) = \sqrt{\frac{S(m,f)^2}{S(m,f)^2 + N(m,f)^2}} \tag{2}$$

where $S(m,f)^2$ and $N(m,f)^2$ represent speech energy and noise energy within a T-F unit at time frame m and frequency channel f, respectively. Fig. 1(a) depicts an example of the IRM. In masking-based approaches for speech separation, the estimated T-F mask is element-wise multiplied by the magnitude spectrum of noisy speech to produce that of enhanced speech, which is subsequently used, along with noisy phase, to reconstruct the time-domain waveform of enhanced speech with an overlap-add method.

2) Phase-Sensitive Mask: The phase sensitive mask (PSM) incorporates the phase information into a T-F mask, and is defined on the STFT magnitudes of clean speech and noisy speech:

$$PSM(m,f) = \frac{|S(m,f)|}{|Y(m,f)|}\cos\theta \tag{3}$$

where $|S(m,f)|$ and $|Y(m,f)|$ denote spectral magnitudes of clean speech and noisy speech within a T-F unit, respectively, and θ represents the difference between the clean speech phase and the noisy speech phase within the unit. With the inclusion of the phase difference, the PSM has been demonstrated to yield a higher signal-to-distortion ratio (SDR) as compared to the IRM. Fig. 1(b) shows an example of the PSM [7]. In this study, the PSM is clipped to between 0 and 1, to fit the range of the sigmoid function.

3) Target Magnitude Spectrum: The target magnitude spectrum (TMS) of clean speech, i.e. $|S(m,f)|$, is a standard training target in mapping-based approaches [25], [16]. An example of the TMS is illustrated in Fig. 1(c). In mapping-based approaches, the estimated magnitude spectrum is combined with noisy phase to produce the enhanced speech waveform.

III. System Description

A. Dilated Convolutions

In convolutional neural networks, contextual information is augmented typically through the expansion of the receptive fields. One way to achieve this goal is to increase the network depth, which decreases computational efficiency and typically results in vanishing gradients [41]. Another way is to enlarge the kernel size, which likewise raises computational burden and training time. To solve this problem effectively, Yu and Koltun [49] first proposed dilated convolutions for multi-scale context aggregation in image segmentation. Their work is based upon the fact that dilated convolutions can exponentially expand receptive fields without losing resolution or coverage. The experimental results indicate their context module increases the accuracy of segmentation systems.

Formally, a 2-D discrete convolution operator $*$, which convolves signal F with kernel k of size $(2m+1)\times(2m+1)$, is defined as

$$(F * k)(\mathbf{p}) = \sum_{\mathbf{s}+\mathbf{t}=\mathbf{p}} F(\mathbf{s})k(\mathbf{t}) \tag{4}$$

where $\mathbf{p},\mathbf{s}\in\mathbb{Z}^2$ and $\mathbf{t}\in[-m,m]^2\cap\mathbb{Z}^2$. Here $\mathbb{Z}$ denotes the set of integers. A dilated version of the operator $*$, which is denoted by $*_r$, can be defined as

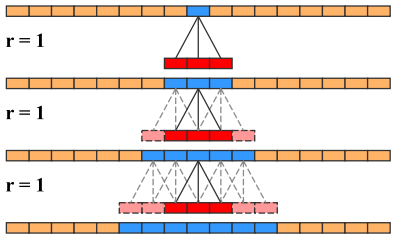
$$(F *_r k)(\mathbf{p}) = \sum_{\mathbf{s}+r\mathbf{t}=\mathbf{p}} F(\mathbf{s})k(\mathbf{t}) \tag{5}$$

where r denotes a dilation rate. Therefore, we refer to $*_r$ as an r-dilated convolution. Note that conventional convolutions can be regarded as 1-dilated convolutions. Analogously, a 1-D r-dilated convolution can be defined as $(F *_r k)(p) = \sum_{s+rt=p} F(s)k(t)$, where $p,s\in\mathbb{Z}$ and $t\in[-m,m]\cap\mathbb{Z}$. Fig. 2 illustrates conventional and dilated convolutions.

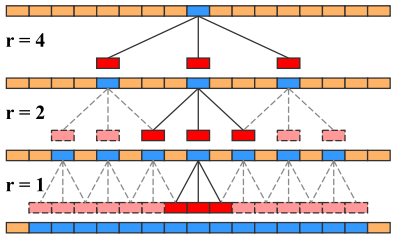
As shown in Fig. 2, the scale of the receptive fields in conventional convolutions increases linearly with the layer depth, whereas the scale of the receptive fields in dilated convolutions increases exponentially with the layer depth if the kernels are applied with exponentially increasing dilation rates.

1) Time-Dilated Convolutions: Sercu and Goel [32] developed so-called time-dilated convolutions for speech recognition by using an asymmetric version of dilated spatial convolutions (or 2-D convolutions) with dilation in the time direction but not in the frequency direction. In this study, we use a 1-D version of time-dilated convolutions, where dilation is applied to temporal convolutions (or 1-D convolutions).

2) Frequency-Dilated Convolutions: To aggregate contextual information over the frequency dimension, we create dilated spatial convolutions with kernels of size 5×5. The dilation is applied to the frequency direction but not in the time direction, and we refer to such convolutions as frequency-dilated convolutions. Note that, unlike the frequency-dilated convolutions in [34], current frequency-dilated convolutions capture contexts over both time and frequency directions.

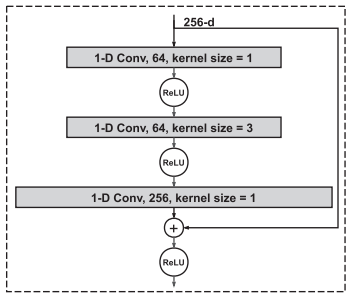


(a) Conventional convolutions

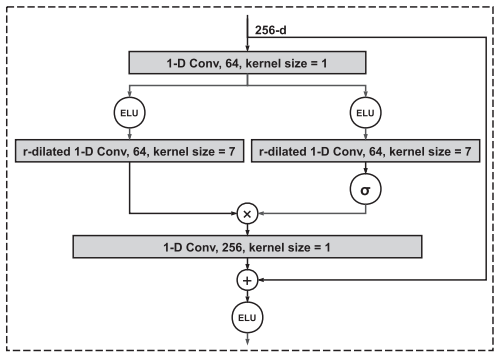


(b) Dilated convolutions

Fig. 2. (Color Online). Illustration of conventional convolutions and dilated convolutions. (a) a 1-D CNN with three conventional convolutional layers. (b) a 1-D CNN with three dilated convolutional layers, where the dilation rates r are 1, 2 and 4, respectively. The blue unit in the top layer is treated as the unit of interest, and the rest of the blue units indicate its receptive fields in each layer.



