IEEE Open Journal of Signal Processing

Received 4 November 2020; revised 9 March 2021; accepted 11 March 2021. Date of publication 18 March 2021; date of current version 16 April 2021. The review of this article was arranged by Editor B. Li.

Digital Object Identifier 10.1109/OJSP.2021.3067147

Speech Enhancement With Phase Sensitive Mask Estimation Using a Novel Hybrid Neural Network

MOJTABA HASANNEZHAD [1] (Student Member, IEEE), ZHIHENG OUYANG [1], WEI-PING ZHU [1] (Senior Member, IEEE), AND BENOIT CHAMPAGNE [2] (Senior Member, IEEE)

[1]Department of Electrical and Computer Engineering, Concordia University, Montreal, QC H3G 1M8, Canada
[2]Department of Electrical and Computer Engineering, McGill University, Montreal, QC H3A 0G4, Canada

CORRESPONDING AUTHOR: MOJTABA HASANNEZHAD (e-mail: m_hasann@encs.concordia.ca)

This work was supported by CRD grant from the Natural Sciences and Engineering Research Council (NSERC) of Canada with industrial sponsor Microchip Canada.

ABSTRACT A natural choice to model strong temporal dynamics of speech is the recurrent neural network (RNN) since it can exploit the sequential information from consecutive acoustic frames and generalizes the model well to unseen speakers. Besides, the convolutional neural network (CNN) can automatically extract sophisticated speech features that can maximize the performance of a model. In this paper, we propose a hybrid neural network model integrating a new low-complexity fully-convolutional CNN and a long short-term memory (LSTM) network, a variation of RNN, to estimate a phase-sensitive mask for speech enhancement. The model is designed to take full advantages of the temporal dependencies and spectral correlations present in the input speech signal while keeping the model complexity low. Also, an attention technique is embedded to recalibrate the useful CNN-extracted features adaptively. Furthermore, a grouping strategy is employed to reduce the LSTM complexity while keeping the performance almost unchanged. Through extensive comparative experiments, we show that the proposed model significantly outperforms some known neural network-based speech enhancement methods in the presence of highly non-stationary noises, while it exhibits a relatively small number of model parameters compared to some commonly employed DNN-based methods.

INDEX TERMS Attention technique, convolutional neural network, grouped long short-term memory, phase sensitive mask, speech enhancement.

I. INTRODUCTION

In real-world environments, speech signals are often corrupted by ambient noises during their acquisition, leading to degradation of quality and intelligibility of the speech for a listener. As one of the central topics in the speech processing area, speech enhancement aims to recover clean speech from such a noisy mixture. It brings significant advantages in various applications such as mobile communication, robust speech recognition, hearing prosthesis, hands-free smart home devices, etc. [1]. Speech enhancement methods can be categorized into single-channel (or monaural), where a single microphone is used to capture the speech, and multi-channel, which takes advantage of spatial information obtained from multiple microphones. Monaural speech enhancement is more challenging as it relies on a smaller set of observations. When combined with spatial filtering (e.g., beamforming), it also forms the basis for multi-channel techniques. In this work, our main interest lies in single-channel processing, although generalization to multi-channel processing is possible.

Researchers have advanced numerous approaches to attenuate or remove noise from a corrupted speech signal in past decades. Some well-known traditional methods such as spectral subtraction [2], which subtracts an estimate of the noise power spectrum from noisy speech spectrum, and the Wiener filtering [1], [3], where the spectrum is multiplied by a Wiener gain function, suffer from artifacts, such as musical

noise due to the presence of residual peaks in the spectrum of the processed speech. Another class of traditional methods is designed based on the estimation (e.g., spectral magnitude) of statistical properties of both speech and noise signals using the minimum mean-square error (MMSE) criterion, as in e.g. [4]. These methods generally yield lower residual noise but still produce speech distortion. Most of these traditional methods rely on some assumptions, such as speech and noise being uncorrelated or stationary, which do not fit real-world scenarios and lead to inaccurate estimation of the underlying model statistics. In particular, these methods often fail to suppress highly non-stationary noises and unexpected adverse real-world scenarios. As a complement to the traditional methods, the spectral masking based methods such as the well-known ideal binary mask (IBM) [5] and ideal ratio mask (IRM) [6] have been introduced to suppress the time-frequency (TF) cells of noisy speech spectrogram where the noise is dominant. These methods can achieve good performance if the mask is accurately estimated.

Nowadays, deep learning as a primary tool to develop data-driven information systems has led to revolutionary advances in numerous areas, including speech enhancement. In this context, speech enhancement is treated as a supervised learning problem, which does not rely on any prior assumption on the speech and noise, nor suffers from the above issues faced by traditional methods. In a supervised speech enhancement method, typically, a deep neural network (DNN) learns the highly complex relationship between a set of input signal features and the desired training target, which could be the speech spectrum or a spectral mask. In effect, an appropriately chosen training target can boost the learning and generalization capabilities of the model in unseen conditions [1].

One of the well-known DNN-based speech enhancement methods was presented in [7]. The authors therein employed a fully-connected (FC) neural network to map the log-power spectrum of the noisy speech to that of the clean one and reported significant improvements in terms of quality and intelligibility over traditional methods. Se *et al.* [8] introduced another mapping-based method where a CNN models the relationship between the noisy and clean speech spectra. Other examples of mapping-based methods for supervised speech enhancement can be found in [9], [10]. However, these approaches entail using a large training dataset to achieve an accurate mapping [11]. Instead, DNN can be exploited to predict a spectral mask applied to the noisy speech spectrogram. Yuxuan *et al.* [12] presented one of the first DNN-based mask estimation models where the network output is an IBM, and showed remarkable improvements in speech intelligibility for both normal-hearing and hearing-impaired listeners. Wang *et al.* [13] carried out a study to evaluate the enhancement performance of DNN models using different training targets. They employed an FC network to either directly estimate the short-time Fourier transform (STFT) of the spectral magnitude of the clean speech or a spectral mask, including IRM and the spectral magnitude mask (SMM). They concluded that estimating a spectral mask is more efficient than directly estimating the clean speech magnitude spectrum as far as quality and intelligibility scores are concerned.

Most of the previous methods focused on enhancing the speech magnitude solely and used the noisy phase to restore the estimated speech, thereby underestimating the impact of phase enhancement on the overall performance. Besides, the lack of clear structures in the phase spectrogram renders its estimation difficult, especially by DNN [14]. Nonetheless, the advantages of exploiting phase information for speech enhancement have been demonstrated in [15], where the authors showed that processing the phase spectrum along with magnitude can further improve perceptual speech quality and boost both objective and subjective enhancement results. Subsequently, estimation of the phase spectrum was attempted in [16] given *a prior* knowledge of the signal magnitude spectrum. Another phase enhancement method was introduced in [17] where only the phase spectrum of voiced speech frames was enhanced. Besides, some masking-based techniques have also been proposed to enhance the noisy speech phase. Erdoghan *et al.* [18] introduced a phase-sensitive spectrum approximation (PSSA) as an extension of SMM and showed that the incorporation of phase information leads to a better signal-to-distortion ratio (SDR) of the estimated clean speech in comparison to SMM. Williamson *et al.* [14] introduced a DNN-based technique to predict a complex IRM (cIRM) to enhance the noisy speech phase and magnitude simultaneously. In [14] and [19], respectively, an FC network and a composite model were employed to estimate cIRM, leading to improved quality and intelligibility. Instead of estimating the real and imaginary parts of a mask, direct estimation of the clean real and imaginary speech spectra was suggested in [20] and our previous work [11].