(a) A common bottleneck residual block



(b) The proposed residual block

Fig. 3. Illustration of a common bottleneck residual block and our proposed residual block. Note that σ denotes a sigmoid function and 'Conv' convolution.

B. Gated Linear Units

Gating mechanisms were first designed to facilitate the information flow over time in an RNN [18]. Long short-term memory in RNN, allows for long-term memory by introducing a memory cell controlled by an input gate and a forget gate [12]. These gates alleviate the vanishing or exploding gradient problem arising when the recurrent connections are trained with backpropagation through time [47], [27]. Van den Oord *et al.* [36] developed a multiplicative unit in the form of LSTM gates for convolutional modeling of images:

$$\begin{aligned}\mathbf{y} &= \tanh(\mathbf{x} * \mathbf{W}_1 + \mathbf{b}_1) \odot \sigma(\mathbf{x} * \mathbf{W}_2 + \mathbf{b}_2) \\ &= \tanh(\mathbf{v}_1) \odot \sigma(\mathbf{v}_2) \end{aligned} \tag{6}$$

where $\mathbf{v}_1 = \mathbf{x} * \mathbf{W}_1 + \mathbf{b}_1$ and $\mathbf{v}_2 = \mathbf{x} * \mathbf{W}_2 + \mathbf{b}_2$. $\mathbf{W}$'s and $\mathbf{b}$'s denote kernels and biases, respectively, σ represents *sigmoid* function, and $\odot$ denotes element-wise multiplication. Their work suggests LSTM-style gating potentially facilitates more complex interactions by controlling the information flow in CNNs. The gradient of LSTM-style gating is

$$\begin{aligned}\nabla[\tanh(\mathbf{v}_1) \odot \sigma(\mathbf{v}_2)] = \tanh'(\mathbf{v}_1)\nabla\mathbf{v}_1 \odot \sigma(\mathbf{v}_2) \\ + \sigma'(\mathbf{v}_2)\nabla\mathbf{v}_2 \odot \tanh(\mathbf{v}_1)\end{aligned} \tag{7}$$

where $\tanh'(\mathbf{v}_1), \sigma'(\mathbf{v}_2) \in (0, 1)$, and the prime symbol denotes differentiation. Typically, the vanishing gradient problem arises as the network depth increases, and it becomes more severe with such gating due to the downscaling factors $\tanh'(\mathbf{v}_1)$ and $\sigma'(\mathbf{v}_2)$. To tackle this problem, Dauphin *et al.* [6] introduced gated linear units (GLUs):

$$\begin{aligned}\mathbf{y} &= (\mathbf{x} * \mathbf{W}_1 + \mathbf{b}_1) \odot \sigma(\mathbf{x} * \mathbf{W}_2 + \mathbf{b}_2) \\ &= \mathbf{v}_1 \odot \sigma(\mathbf{v}_2)\end{aligned} \tag{8}$$

The gradient of the GLUs

$$\nabla[\mathbf{v}_1 \odot \sigma(\mathbf{v}_2)] = \nabla\mathbf{v}_1 \odot \sigma(\mathbf{v}_2) + \sigma'(\mathbf{v}_2)\nabla\mathbf{v}_2 \odot \mathbf{v}_1 \tag{9}$$

includes a path $\nabla\mathbf{v}_1 \odot \sigma(\mathbf{v}_2)$ without downscaling (value compression), allowing for the gradient flow through layers while retaining nonlinearity.

C. Residual Learning

He *et al.* [17] developed a deep residual learning framework by introducing the identity shortcuts, which dramatically alleviate the vanishing gradient problem. Fig. 3(a) depicts a 1-D version of the bottleneck residual block in [17]. The bottleneck design decreases the network depth while maintaining the performance. By incorporating time-dilated convolutions and GLUs into the common bottleneck residual block, we introduce a novel residual block shown in Fig. 3(b), where the kernel size in the middle layer is increased to 7 to further expand receptive fields. In addition, we replace ReLUs with ELUs to accelerate learning and improve the generalization performance.

D. Network Architecture

Our proposed GRN includes three modules, i.e. frequency-dilated module, time-dilated module and prediction module. Fig. 4 depicts the network architecture. A more detailed

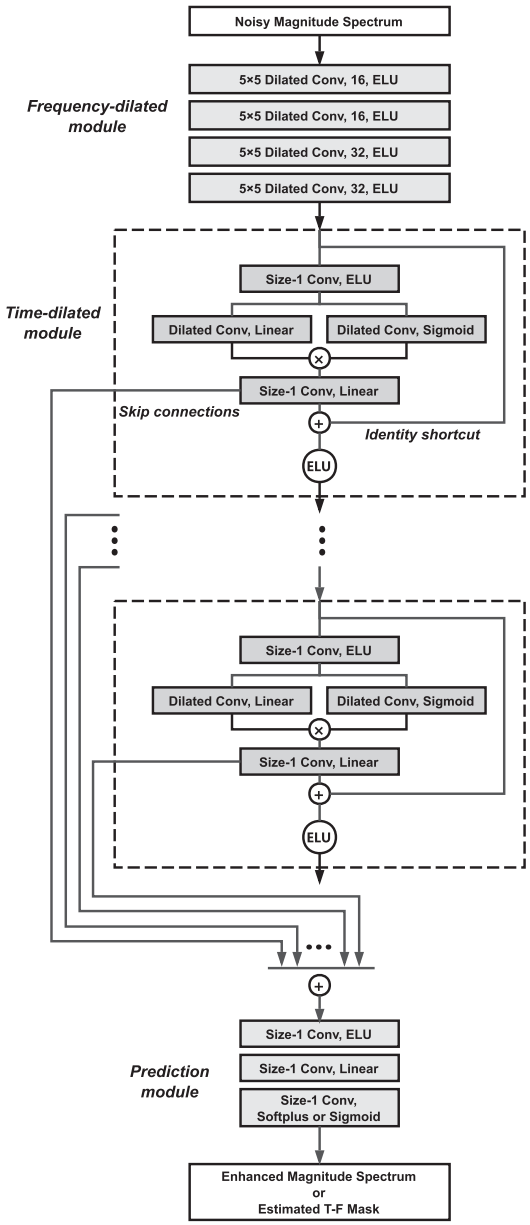


Fig. 4. Network architecture of the proposed GRN, which comprises three modules: frequency-dilated module, time-dilated module and prediction module. More details are provided in Table I.