As discussed in the literature, DNN-based methods for speech enhancement often employ an FC network that comprises a large number of parameters. More importantly, these methods neglect temporal information even though speech exhibits strong temporal dependencies. Unlike an FC network that processes input samples independently, RNN with self-connections (to feedback previous hidden activations) treats input samples as a sequence and models the information flow over time. It makes RNN a natural choice to model the temporal dynamics of speech using information extracted from previous frames; thus, RNN can be employed as a learning machine for speech enhancement [1], [21]. Since a traditional RNN acts like an FC network with an infinite number of hidden layers and thus suffers from the vanishing and exploding gradient problem, long short-term memory (LSTM) networks were introduced instead for sequence learning. Jitong *et al.* [21] employed an LSTM network to estimate IRM and showed that compared to FC networks, LSTM significantly improves speech enhancement performance while boosting speaker generalization capability. In [22], a modified version of LSTM called bidirectional LSTM (BLSTM) was proposed, which combines information from past and future states to calculate the output sequence, thus taking full advantage of the input signal's contextual information.

The choice of features for the network input is very important in the learning process since inappropriate inputs may result in deviation of the output from its reference value. Furthermore, the more discriminative are the applied features, the less demand for the learning machine [1]. Masood *et al.* [23] conducted an extensive survey on different conventional acoustic-phonetic features and evaluated how different features affect enhancement performance. Instead of using conventional features, a CNN can be employed to extract the most appropriate input speech features for the enhancement task. As an efficient method of feature extraction, a traditional CNN made up of cascade connections of convolutional and pooling layers was employed for speech recognition in [24] and for acoustic scene classification in [25]. However, due to the small receptive field of CNN filters, the general contextual information of speech is suppressed. To address this problem, a new CNN structure with 1D convolution in the frequency domain and 2D dilated convolution in the time-frequency domain was proposed in [11] to enlarge the receptive field while keeping the number of parameters and memory footprint small.

In this paper, we propose a novel hybrid neural network that integrates a CNN and LSTM for speech enhancement based on phase sensitive mask (PSM) estimation. The novel contributions of the paper are summarized as follows.

1) A new low-complexity fully-convolutional CNN that facilitates learning, accelerates convergence, and reduces the number of model parameters is proposed to extract the most appropriate features of the input speech.
2) An attention technique is adopted to adaptively emphasize the valuable features extracted by CNN and suppress less important ones.
3) An RNN is employed to take advantage of temporal dependencies of speech and accomplish the regression between the CNN-extracted features and the mask values.
4) Different RNN variations are evaluated and analyzed to optimize the network structure in terms of performance, computation time, memory footprint, and number of model parameters.
5) A grouping strategy is adopted to reduce the number of RNN parameters. Moreover, different forms of grouping strategy are compared in terms of the objective quality of the enhanced speech and the number of trainable parameters.
6) The proposed model is evaluated using different datasets and compared to some related DNN-based methods. Different training targets are also investigated to exploit the phase information alongside magnitude enhancement so as to achieve the best performance.

The rest of this paper is organized as follows: Section II briefs several training targets incorporating phase information. Section III introduces the proposed model's network structure. Experimental results and comparisons are presented in Section IV, and finally, Section V concludes the paper.

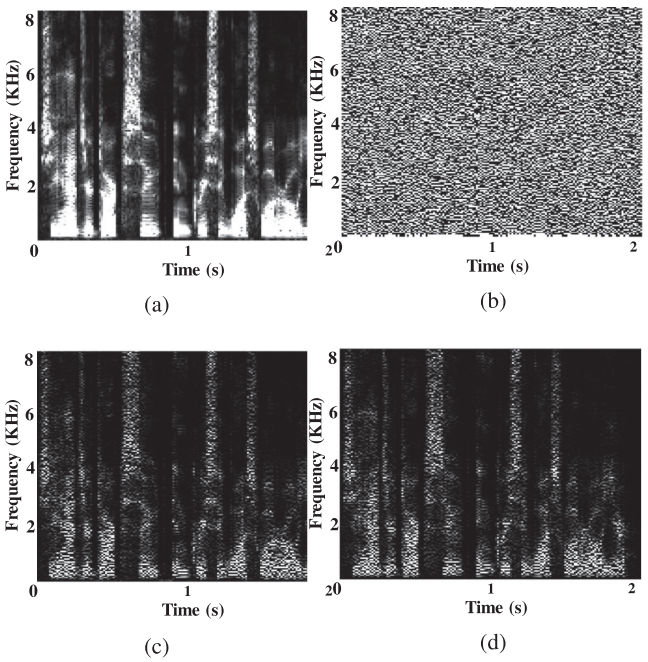


FIGURE 1. Spectrogram plot of clean speech (a) magnitude, (b) phase, (c) real component, and (d) imaginary component.

II. TRAINING TARGETS

Consider a noisy speech signal, $y(t)$, as the sum of clean speech, $x(t)$, and additive noise, $n(t)$, i.e., $y(t) = x(t) + n(t)$, where t denotes the discrete-time. Accordingly, the spectrogram of the noisy speech can be expressed as,

$$Y(k,l) = X(k,l) + N(k,l) \tag{1}$$

where $Y(k,l)$, $X(k,l)$, $N(k,l)$ represent the STFTs of $y(t)$, $x(t)$, and $n(t)$, over consecutive frames, respectively, and k and l denote the time frame and frequency discrete indices, respectively. In the sequel, we shall often represent complex STFT coefficients in terms of their magnitude and phase, as $X(k,l) = |X(k,l)|\angle X(k,l)$, or in terms of the real and imaginary parts, as $X(k,l) = \Re(X(k,l)) + i\Im(X(k,l))$. Fig. 1(a) and (b) show the spectrogram plots of the magnitude and phase of a representative clean speech utterance, while Fig. 1(c) and (d) depict the real and imaginary spectrogram components of the same utterance, respectively. As shown, the magnitude spectrum of the clean speech exhibits a clear structure that is amenable to supervised learning and thus has been considered as the training target in many studies, such as [9], [10], where the DNN-estimated magnitude spectrogram is combined with the noisy phase to resynthesize the clean speech. Besides, some studies such as [6], [13], consider IRM as the training target, as below,

$$IRM(k,l) = \sqrt{\frac{X^2(k,l)}{X^2(k,l) + N^2(k,l)}} \tag{2}$$

The estimated IRM will be multiplied by the noisy speech spectrogram, and then the estimated clean speech will be

resynthesized using the noisy phase. Meanwhile, phase processing is prominent for speech enhancement, particularly at low SNR levels, as the phase of background noise is dominant at these SNR levels. Hence, the PSM as an extension to IRM was introduced in [18] to exploit the phase information in the enhancement procedure, which is defined as,

$$\begin{aligned} PSM(k,l) &= \Re\left(\frac{X(k,l)}{Y(k,l)}\right) \\ &= \Re\left(\frac{|X(k,l)|}{|Y(k,l)|}e^{i(\angle X(k,l)-\angle Y(k,l))}\right) \\ &= \frac{|X(k,l)|}{|Y(k,l)|}cos(\zeta) \end{aligned} \tag{3}$$

where ζ is the difference of noisy and clean speech phases within each TF cell.