TABLE I
ARCHITECTURE OF THE PROPOSED GRN. RESIDUAL BLOCKS ARE SHOWN IN PARENTHESES (SEE ALSO FIG. 3(B))

layer name	input size	layer hyperparameters	output size
expand_dims	$T \times 161$	-	$1 \times T \times 161$
conv2d_1	$1 \times T \times 161$	$5 \times 5, (1, 1), 16$	$16 \times T \times 161$
conv2d_2	$16 \times T \times 161$	$5 \times 5, (1, 1), 16$	$16 \times T \times 161$
conv2d_3	$16 \times T \times 161$	$5 \times 5, (1, 2), 32$	$32 \times T \times 161$
conv2d_4	$32 \times T \times 161$	$5 \times 5, (1, 4), 32$	$32 \times T \times 161$
reshape	$32 \times T \times 161$	-	$T \times 5152$
conv1d_1	$T \times 5152$	$1, 1, 128$	$T \times 128$
conv1d_2	$T \times 64$	$\left(\begin{array}{c} 1, 1, 64 \\ 7, \underline{1}, 64 \\ 1, 1, 256 \\ 1, 1, 64 \\ 7, \underline{2}, 64 \\ 1, 1, 256 \\ 1, 1, 64 \\ 7, \underline{4}, 64 \\ 1, 1, 256 \\ 1, 1, 64 \\ 7, \underline{8}, 64 \\ 1, 1, 256 \\ 1, 1, 64 \\ 7, \underline{16}, 64 \\ 1, 1, 256 \\ 1, 1, 64 \\ 7, \underline{32}, 64 \\ 1, 1, 256 \end{array}\right) \times 3$	$T \times 256$
conv1d_3	$T \times 256$	$1, 1, 256$	$T \times 256$
conv1d_4	$T \times 256$	$1, 1, 128$	$T \times 128$
conv1d_5	$T \times 128$	$1, 1, 161$	$T \times 161$

description of the architecture is given in Table I. In the table, the input sizes and the output sizes of layers are specified in the *featureMaps* $\times$ *timeSteps* $\times$ *frequencyChannels* format for 2-D convolutions, and in the *timeSteps* $\times$ *featureMaps* format for 1-D convolutions. The layer hyperparameters are shown in the $(kernelSize, dilationRate, outputChannels)$ format. Note that we apply zero-padding to all the convolutions. Batch normalization [20] is adopted in the time-dilated module and the prediction module.

1) Frequency-Dilated Module: The frequency-dilated module takes the STFT magnitude spectrum of a noisy utterance as input. The frequency-dilated module contains four stacked 2-D convolutional layers, which are used to capture local spatial patterns in the magnitude spectrum. The dilation is applied to the layers along the frequency direction with rates of 1, 1, 2 and 4, respectively. The features learned by the frequency-dilated module are then reshaped to a proper dimensionality to fit 1-D convolutions in the next module.

2) Time-Dilated Module: To model temporal dependencies, a number of residual blocks (see Fig. 3(b)) are stacked to perform time-dilated convolutions. This amounts to the time-dilated module that takes the outputs of the frequency-dilated module. We assign the dilation rates following a sawtooth wave-like fashion [42]: a set of residual blocks is grouped to form the "rising edge" of the wave which has exponentially increasing dilation rates, and two succeeding groups repeat the same pattern, e.g. 1, 2, 4, 8, 16, 32; 1, 2, 4, 8, 16, 32; 1, 2, 4, 8, 16, 32. As suggested in [49], such residual block groups enable exponential expansion of the receptive field while retaining the input resolution, which allows for aggregation of long-term contexts. Unlike the previous version of the GRN in [34], we use a type of skip connections (see Fig. 4) designed in the WaveNet [35]. In contrast to the time-dilated module in [34], such skip connections give the next module access to the outputs of all the residual blocks in the time-dilated module. An advantage is that such skip connections facilitate training by improving the flow of information and gradients throughout the network.

3) Prediction Module: After the frequency-dilated module and the time-dilated module systematically aggregate the contexts in the inputs, we employ a prediction module to perform mask estimation or spectral mapping. The prediction module comprises three convolutional layers with size-1 kernels. Of the three layers, two successive layers with ELUs and linear activations are responsible for cross-channel pooling and dimension reduction. The two layers are then followed by an output layer. There are two options for nonlinear activations in the output

layer, depending on the training target. If we use the IRM or the PSM as the training target, a sigmoid nonlinearity is applied to the output layer. If we use the TMS, a softplus activation [13] is adopted, and it is a smooth approximation to the ReLU function and can constrain the output of a network to always be positive.

The motivation for applying dilation in the time and the frequency directions separately is two-fold. First, the frequency-dilated module extracts local features, which are used by the time-dilated module to model temporal dependencies. This configuration is similar to [1], in which a vertical convolution layer captures local timbre information and a horizontal convolution layer subsequently models temporal evolution. Second, the time dimension is larger than the frequency dimension. In order to sufficiently leverage the contexts in both directions, it may be better to separately aggregate the contexts in the frequency direction and the time direction.

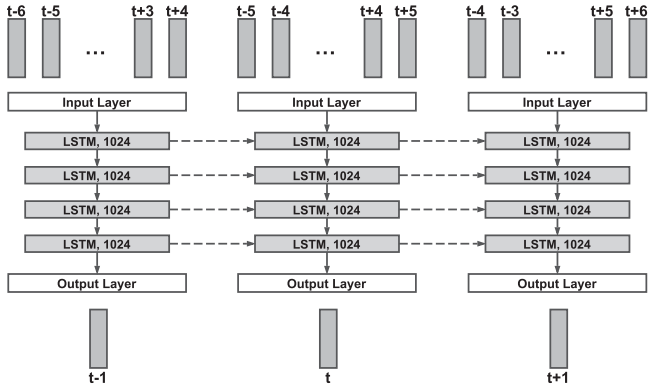


Fig. 5. An LSTM baseline with a feature window of 11 frames (5 to each side). At each time step, the 11 input frames are concatenated into a feature vector.