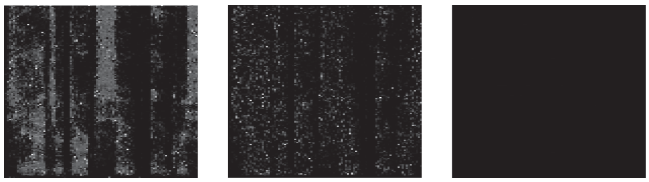
As shown in Fig. 1(b), the phase spectrogram looks quite random since the wrapped phase values fall in $(-\pi, \pi]$. Thus, direct estimation of the phase spectrogram is intractable for DNN [14]. Hence, some studies like [11] considered the complex spectrogram's real and imaginary components as the training target and had the neural network directly estimate them. Since both components appear similar, except for a shift of $\pi/2$ radians, and possess a clear structure akin to the magnitude spectrogram, as shown in Fig. 1(c) and (d), they are amenable to supervised learning. Furthermore, the similarity and correlation between the two components make it possible to estimate both components by a single neural network. From another perspective, the parameter sharing mechanism to simultaneously predict both components boosts learning and generalization capability. In particular, the authors of [11] and [26] improved the estimation of these real and imaginary components as two highly correlated subtasks through parameter sharing.

In [14], cIRM was suggested as the training target of the neural network. From $X(k,l) = M(k,l) \circ Y(k,l)$, the complex ratio mask $M(k,l)$ can be computed as,

$$M = \frac{Y_r X_r + Y_i X_i}{{Y_i}^2 + {Y_r}^2} + i\frac{Y_r X_i - Y_i X_r}{{Y_i}^2 + {Y_r}^2} \tag{4}$$

where r and i denote the real and imaginary components, and $\circ$ represents element-wise multiplication. Here, the argument (k,l) is discarded for brevity. The authors of [14] considered both real and imaginary components of M as two subtasks to be estimated by a single DNN to enhance magnitude and phase simultaneously.

Different statements about whether mapping or masking performs better for speech enhancement can be found in the literature. In [13] , it is claimed that estimating cIRM outperforms direct estimation of real and imaginary components of a complex clean speech spectrogram for speech enhancement, while in [26] the advantage of direct estimation of a complex spectrogram over cIRM is stressed. These controversial comments likely stem from different DNNs and training datasets employed in these methods. An ideal cIRM can faithfully recover the complex spectrogram of clean speech and the clear



(a) (b) (c)

FIGURE 2. (a) cIRM real part, (b) cIRM imaginary part, (c) Estimated cIRM imaginary part.

structure of its real and imaginary components, as shown in Fig. 2(a) and (b), makes it amenable to supervised learning. However, the neural network is surprisingly unable to estimate the imaginary component of cIRM. In [27], the authors supposed that it is because of the lack of a learnable pattern in the imaginary component of the cIRM. Fig. 2(c) shows the imaginary part of the cIRM estimated by a well-trained DNN. As shown, there is no information in the imaginary part. This complies with the argument made in [28] about the disability of DNN to estimate the imaginary component of cIRM, and supports the results shown in [29]. Meanwhile, the marginal advantage of cIRM over PSSA reported in [14] could result from using a different number of model parameters to estimate the training target. Based on the above observations, we are motivated to employ PSM as the training target in our proposed hybrid model.

III. SYSTEM DESCRIPTION

Fig. 3 shows the proposed hybrid model. In this model, feature extraction is executed by a fully-convolutional CNN with 1D frequency dilated convolutions. An attention block emphasizes the valuable features, and a grouped LSTM network then maps these features to the training target by exploiting the speech's temporal information. The key components of the model are discussed in the following.

A. CNN WITH FREQUENCY DILATION

A conventional CNN structure comprises pairs of convolutional and pooling layers. The conventional CNN kernels were initially designed to capture local correlations of an image for image processing purpose because image usually exhibits local correlations while speech spectrogram mostly possesses non-local correlations along the frequency axis [28].

On the one hand, non-local correlations in speech spectrogram, like the correlation of harmonics, can be exploited to improve the clean speech spectrogram's prediction. However, since the frequency dimension of speech spectrogram as the input of CNN is at the rate of a few hundred, limitation of the receptive field of convolution layers results in destroying global correlations of speech spectrogram [11]. On the other hand, the pooling layer reduces resolution and sensitivity to local variations [1]. As pointed out in [30], max-pooling keeps merely very rough information and discards the rest. Besides, average pooling neglects the importance of local structure by attenuating individual grid contribution in a local region.

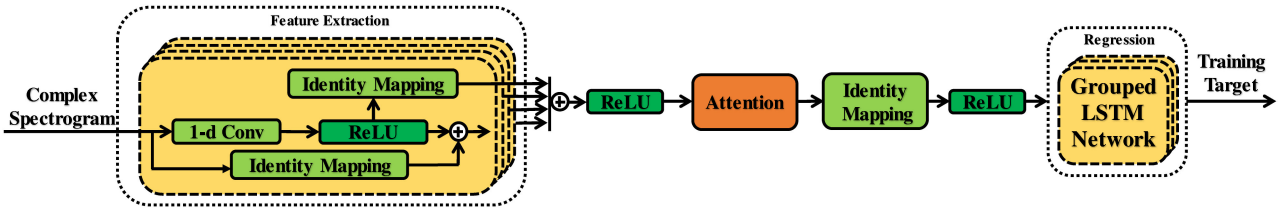


FIGURE 3. Proposed Hybrid Model.

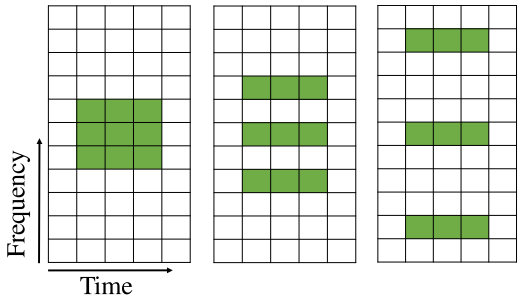


FIGURE 4. Frequency-dilated convolution. With filter size 3 × 3, the dilation rate from left to right is 1, 2, and 4, respectively. No dilation along the time axis [11].

To meet large receptive field requirements, a common practice is to enlarge the CNN kernel size along the frequency axis, like in [20] where filters with size of 25 are utilized, which leads to high complexity and consequently low speed. Another approach to enlarge the receptive field of the convolution layer is to use stride convolution, which also improves run time by reducing the size of intermediate representations and introduces some translation invariance [31]. However, striding makes the TF cell prediction overly smooth and less accurate which means reducing the spatial resolution.

To overcome the limitations mentioned above, dilated convolution was introduced and already successfully applied for imaging segmentation [32] and speech synthesis [33]. Further, by stacking dilation convolution layers, the receptive field can be exponentially expanded, while the input resolution and coverage is kept intact.

Based on the conventional convolution of a 1D signal F and a kernel k, we define the dilated convolution with dilation factor l as,

$$(k *_l F)(t) = \sum_{\tau=-\infty}^{\infty} k(\tau)F(t - l\tau) \tag{5}$$

where t and $*_l$ denote the discrete time and dilated convolution, respectively. Obviously, this definition reduces to a regular convolution when $l = 1$. Also, it can be easily extended to 2D convolution. Fig. 4 shows a dilated 2D frequency convolution with an increasing dilation factor along the frequency axis. It is worth pointing out that the CNN in the hybrid model is designed to capture the spectral information; thus, there is no dilation alongside the time axis.

Inspired by [33], we employ a fully-convolutional CNN with frequency dilated convolution to exploit the most appropriate speech features. This CNN obtains a large receptive field along the frequency axis while keeping CNN filter size relatively small. Furthermore, to facilitate the model training, residual learning and skip connection techniques [34] are adopted. It is worth mentioning that there is no pooling layer in this CNN structure.

B. ATTENTION TECHNIQUE

Attention technique is introduced in neural networks to concentrate on valuable information and ignore the rest selectively. In a CNN structure, there are many feature maps in each layer that do not have the same importance level. Attention techniques attempt to dynamically find the informative feature maps and highlight them to improve CNN's performance.

Hu *et al.* in [35] introduced a channel-wise squeeze and excitation (SAE) attention mechanism to recalibrate feature maps' information adaptively. SAE is made up of two phases: squeeze and excitation, as shown in Fig. 5(a). A channel descriptor is produced in the squeeze phase by spatially aggregating information of feature maps. Afterward, an FC network is employed to capture channel-wise dependencies. The FC-generated activations are finally multiplied with the input feature maps in the excitation phase to be delivered to the subsequent layer. In [36], a spatial SAE was introduced, as shown in Fig. 5(b), to benefit from pixel-wise spatial information. Spatial SAE performs channel squeeze and spatial excitation, i.e., the feature maps are squeezed along channels to generate a matrix with the same size of input which is then element-wise multiplied by the input feature maps to reweight each pixel of them.

Inspired by [36], we employ a spatial SAE between CNN and LSTM to emphasize significant features generated by CNN. In particular, we use both average and max pooling across different channels to squeeze the input information, as shown in Fig. 5(c). A convolutional layer then combines the results, and finally, the output is element-wise multiplied with the input feature maps.

C. LSTM AND POSSIBLE VARIATIONS

Most studies on speech enhancement try to take advantage of strong temporal dependencies of speech and provide useful temporal contextual information to a neural network by utilizing a window of frames as input due to the impact of contiguous frames on the current frame [1], [21]. However, not all

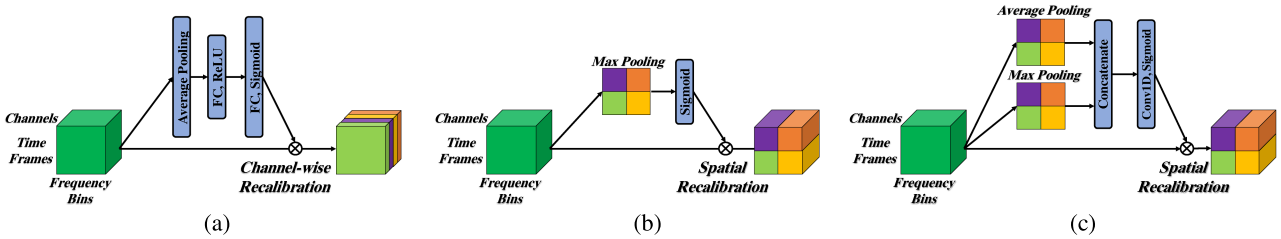


FIGURE 5. SAE attention techniques, (a) channel-wise, (b) spatial, (c) spatial using both max and average pooling.

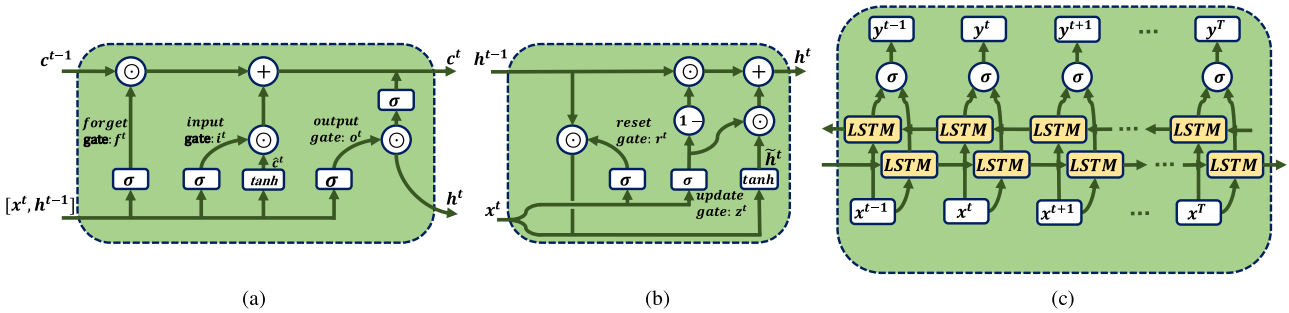


FIGURE 6. RNN variations. Block diagrams of (a) LSTM, (b) GRU, (c) BLSTM.

contiguous frames have the same impact on the current frame. Also, the information beyond this window is not exploited. In this context, an RNN treats input samples as a sequence and can model the changes over time, making it the best choice to model the temporal dynamics of speech. Furthermore, it is demonstrated in [37] that LSTM, as the most widely used type of RNN, is beneficial for low-latency enhancement and it, even without using future frames, outperforms a fully connected model with future frames. More importantly, LSTM is an effective approach for speaker-independent speech enhancement compared to an FC network that fails to model various speakers [21].

LSTM prevents a general RNN from vanishing and exploding the gradient, a problem caused by very long-term dependencies. It contains a memory cell with three gates, i.e., input gate, forget gate, and output gate, to facilitate information flow over time. The input gate controls how much information should be added to the cell; the forget gate decides how much previous information should be erased from the cell; and the output gate computes the next hidden state. Fig. 6(a) illustrates an LSTM unit. Assume $x^t \in \mathbb{R}^{M\times 1}$ is an external input at time t, and $h^{t-1} \in \mathbb{R}^{N\times 1}$ is a recurrent hidden state at time $t-1$. Then, the three gates can be defined as i^t, f^t, and $o^t \in \mathbb{R}^{N\times 1}$, respectively, which are expressed as:

$$i^t = \sigma(W_i x^t + U_i h^{t-1} + b_i) \tag{6}$$

$$f^t = \sigma(W_f x^t + U_f h^{t-1} + b_f) \tag{7}$$

$$o^t = \sigma(W_o x^t + U_o h^{t-1} + b_o) \tag{8}$$

where σ is a sigmoid activation function; $W \in \mathbb{R}^{N\times M}$, $U \in \mathbb{R}^{N\times N}$, and $b \in \mathbb{R}^{N\times 1}$ represent weight matrices and bias vector. The current value of memory cell state, c^t, is calculated based on an intermediate candidate and the previous value of the internal memory cell state, represented by $\hat{c}^t$ and c^{t-1}, respectively, as expressed below.

$$\hat{c}^t = tanh(W_c x^t + U_c h^{t-1} + b_c) \tag{9}$$

$$c^t = f^t \odot c^{t-1} + i^t \odot \hat{c}^t \tag{10}$$

$$h^t = o^t \odot tanh(c^t) \tag{11}$$

where $\odot$ denotes element-wise multiplication, and h^t is the current hidden state. Considering the dimension of cell state and input vector as N and M, the total number of parameters for an LSTM network is $4 \times (N^2 + NM + N)$.