IV. Experimental Setup

A. Data Preparation

In our experiments, we use the WSJ0 SI-84 training set which includes 7138 utterances from 83 speakers (42 males and 41 females). Of these speakers, we set aside 6 speakers (3 males and 3 females) as untrained speakers, and train the models with the 77 remaining speakers. To investigate noise generalization of the models, we utilize four test noises which include a speech-shaped noise (SSN), a factory noise from the NOISEX-92 dataset [37], and two highly nonstationary noises (babble and cafeteria) from an Auditec CD (available at http://www.auditec.com). For training, we use 10,000 noises from a sound effect library (available at https://www.sound-ideas.com) and the total duration is about 126 hours. Note that the four test noises are different from the training noises.

Of the utterances from the 77 training speakers, we hold out 150 randomly selected utterances to create a validation set with the babble noise from the NOISEX-92 dataset. Our training set comprises 320,000 mixtures with the total duration of about 500 hours. To create a training mixture, we mix a randomly drawn training utterance with a random cut from the 10,000 training noises at an SNR level that is randomly chosen from $\{-5, -4, -3, -2, -1, 0\}$ dB.

To investigate speaker generalization of the models, we create two test sets for each noise using 6 untrained speakers and 6 trained speakers (3 males and 3 females). One test set contains 150 mixtures created from 25×6 utterances of 6 trained speakers, while the other contains 150 mixtures created from 25×6 utterances of 6 untrained speakers. We use three SNR levels for test mixtures, i.e. -5, 0 and 5 dB. Note that all test utterances are excluded from the training set.

B. Baselines and Training Details

In our experiments, we compare our proposed GRN with three other baselines, i.e. a feedforward DNN, a unidirectional LSTM model employed in [2], and a bidirectional LSTM model. For the DNN, the LSTM and the BLSTM, a feature window of 11 frames (5 to each side) is employed to estimate one frame of the target. From the input layer to the output layer, the DNN has 11×161, 2048, 2048, 2048, 2048, 2048, and 161 units, respectively; the LSTM has 11×161, 1024, 1024, 1024, 1024, and 161 units, respectively; the BLSTM has 11×161, 512, 512, 512, 512, and 161 units, respectively. Note that the features are expanded by the 11-frame feature window at each time frame for the LSTM and the BLSTM, as shown in Fig. 5.

We train the models with the Adam optimizer [22]. The initial learning rate is set to 0.001 and halved every five epochs. We use mean squared error (MSE) as the objective function. The proposed GRN, the LSTM and the BLSTM are trained with a minibatch size of 16 at the utterance level. Within a minibatch, all samples are zero-padded to have the same number of time steps as the longest sample. The feedforward DNN is trained with a minibatch size of 1024 at the frame level. The best models are selected by cross validation.

V. Experimental Results and Analysis

A. Speaker and Noise Generalization

Tables II and III present comprehensive evaluations for different models and training targets on babble ('BAB') noise and cafeteria ('CAF') noise. The numbers represent the averages over the test samples in each case. Table II lists STOI and PESQ scores for trained speakers, and Table III lists those for untrained speakers. The best scores in each case are highlighted by boldface. Overall, regardless of the training target of choice, the proposed GRN yields significant improvements over the unprocessed mixtures in terms of STOI and PESQ scores. In the -5 dB SNR case, for example, the GRN with the IRM improves the STOI score by 20.55% and the PESQ score by 0.57 as compared to the unprocessed mixtures for trained speakers. Among the three training targets, the TMS produces the best performance in both metrics. The IRM and the PSM yield similar STOI scores, while the PSM produces slightly higher PESQ scores than the IRM. Let us analyze speaker generalization of the GRN using the TMS target. For the six trained speakers, the GRN achieves 22.73% STOI improvements and 0.70 PESQ improvements over the unprocessed mixtures at -5 dB. Compared to the trained speakers, the GRN achieves similar STOI improvements (i.e. 21.81%) and PESQ improvements (i.e. 0.70) for the six untrained speakers. This reveals that, with a large

TABLE II
COMPARISONS BETWEEN MODELS AND TRAINING TARGETS IN TERMS OF STOI AND PESQ ON TRAINED SPEAKERS

metrics	STOI (in %)									PESQ								
test SNR	-5 dB			0 dB			5 dB			-5 dB			0 dB			5 dB		
noises	BAB	CAF	Avg.	BAB	CAF	Avg.	BAB	CAF	Avg.	BAB	CAF	Avg.	BAB	CAF	Avg.	BAB	CAF	Avg.
unprocessed	58.77	57.29	58.03	71.19	70.27	70.73	82.56	82.13	82.35	1.62	1.52	1.57	1.88	1.82	1.85	2.15	2.15	2.15
DNN + IRM	66.56	67.91	67.24	79.77	80.29	80.03	88.17	88.25	88.21	1.70	1.77	1.74	2.11	2.22	2.17	2.47	2.59	2.53
LSTM + IRM	77.11	74.52	75.76	86.47	84.95	85.71	91.62	91.07	91.35	2.00	2.03	2.02	2.43	2.46	2.45	2.79	2.81	2.80
BLSTM + IRM	77.57	74.22	75.90	86.53	85.10	85.82	91.84	91.23	91.54	2.01	2.02	2.02	2.45	2.47	2.46	2.80	2.83	2.82
GRN + IRM	**79.35**	**77.80**	**78.58**	**87.36**	**86.67**	**87.02**	**92.24**	**91.99**	**92.12**	**2.10**	**2.17**	**2.14**	**2.53**	**2.60**	**2.57**	**2.86**	**2.94**	**2.90**
DNN + PSM	66.27	67.74	67.01	79.62	80.09	79.86	87.94	87.83	87.89	1.67	1.83	1.75	2.13	2.28	2.21	2.53	2.65	2.59
LSTM + PSM	75.87	74.03	74.95	86.31	85.29	85.80	92.03	91.54	91.79	2.03	2.10	2.07	2.55	2.60	2.58	2.94	2.99	2.97
BLSTM + PSM	77.31	74.41	75.86	87.36	85.86	86.61	92.49	91.74	92.12	2.08	2.10	2.09	2.62	2.62	2.62	3.00	3.02	3.01
GRN + PSM	**79.54**	**77.80**	**78.67**	**87.81**	**87.05**	**87.43**	**92.97**	**92.68**	**92.83**	**2.17**	**2.25**	**2.21**	**2.65**	**2.72**	**2.69**	**3.01**	**3.08**	**3.05**
DNN + TMS	69.61	70.76	70.19	82.77	82.54	82.66	89.40	89.03	89.22	1.81	1.88	1.85	2.31	2.35	2.33	2.67	2.69	2.68
LSTM + TMS	79.27	76.79	78.03	88.57	87.11	87.84	92.80	92.14	92.47	2.15	2.15	2.15	2.63	2.60	2.62	2.97	2.94	2.96
BLSTM + TMS	79.47	76.90	78.19	88.63	87.13	87.88	93.01	92.19	92.60	2.16	2.14	2.15	2.64	2.61	2.63	2.98	2.95	2.97
GRN + TMS	**81.64**	**79.88**	**80.76**	**89.44**	**88.03**	**88.74**	**93.59**	**92.81**	**93.20**	**2.26**	**2.27**	**2.27**	**2.68**	**2.67**	**2.68**	**3.01**	**3.00**	**3.01**