Combining the forget and input gates in LSTM into a single one, GRU is introduced with two gates r^t and z^t, named reset and update gates, respectively. GRU as a variation of LSTM is faster and computationally more efficient than LSTM, while in some cases, it yields even better performance on less training data [38]. GRU structure is depicted in Fig. 6(b). At each step, GRU is implemented by the following set of equations,

$$z^t = \sigma(W_z x^t + U_z h^{t-1} + b_z) \tag{12}$$

$$r^t = \sigma(W_r x^t + U_r h^{t-1} + b_r) \tag{13}$$

$$\hat{h}^t = \sigma(W_h x^t + U_h(r^t \odot h^{t-1}) + b_h) \tag{14}$$

$$h^t = (1 - z^t) \odot h^{t-1} + z^t \odot \hat{h}^t \tag{15}$$

Equations (14) and (15) are similar to (9) and (10) where one gate and its associated parameters are omitted. The total number of parameters for a GRU-based network is then $3 \times (N^2 + NM + N)$ [39]. GRU has no memory unit and exposes full hidden content without any control. Thus, it

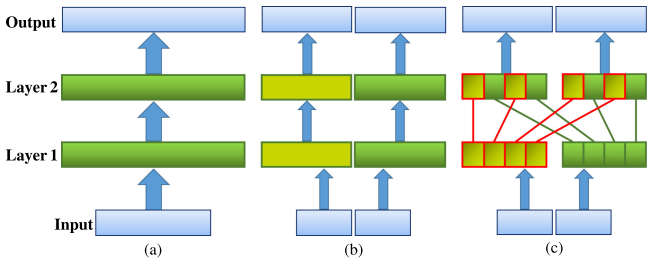


FIGURE 7. A two-layer RNN network with (a) no group strategy, (b) group strategy, (c) group strategy and representation rearrangement.

is computationally more efficient, while its performance is sometimes on par with LSTM [38].

Other variations of RNNs are bidirectional networks, such as BLSTM and BGRU, introduced to take full advantage of input information. A bidirectional cell is made up of two LSTM layers connected to the same output where the output sequence is calculated using both forward and backward hidden sequences, i.e., past and future states, simultaneously. Fig. 6(c) illustrates a BLSTM structure.

As will be seen from our experimental results and comparison in Section IV-D, LSTM is the best trade-off among the RNN variations for the proposed model for speech enhancement in terms of quality of the results, computational time, memory footprint, and the number of model parameters. As such, an LSTM network is employed to perform the final regression in our enhancement system.

D. COMPLEXITY REDUCTION USING GROUPED RECURRENT NETWORKS

RNNs have been widely used for sequence learning and achieved state-of-art results in many applications. However, RNNs suffer from high complexity caused by parameter redundancies in weight matrices that transfer hidden states between different steps and those transforming feature representations from a low to a high level. To alleviate this issue, group recurrent layers were introduced in [40] that reduce the complexity of RNN while maintaining the same level of performance. This technique is successfully employed in the RNN part of a high complexity gated convolutional recurrent networks [41].

Ignoring the bias vector b_i, the number of required parameters to implement equation (6) is $N^2 + N \times M$. If the input x and recurrent layer h are split into K disjoint groups performing independently, the number of parameters becomes,

$$K \times \left((\frac{N}{K})^2 + \frac{N}{K} \times \frac{M}{K} \right) = \frac{N^2 + N \times M}{K} \qquad (16)$$

As such, the number of RNN parameters drops by K. Fig. 7(a) and (b) depicted a standard and grouped RNN, respectively. In a grouped RNN, intra-group dependencies are efficiently learned. However, inter-group dependencies, i.e., the dependencies across different groups, are lost since individual groups perform independently. Since inter-group correlations are cut off in this architecture, the representation power drops. To tackle this issue, a parameter-free representation rearrangement technique between consecutive group layers was introduced [40], as illustrated in Fig. 7(c). It is to grant the subsequent layers access to all groups' outputs to capture the inter-group dependencies. This regrouped RNN reduces our model's complexity while keeping the performance nearly intact.

E. PSM ESTIMATION USING PROPOSED HYBRID MODEL

As shown in Fig. 3, the first stage of the proposed hybrid model is to exploit a CNN with dilated 1D frequency convolution to extract an enriched set of input speech features. CNN's input is the real and imaginary parts of the noisy speech spectrogram. In this CNN, four 1D convolutional layers are stacked with an increasing dilation rate of two, i.e., 1, 2, 4, and 8. All kernel sizes are 1×7, and the number of channels for four layers is 16, 32, 16, and 8 in order for the CNN structure to be symmetric. ReLU is employed as the activation function. Residual learning is also applied to ease training by bypassing each layer's input to its output by an identity mapping layer, with 1×1 kernels, which fixes the number of channels. The output of the 1D convolution layer and the bypassed input signal are summed as input to the next layer. Each layer's skipped outputs are then forwarded using 32-channel identity mapping layers, and their summation is later fed to the attention block. It is worth mentioning that, instead of summation, the outputs could be stacked; but, we found that the summation here yields better results. Average and max pooling operations are simultaneously applied to the input feature maps of the attention block, and the results are concatenated and then combined using a convolutional layer with kernel size of 1×7 and sigmoid activation function. The input feature maps are reweighted by their multiplication with the output of this convolutional layer. The attention block's output is later fed to the last two-channel identity mapping layer. Consequently, both channels' outputs are reshaped and concatenated to be delivered to the LSTM network.

The LSTM network has three hidden layers, each comprising 256 units. Grouping strategy and representation rearrangement are applied between layers 2 and 3. A dynamic RNN is used to perform fully-dynamic unrolling of inputs which speeds up the process in the sense that the input can have variable time steps. Here, the time step is the number of previous frames the cell used to compute the hidden state. Recurrent dropout is applied at a rate of 0.3 to mitigate the probable over-fitting problem. It is worth pointing out that because the number of parameters is limited compared to the high volume of the training dataset, the model learns merely basic data information. It means that the network does not suffer from over-fitting and is well generalized to unseen conditions.

Finally, a single affine dense layer transforms the LSTM network output to the PSM. On the one hand, choosing a mask as the training target addresses the global variance problem. As reported in [7], direct estimation of spectrogram causes an over-smoothing problem in the estimated signal compared to the reference signal, leading to a muffling effect, while a

masking-based approach does not encounter this problem. On the other hand, mask estimation narrows the dynamic range the network has to deal with. Besides, the advantage of PSM over other training targets in the hybrid model is demonstrated in Section IV-G.