TABLE III
COMPARISONS BETWEEN MODELS AND TRAINING TARGETS IN TERMS OF STOI AND PESQ ON UNTRAINED SPEAKERS

metrics	STOI (in %)									PESQ								
test SNR	-5 dB			0 dB			5 dB			-5 dB			0 dB			5 dB		
noises	BAB	CAF	Avg.	BAB	CAF	Avg.	BAB	CAF	Avg.	BAB	CAF	Avg.	BAB	CAF	Avg.	BAB	CAF	Avg.
unprocessed	58.52	57.45	57.99	70.25	69.70	69.98	81.35	81.02	81.19	1.56	1.44	1.50	1.81	1.77	1.79	2.12	2.12	2.12
DNN + IRM	65.03	67.63	66.33	78.72	80.05	79.39	87.64	88.13	87.89	1.60	1.71	1.66	2.06	2.16	2.11	2.45	2.56	2.51
LSTM + IRM	74.54	73.04	73.79	84.88	83.89	84.39	90.84	90.53	90.69	1.85	1.92	1.89	2.33	2.36	2.35	2.70	2.73	2.72
BLSTM + IRM	75.23	74.12	74.68	85.05	84.44	84.75	90.96	90.79	90.88	1.88	1.96	1.92	2.35	2.40	2.38	2.71	2.76	2.74
GRN + IRM	**77.32**	**76.91**	**77.12**	**86.17**	**86.19**	**86.18**	**91.62**	**91.63**	**91.63**	**1.98**	**2.07**	**2.03**	**2.44**	**2.52**	**2.48**	**2.80**	**2.86**	**2.83**
DNN + PSM	64.79	67.59	66.19	78.56	80.02	79.29	87.46	87.92	87.69	1.60	1.77	1.69	2.09	2.24	2.17	2.52	2.64	2.58
LSTM + PSM	74.12	73.34	73.73	84.90	84.66	84.78	91.28	91.18	91.23	1.91	2.04	1.98	2.45	2.53	2.49	2.86	2.92	2.89
BLSTM + PSM	74.67	73.65	74.16	85.64	84.86	85.25	91.55	91.24	91.40	1.91	2.04	1.98	2.49	2.53	2.51	2.89	2.92	2.91
GRN + PSM	**77.45**	**77.41**	**77.41**	**86.70**	**86.62**	**86.66**	**92.15**	**92.13**	**92.14**	**2.06**	**2.19**	**2.13**	**2.57**	**2.65**	**2.61**	**2.95**	**3.02**	**2.99**
DNN + TMS	68.13	70.78	69.46	81.99	82.93	82.46	89.43	89.58	89.51	1.71	1.85	2.28	2.25	2.31	2.28	2.64	2.66	2.65
LSTM + TMS	76.38	75.76	76.07	87.37	86.54	86.96	92.64	92.08	92.36	1.99	2.08	2.04	2.53	2.52	2.53	2.90	2.87	2.89
BLSTM + TMS	76.98	76.23	76.61	87.73	86.79	87.26	92.80	92.14	92.47	2.01	2.09	2.05	2.53	2.53	2.53	2.91	2.88	2.90
GRN + TMS	**80.18**	**79.42**	**79.80**	**88.92**	**88.04**	**88.48**	**93.40**	**92.88**	**93.14**	**2.16**	**2.23**	**2.20**	**2.63**	**2.62**	**2.63**	**2.97**	**2.96**	**2.97**

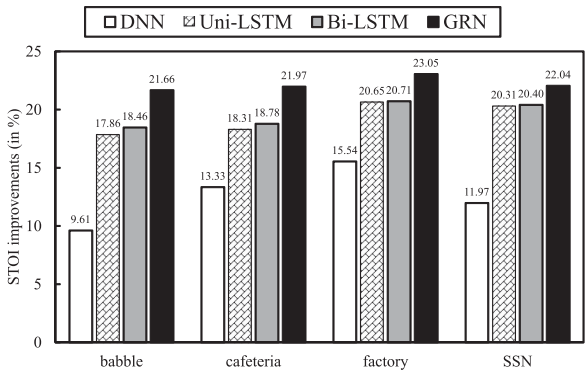


Fig. 6. Comparisons of DNN, LSTM, BLSTM and GRN in terms of STOI improvements over unprocessed mixtures for the six untrained speakers on four different noises at −5 dB SNR.

number of training speakers, the GRN generalizes very well to untrained speakers.

Fig. 6 shows the performance of different models using the TMS in terms of STOI improvements for untrained speakers and different noises. Four noises (i.e. babble, cafeteria, factory and SSN) are used to evaluate the models. As shown in Fig. 6, the GRN consistently provides significant STOI improvements for all the noises, which implies the GRN model is noise-independent.

B. Model Comparisons

We first compare the DNN with the other three models. As shown in Tables II and III, the DNN achieves about 8.2% to 12.2% STOI improvements and 0.16 to 0.27 PESQ improvements over the unprocessed mixtures. Going from DNN to LSTM substantially improves the two metrics. This result is consistent with the findings in [2]. Even with a large context window (i.e. 11 frames), the DNN is unable to track a target speaker when exposed to a wide range of training speakers. In contrast, the other three models are capable of characterizing a target speaker by learning the long-term dependencies.

Unlike the feedforward DNN, the two RNNs (i.e. LSTM and BLSTM) model the changes over time by allowing recurrent connections. The RNNs treat speech separation as a sequence-to-sequence mapping, which is more advantageous for speaker characterization. It is worth noting that BLSTM splits the units into two directions, one for future direction (forward states) and another for past direction (backward states) [31]. Unlike LSTM that utilizes only the future information within a context window, BLSTM can access all future time frames via the backward states. As shown in Tables II and III, however, similar performance is obtained by LSTM and BLSTM, while BLSTM generalizes slightly better to untrained speakers.