IV. EXPERIMENTS

A. EXPERIMENTAL SETUP

The proposed hybrid model is evaluated with TIMIT dataset [42] consisting of utterances from 630 males and females speakers representing 8 major dialect divisions of American English, each speaking 10 phonetically-rich sentences. A 60-utterance subset is randomly selected from the dataset and kept aside for the testing stage, i.e., it is not used in training. Highly non-stationary noises from NOISEX-92 corpus [43], namely restaurant, babble, street, and factory, are selected to evaluate the model. Each noise file is divided into two parts, one for training and the other for testing, to ensure that the noise is unseen during the testing stage. Mixing random chunks of the first part of the noises mentioned above with the clean utterances at SNR levels of $-5, 0, 5$, and 10 dB results in more than 100 k mixtures (6300 utterances $\times 4$ SNR levels $\times 4$ noises) for the training stage. In the testing stage, the unseen utterances are mixed with random cuts of the unseen noise part at unmatched SNR levels of $-6, 0, 6$, and 12 dB, which gives 960 utterances (60 utterances $\times 4$ SNR levels $\times 4$ noises), half males and half females.

Furthermore, the proposed model is trained with 300 utterances from IEEE corpus [44] mixed with the first half of 20 noises from NOISEX-92[1] at SNR levels of $-5, 0, 5$, and 10 dB, i.e., 24 k mixtures (300 utterances $\times 4$ SNR levels $\times 20$ noises.) Then, 50 unseen utterances mixed with random cuts of unseen part of different noises at unmatched SNR levels of $-6, 0, 6$, and 12 dB, i.e. 4 k mixtures (50 utterances $\times 4$ SNR levels $\times 20$ noises) for the testing stage. Moreover, the model is evaluated with totally unseen noises named *Coffee Shop* and *Busy City Street* from *www.premiumbeat.com* to show the generalization capability of the model to unseen noises at unmatched SNR levels.

The sampling rate is 16 kHz for all utterances segmented using the Hanning window with a frame length of 20 ms and 50% overlap between adjacent frames, i.e., 10 ms frameshift. A 320-point discrete Fourier transform (DFT) is computed where each frame consists of 160 samples.

The cost function is defined as the mean square error (MSE) to measure the difference between the mask-filtered and ground-truth spectrograms, as follows,

$$MSE = \frac{1}{LK} \sum_{l} \sum_{k} \left[\hat{M}(k,l) Y(k,l) - X(k,l) \right]^2 \qquad (17)$$

where L and K are, respectively, the total number of time frames and that of frequency bins in each batch. An alternative is to measure the error between the estimated and ground-truth mask values, as follows,

$$MSE = \frac{1}{LK} \sum_{l} \sum_{k} \left[M(k,l) - \hat{M}(k,l) \right]^2 \qquad (18)$$

We have found that better results can be obtained if the cost function is defined between the estimated and ground-truth mask values. Adam optimizer [45] as an extension to stochastic gradient descent is used to update model parameter values during the training stage iteratively. The learning rate is initially 0.001, and then decays at a rate of 0.9 after each 1000 training steps.

Perceptual evaluation of speech quality (PESQ) and segmental signal-to-noise ratio (SSNR) are utilized as common objective metrics to evaluate speech enhancement performance. PESQ measures speech quality by comparing clean and enhanced speech. This metric's range is between -0.5 and 4.5, the higher, the better quality. SSNR computes the average signal-to-noise ratio over speech active frames. It precisely quantifies the real level of non-stationary noise in speech [46]. As per [47], both SSNR and PESQ are fairly correlated with subjective speech quality measures.

It is worth mentioning that all the experiments are performed using a single NVIDIA GeForce RTX 2080 GPU with 8 GB memory and a 2.2 GHz AMD Ryzen Threadripper 2920X 12-Core Processor. The average processing time of a 1-second utterance using the proposed model is around 8 milliseconds.

B. FEATURE EXTRACTION

Numerous acoustic-phonetic feature types have been introduced in the literature, and each could outperform others depending on the application. To investigate the impact of different inputs on the hybrid model performance, we evaluate the network with high-quality Gammatone-domain MRCG features [48], spectrum-based log Mel-filterbank energy features [49], and CNN-extracted features.

It is a common practice to concatenate original features (static) with their delta (first-order time derivative) and acceleration (second-order time derivative), called dynamic features, as they carry the temporal information of the static features [50]. As such, the dimension of log Mel-filterbank and MRCG features is 78 (26 static + 2 $\times$ 26 dynamic) and 768 (256 static + 2 $\times$ 256 dynamic), respectively. However, static and dynamic features appear in different ranges. Fig. 8(a) shows log Mel-filterbank energy features concatenated with their delta and acceleration for several frames of a speech signal where there is a considerable gap between the difference of static and dynamic features in terms of mean and variance. To unify the values and also provide unbiased involvement of different elements of feature vectors, normalization to a standard range across all the features is required [22]. Input features are commonly normalized to zero mean and unit variance, as shown in Fig. 8(b).

[1] 20 noises from NOISEX-92: airport, babble, buccaneer1, car, destroyerengine, destroyerops, exhibition, f16, factory, hfchannel, leopard, m109, machinegun, pink, restaurant, street, subway, train, volvo, and white.

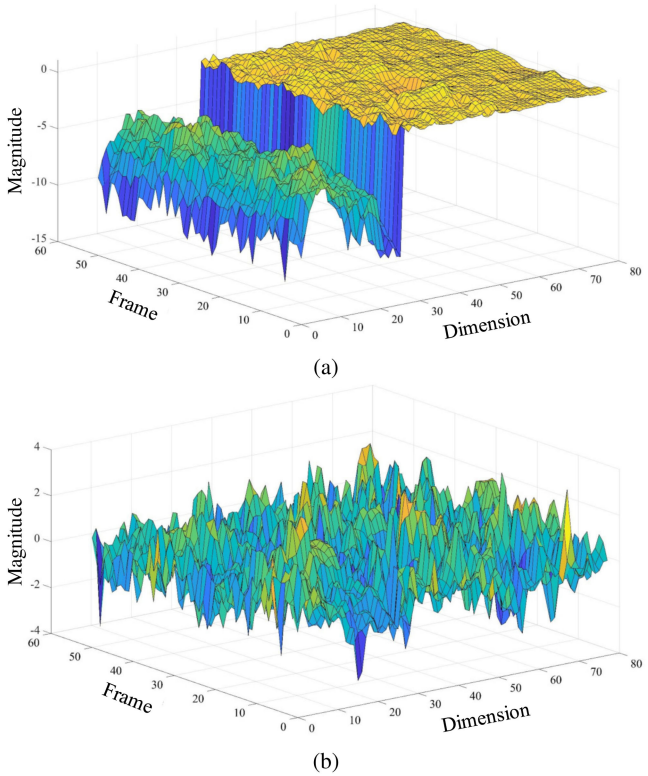


FIGURE 8. Input features visualization, (a) Log Mel-filterbank energy features concatenated with their delta and acceleration, (b) Normalized to zero mean and unit variance log Mel-filterbank energy features concatenated with their delta and acceleration.

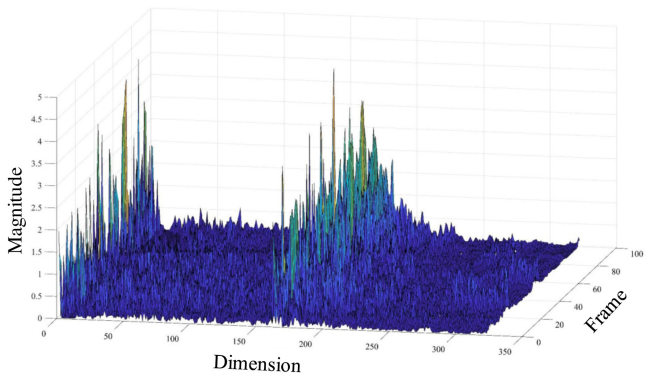


FIGURE 9. Features extracted by CNN.

Besides, since the mapping is to be done by a neural network, we can let the network also decide what sort and combination of features are better to be exploited to improve the performance. To this end, we employed a CNN with dilated 1D and 2D frequency convolution with a kernel size of 1×7 and 5×7, respectively, to observe which one gives better performance. To get a perspective about how the CNN-extracted features look like, the features for several consecutive frames are shown in Fig. 9.

Fig. 10 shows the average PESQ score improvement resulting from using different features and illustrates a comparison

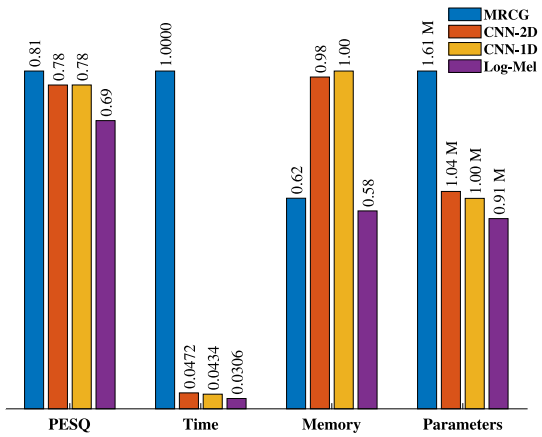


FIGURE 10. Feature comparison. (a) Average PESQ score improvement, (b) Comparison of computational time, memory, and number of parameters (in Million).

of computational time, memory footprint, and the number of the model's whole parameters using different feature extraction methods. It is to be mentioned that time and memory are normalized to 1 for brevity, and the number of model parameters is in million. Note that the comparisons are performed using TIMIT dataset and four noises, namely, babble, factory, restaurant, and street, as mentioned in Section IV. As shown in the figure, on the one hand, log Mel-filterbank energy features concatenated with dynamic features do not lead to satisfactory enhancement results in comparison with other experimented feature extraction methods. In contrast, in terms of computational time, memory footprint, and the number of parameters, they lead to the lowest. The reason is that these features do not bear the necessary and adequate information required for the network to establish an accurate mapping. On the other hand, MRCG features give very good results, indicating that this high-dimensional feature set carries a significant amount of information. Obviously, this feature set's high dimensionality leads to a high number of model parameters. Also, these features benefit from both local and contextual information as they are computed from four cochleagrams at different spectro-temporal resolutions with enriched information. However, Gammatone-domain feature extraction usually takes a long time. As shown in the figure, extracting MRCG features takes the longest time. Besides, the performance using CNN with 1D convolution is similar to that using CNN with 2D convolution, while it is better than using log-Mel features as CNN extracts the most appropriate features in our model. Exacting features using CNN takes less time than Gammatone-domain features like MRCG, requiring more memory for its computations. As seen in the figure, CNN with 1D convolution entails less time and parameters. As such, we can conclude that the best trade-off regarding performance, computational time, memory, and number of parameters is to extract features using a CNN with 1D convolution.

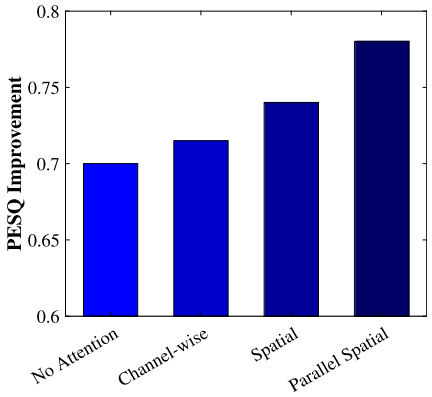


FIGURE 11. Comparison of different attention techniques in the hybrid model.

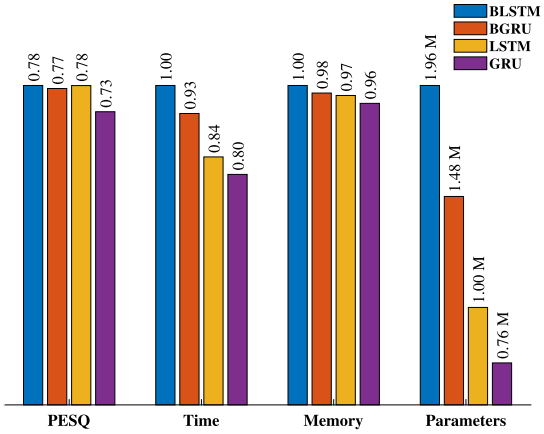


FIGURE 12. Comparison of different units, (a) Average PESQ score improvement, (b) Comparison of computational time, memory, and number of parameters (in Million).

C. BENEFIT OF ATTENTION

As explained in Section III-E, the CNN output contains 32 feature maps that will be sent to the attention block. The attention mechanism is to model the interdependencies among feature maps to boost their representative capability. As described in Section III-B, three different attention mechanisms, namely, channel-wise, spatial, and parallel spatial, are investigated in the hybrid model. The comparison in terms of the average PESQ improvement is shown in Fig. 11. Clearly, using the attention technique improves the performance in general, and moreover, the parallel spatial attention outperforms the other two techniques. This is because the importance of the feature maps' pixels is emphasized through both average and max pooling operations. As such, we adopt the parallel spatial attention technique in the hybrid model.

D. COMPARISON OF RNN TYPES

In this section, we aim to investigate the model performance using LSTM, BLSTM, GRU, and BGRU. All the networks are trained and tested with the same configuration, each comprising 3 hidden layers of 256 units. The training and testing datasets are as mentioned in Section IV-B. Fig. 12 shows the average of PESQ improvement for different noises and SNR levels, as well as computational time, memory footprint, and the number of parameters. As shown, GRU does not yield satisfactory results PESQ-wise, while in terms of other measurements, it achieves the lowest. BLSTM, BGRU, and LSTM yield almost the same results in terms of PESQ score, while the number of model parameters using BLSTM and BGRU is roughly twice and 1.5 times than LSTM. Consequently, BLSTM and BGRU take longer computational time and entail more memory than LSTM. Hence, we can conclude that LSTM is the most appropriate RNN variation for mask estimation in the proposed model.