Our proposed GRN consistently outperforms LSTM and BLSTM in all conditions. Take, for example the −5 dB SNR case where the TMS is used as the training target. On trained speakers, the proposed GRN improves STOI by 2.57% and PESQ by 0.12 over BLSTM. On untrained speakers, the proposed GRN improves STOI by 3.19% and PESQ by 0.15 over BLSTM. For higher SNRs, the GRN yields smaller improvements over LSTM and BLSTM. To assess the significance of the STOI and PESQ differences between the GRN and the BLSTM, we conduct one-tailed two-paired Kolmogorov-Smirnov (KS) tests. The one-tailed KS tests reject the null hypothesis for a p-value lower than 0.05, which indicates that the GRN group

TABLE IV
p-VALUES FROM ONE-TAILED TWO-PAIRED KS SIGNIFICANCE TESTS FOR TRAINED SPEAKERS

metrics	STOI			PESQ		
test SNR	-5 dB	0 dB	5 dB	-5 dB	0 dB	5 dB
IRM	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$
PSM	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$
TMS	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$

TABLE V
p-VALUES FROM ONE-TAILED TWO-PAIRED KS SIGNIFICANCE TESTS FOR UNTRAINED SPEAKERS

metrics	STOI			PESQ		
test SNR	-5 dB	0 dB	5 dB	-5 dB	0 dB	5 dB
IRM	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$
PSM	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$
TMS	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$

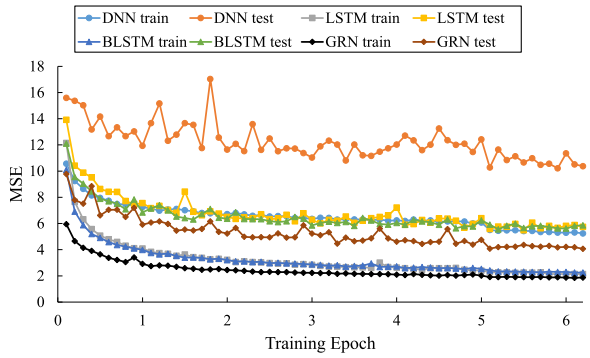


Fig. 7. (Color Online). Mean squared errors over training epochs for DNN, LSTM, BLSTM and GRN on the training set and the test set. All models are evaluated with a test set of six untrained speakers on the untrained babble noise.

of samples is significantly higher than the BLSTM group of samples. Tables IV and V show the p-values for the KS tests on trained speakers and untrained speakers, respectively, where each evaluation score was averaged over the two test noises (babble and cafeteria) before the KS tests are conducted. In all cases, the KS tests indicate the significance of STOI and PESQ improvements of GRN over BLSTM.

Fig. 7 compares the training and test MSEs of different models over training epochs. We observe that the GRN converges faster and achieves a lower training MSE and a lower test MSE than the other three models. In Fig. 8, we illustrate the STFT magnitudes of an enhanced speech utterance using the DNN, LSTM, BLSTM and GRN. The magnitudes are plotted on a log scale. We can see that the DNN-separated speech is still quite noisy. The separated speech using the other three models preserves the spectrotemporal modulation patterns of the clean speech, which are important for speech intelligibility [29]. In addition, the BLSTM separated speech and the GRN separated speech have sharper spectral transitions and less distortion compared to the LSTM separated speech.

Finally, we compare the GRN with a fully convolutional network without dilation, gating, and skip connections. The FCN is constructed by simplifying the GRN architecture. Specifically, each dilated convolution is replaced by a corresponding conventional convolution and each residual block by one convolutional layer with a kernel size of 7. Moreover, the skip connections are removed. The remaining hyperparameters are unaltered. This

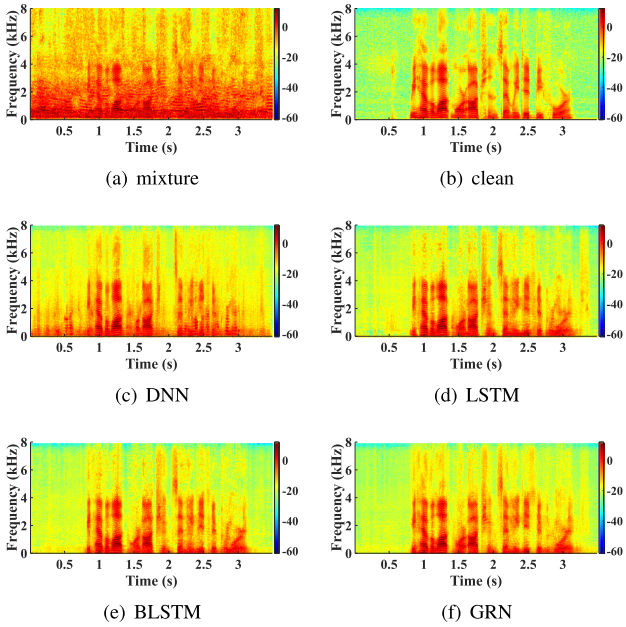


Fig. 8. (Color Online). STFT magnitudes (log scale) of a separated speech using different models. We use TMS as the training target. The unprocessed mixture is generated by mixing an utterance of an untrained speaker with babble noise at -5 dB.

TABLE VI
COMPARISONS BETWEEN FCN AND GRN IN TERMS OF STOI AND PESQ ON TRAINED SPEAKERS. THE IRM IS USED AS THE TRAINING TARGET

metrics	STOI (in %)			PESQ		
test SNR	-5 dB	0 dB	5 dB	-5 dB	0 dB	5 dB
FCN	71.88	82.81	89.80	1.89	2.30	2.66
GRN	78.58	87.02	92.12	2.14	2.57	2.90

TABLE VII
COMPARISONS BETWEEN FCN AND GRN IN TERMS OF STOI AND PESQ ON UNTRAINED SPEAKERS. THE IRM IS USED AS THE TRAINING TARGET

metrics	STOI (in %)			PESQ		
test SNR	-5 dB	0 dB	5 dB	-5 dB	0 dB	5 dB
FCN	70.83	82.23	89.48	1.80	2.24	2.62
GRN	77.12	86.18	91.63	2.03	2.48	2.83

amounts to a 26-layer FCN, which has about 1.29 million trainable parameters. Tables VI and VII present STOI and PESQ scores for trained speakers and untrained speakers, respectively. The scores are averaged over the two test noises (babble and cafeteria). As shown in the tables, the GRN substantially outperforms the FCN in all scenarios, which reveals the contributions of dilation, gating and skip connections.

C. Impact of Time-Dilated Submodules

Before we investigate the impact of time-dilated submodules in the GRN architecture, we first analyze the receptive field size of a unit in the top layer. Note that we only calculate the receptive field size for the time direction. In our proposed GRN architecture, the frequency-dilated module consists of four convolutional layers with 5×5 kernels and dilation rates 1, 1, 2 and 4, which leads to a receptive field size of $1 + (5 - 1) \times (1 + 1 + 2 + 4) = 33$. The time-dilated module comprises three submodules, each of which amounts to an

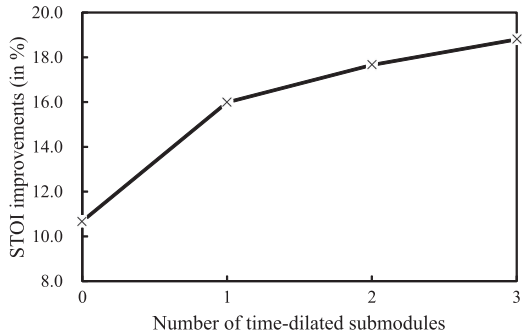


Fig. 9. Impact of the time-dilated submodules on the performance of the GRN in terms of STOI improvements over unprocessed mixtures. The models are evaluated with the six untrained speakers and the unseen babble noise. We use IRM as the training target.

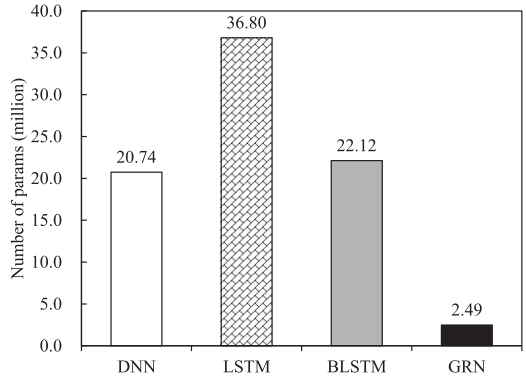


Fig. 10. Parameter efficiency comparison of DNN, LSTM, BLSTM and GRN. We compare the number of trainable parameters in different models.

additional receptive field size of $(7-1)\times(1+2+4+8+16+32)=378$. In the prediction module, all three convolutional layers use size-1 kernels, which do not expand the receptive field. Therefore, the total receptive field size of a unit in the top layer is $33+378\times 3=1167$. In other words, a unit in the top layer is affected by at most 1167 time frames of input features. Since we use a 10-ms frame shift, 1167 time frames are equivalent to $1167\times 0.01=11.67$ s (5.835 s to the past and 5.835 s to the future). Thus the proposed GRN leverages a large amount of future information like BLSTM.

We now evaluate the GRNs with different numbers of time-dilated submodules with the six untrained speakers and the untrained babble noise. Specifically, we evaluate the GRNs with 0, 1, 2 and 3 time-dilated submodules, which correspond to receptive field sizes of 33, 441, 789 and 1167, respectively. Fig. 9 compares the impact of the time-dilated submodules on the enhancement performance in terms of STOI improvements. We can see that the performance of the GRN is improved with more time-dilated submodules as more contextual information is leveraged.

D. Parameter Efficiency

Our proposed GRN provides higher parameter efficiency compared with the DNN and the RNNs due to the use of shared weights in convolution operations. Fig. 10 presents the numbers of learnable parameters in the four different models. The GRN has much fewer parameters than the other three models even though the GRN is far deeper than them. Note that we can adjust the parameter efficiency of the GRN simply by altering the number of the time-dilated submodules as discussed in Section V-C. Since computational resources are sometimes limited for real-world applications, it may be essential to achieve an optimal trade-off between enhancement performance and parameter efficiency of the model.

VI. Concluding Remarks

In this study, we have proposed a GRN model for monaural speech enhancement. The proposed model incorporates dilated convolutions, gating mechanisms and residual learning. With the formulation of speech enhancement as a sequence-to-sequence mapping, the GRN benefits from its large receptive fields upon the input T-F representation. This allows the GRN to model long-term dependencies that are critical to speaker characterization for speaker-independent enhancement. RNNs likewise learn temporal dynamics of speech, but they utilize frequency information inadequately. The proposed GRN, however, systematically aggregates contexts along both the frequency and the time directions. Our experimental results demonstrate that the GRN generalizes very well to untrained speakers and untrained noises. It consistently outperforms a DNN, a unidirectional LSTM model and a bidirectional LSTM model in terms of STOI and PESQ for both trained and untrained speakers. Another advantage of the GRN is its parameter efficiency due to the shared weights in convolutions. The GRN has one order of magnitude lower number of trainable parameters than that of an RNN with four hidden LSTM layers. This reveals the potential of CNN models for real-world speech enhancement applications in which computational efficiency is essential. We believe that the design of the CNN architecture presented in this paper is an important step towards practical monaural speech enhancement.

It should be noted that the proposed model utilizes a large amount of future information like BLSTM. Such a model cannot be used for real-time processing, which is a demand of many real-world applications. In future studies, we would devote efforts to the design of new CNN architectures that are causal or have a low latency, to meet the need of real-time speech enhancement.

References

[1] P. Chandna, M. Miron, J. Janer, and E. Gómez, "Monoaural audio source separation using deep convolutional neural networks," in *Proc. Int. Conf. Latent Variable Anal. Signal Separation*, 2017, pp. 258–266.

[2] J. Chen and D. L. Wang, "Long short-term memory for speaker generalization in supervised speech separation," *J. Acoust. Soc. Am.*, vol. 141, no. 6, pp. 4705–4714, 2017.

[3] J. Chen, Y. Wang, S. E. Yoho, D. L. Wang, and E. W. Healy, "Large-scale training to increase speech intelligibility for hearing-impaired listeners in novel noises," *J. Acoust. Soc. Am.*, vol. 139, no. 5, pp. 2604–2612, 2016.

[4] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and full connected CRFs," in *Proc. Int. Conf. Learn. Represent.*, 2015.

[5] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," 2015, arXiv:1511.07289.

[6] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, vol. 70, pp. 933–941.
[7] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 708–712.
[8] S.-W. Fu, T.-y. Hu, Y. Tsao, and X. Lu, "Complex spectrogram enhancement by convolutional neural network with multi-metrics learning," in *Proc. IEEE 27th Int. Workshop Mach. Learn. Signal Process.*, 2017, pp. 1–6.
[9] S.-W. Fu, Y. Tsao, and X. Lu, "SNR-aware convolutional neural network modeling for speech enhancement," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2016, pp. 3768–3772.
[10] S.-W. Fu, Y. Tsao, X. Lu, and H. Kawai, "End-to-end waveform utterance enhancement for direct evaluation metrics optimization by fully convolutional neural networks," 2017, arXiv:1709.03658.
[11] S.-W. Fu, Y. Tsao, X. Lu, and H. Kawai, "Raw waveform-based speech enhancement by fully convolutional networks," 2017, arXiv:1703.02205.
[12] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: continual prediction with LSTM," *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, 2000.
[13] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif Intell. Statist.*, 2011, pp. 315–323.
[14] E. M. Grais and M. D. Plumbley, "Single channel audio source separation using convolutional denoising autoencoders," in *Proc. IEEE Global Conf. Signal Inf. Process.*, 2017, pp. 1265–1269.
[15] E. M. Grais, H. Wierstorf, D. Ward, and M. D. Plumbley, "Multi-resolution fully convolutional neural networks for monaural audio source separation," in *Proc. Int. Conf. Latent Variable Anal. Signal Separation*, 2018, pp. 340–350.
[16] K. Han, Y. Wang, and D. L. Wang, "Learning spectral mapping for speech dereverberation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 4628–4632.
[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 770–778.
[18] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[19] L. Hui, M. Cai, C. Guo, L. He, W.-Q. Zhang, and J. Liu, "Convolutional maxout neural networks for speech separation," in *Proc. IEEE Int. Symp. Signal Process. Inf. Technol.*, 2015, pp. 24–27.
[20] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
[21] A. Jansson, E. Humphrey, N. Montecchio, R. Bittner, A. Kumar, and T. Weyde, "Singing voice separation with deep u-net convolutional networks," in *Proc. Int. Soc. Music Inf. Retrieval Conf.*, 2017, pp. 323–332.
[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, arXiv:1412.6980.
[23] M. Kolbæk, Z.-H. Tan, and J. Jensen, "Speech intelligibility potential of general and specialized deep neural network based speech enhancement systems," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 1, pp. 153–167, Jan. 2017.
[24] T. Kounovsky and J. Malek, "Single channel speech enhancement using convolutional neural network," in *Proc. IEEE Int. Workshop Electron., Control, Meas., Signals Their Appl. To Mechatronics*, 2017, pp. 1–5.
[25] X. Lu, Y. Tsao, S. Matsuda, and C. Hori, "Speech enhancement based on deep denoising autoencoder," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2013, pp. 436–440.
[26] S. R. Park and J. Lee, "A fully convolutional neural network for speech enhancement," 2016, arXiv:1609.07132.
[27] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 1310–1318.
[28] D. B. Paul and J. M. Baker, "The design for the wall street journal-based CSR corpus," in *Proc. Workshop Speech Natural Lang.*, 1992, pp. 357–362.
[29] R. Plomp, *The Intelligent Ear: On the Nature of Sound Perception*. Hove, U.K.: Psychology Press, 2001.
[30] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2001, vol. 2, pp. 749–752.
[31] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.
[32] T. Sercu and V. Goel, "Dense prediction on sequences with time-dilated convolutions for speech recognition," 2016, arXiv:1611.09288.
[33] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 7, pp. 2125–2136, Sep. 2011.
[34] K. Tan, J. Chen, and D. L. Wang, "Gated residual networks with dilated convolutions for supervised speech separation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2018, pp. 21–25.
[35] A. van den Oord *et al.*, "Wavenet: A generative model for raw audio," 2016, arXiv:1609.03499.
[36] A. van den Oord *et al.*, "Conditional image generation with pixelcnn decoders," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 4790–4798.
[37] A. Varga and H. J. Steeneken, "Assessment for automatic speech recognition: Ii. noisex-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Commun.*, vol. 12, no. 3, pp. 247–251, 1993.
[38] D. Wang, Y. Zou, and W. Shi, "A deep convolutional encoder-decoder model for robust speech dereverberation," in *Proc. 22nd Int. Conf. Digi. Signal Process.*, 2017, pp. 1–5.
[39] D. L. Wang, "On ideal binary mask as the computational goal of auditory scene analysis," in *Speech Separation by Humans and Machines*. New York, NY, USA: Springer, 2005, pp. 181–197.
[40] D. L. Wang and G. J. Brown, editors,*Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. New York, NY, USA: Wiley-IEEE Press, 2006.
[41] D. L. Wang and J. Chen, "Supervised speech separation based on deep learning: an overview," 2017, arXiv:1708.07524.
[42] P. Wang *et al.*, "Understanding convolution for semantic segmentation," 2017, arXiv:1702.08502, 2017.
[43] Y. Wang, A. Narayanan, and D. L. Wang, "On training targets for supervised speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 12, pp. 1849–1858, Dec. 2014.
[44] Y. Wang and D. L. Wang, "Towards scaling up classification-based speech separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 7, pp. 1381–1390, Jul. 2013.
[45] F. Weninger *et al.*, "Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR," in *Proc. Int. Conf. Latent Variable Anal. Signal Separation*, 2015, pp. 91–99.
[46] F. Weninger, F. Eyben, and B. Schuller, "Single-channel speech separation with memory-enhanced recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 3709–3713.
[47] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proc. IEEE*, vol. 78, no. 10, pp. 1550–1560, Oct. 1990.
[48] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Process. Lett.*, vol. 21, no. 1, pp. 65–68, Jan. 2014.
[49] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. Int. Conf. Learn. Represent.*, 2016.
[50] F. Yu, V. Koltun, and T. Funkhouser, "Dilated residual networks," in *Proc. Comput. Vis. Pattern Recognit.*, 2017, p. 3.

Ke Tan received the B.E. degree in electronic information engineering from the University of Science and Technology of China, Hefei, China, in 2015. He is currently working toward the Ph.D. degree with the Department of Computer Science and Engineering, Ohio State University, Columbus, OH, USA. His research interests include speech separation and enhancement, robust speech recognition, keyword spotting, and deep learning.

Jitong Chen photograph and biography not available at the time of publication.

DeLiang Wang photograph and biography not available at the time of publication.