E. EVALUATION OF DIFFERENT GROUPED RNN CONFIGURATIONS

In this section, we evaluate five LSTM network configurations as shown in Fig. 13 in terms of the PESQ score of the results and the number of model parameters to find the best trade-off for our model. Fig. 13(a) shows a standard three-layer LSTM structure with 256 units per layer where no grouping strategy is adopted. Fig. 13(b) and (c) illustrate the same network using a grouping strategy where both input and hidden layers are split into 2 or 4 groups, respectively, and representation rearrangement is applied to the hidden layers. Fig. 13(d) and (e) show similar architectures, but the grouping strategy and representation rearrangement are only adopted between layers 2 and 3, respectively.

TABLE 1. Comparison Results of Different Grouped RNN Configurations

Model	a	b	c	d	e
Avg. PESQ improvement	0.64	0.61	0.54	0.63	0.56
No. Parameters (Million)	1.53	0.79	0.42	1.00	0.74

The comparison results, in terms of the average quality and the number of parameters, are shown in Table 1. The training and testing datasets are as mentioned in Section IV-B. As illustrated, a standard LSTM network (a) yields an average PESQ score of 0.64 with 1.53 M parameters, while using the grouping strategy only between layers 2 and 3 (d) not only does yield roughly the same results concerning quality but also cuts the number of whole model parameters by 35%. Also, using the grouping strategy between every contiguous layer (b) gives 0.61 for quality with only 0.79 M parameters which means the number of whole model parameters is cut by 52%. As shown, grouping by 4 does not give good results despite whether grouping for all layers (c) or two layers (e). In this paper, we choose the grouping strategy by two between layers 2 and 3 (d).

F. LABEL COMPRESSION

A neural network would be better and easier trained if input and output are in the same range. Since the mask values (equation (3)) might have a wide range, a compression function should be adopted to make these values amenable to a neural network. The most straightforward compression method

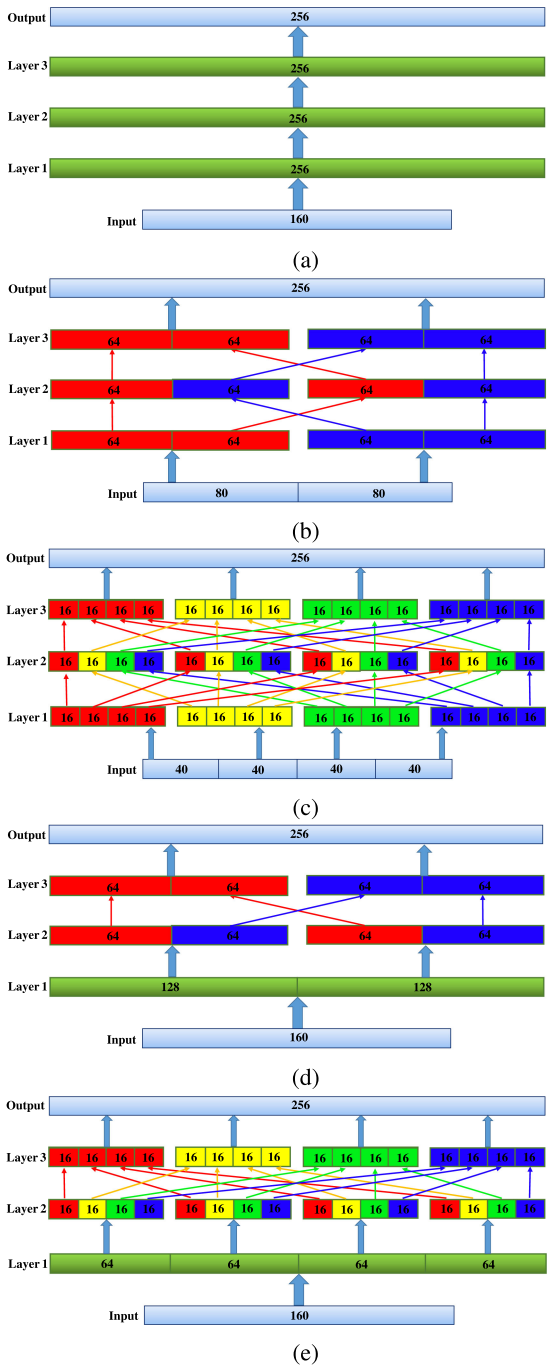


FIGURE 13. LSTM network with grouping strategy and representation rearrangement, (a) a standard LSTM network, (b) LSTM network with 2 groups and representation rearrangement for input and all layers, (c) LSTM network with 4 groups and representation rearrangement for input and all layers, (d) LSTM network with 2 groups and representation rearrangement between layers 2 and 3, (e) LSTM network with 4 groups and representation rearrangement between layers 2 and 3.

might be to limit the values within $[-1, 1]$. This technique's problem is that some mask values can go very high because of a small denominator. As such, normalization to unity with respect to these large values will result in undesired TF cells' over-compression. Other methods are hyperbolic tangent, and a variation of it introduced in [51] which we call QC, as shown in Fig. 14(a) and (b). Fig. 14(c) illustrates a slice of the label vector showing how different compression techniques influence label magnitude. As shown, employing hyperbolic tangent compression gives a better resolution while limits the label values to -1 and 1. To show the impact of label compression on the enhancement performance, we evaluated the hybrid model with different compression methods to compare the average PESQ score improvement. As shown in Fig. 14(d), hyperbolic tangent gives the best results for our model.

G. COMPARISON OF DIFFERENT TRAINING TARGETS

As mentioned in Section II, there are different claims in the literature about which training target is preferred for a DNN-based speech enhancement. As such, we compare different training targets including IRM [13], cIRM [14], complex spectrogram (CS) [11], and PSM [18] in terms of average PESQ score improvement in the hybrid model with IEEE dataset and 20 noises, as mentioned in Section IV-A. As known, all these training targets consider phase information alongside magnitude enhancement except for IRM.

Fig. 15 shows the results of comparison. Comparing IRM with other training targets reveals the advantage of incorporating phase information and its direct impact on the quality of results. Also, we can see that the quality improvement using cIRM as a mask is better than the direct estimation of complex spectrogram using the hybrid model. It is because complex spectrogram estimation is more challenging as the network has to precisely estimate every single element of the complex spectrogram, leading to more cumulative error while the network amounts to a subset of TF cells in the cIRM case. However, the hybrid model using PSM performs the best compared to using other training targets, while the number of model parameters using PSM is almost 5% less than using complex spectrogram and cIRM. This reduction in the number of model parameters stems from the PSM training target size, which is one-half of that of other training targets.

H. COMPARISON WITH OTHER DNN-BASED METHODS

We compare the proposed model with some other mapping- and masking-based techniques. For brevity's sake, we call different methods with their training targets. FFT-Mag and target magnitude spectrum (TMS) are two direct mapping-based methods introduced in [13] and [7], respectively. Both methods use FC networks with three hidden layers with 1024 and 2048 units per layer, respectively. The former captures 5 frames to exploit contextual information and uses a set of complementary features as the neural network input, while the latter uses 11 frames and log-power spectral magnitude as the neural network input. FFT-MAG and TMS predict the STFT magnitude and log-power spectral magnitude of clean speech, while both methods utilize the noisy phase to resynthesize the clean speech.

SMM, IRM, and cIRM are three masking-based methods first introduced in [13] and [14], each tested with a 3 hidden layer network and 1024 units in each layer. For all of them,

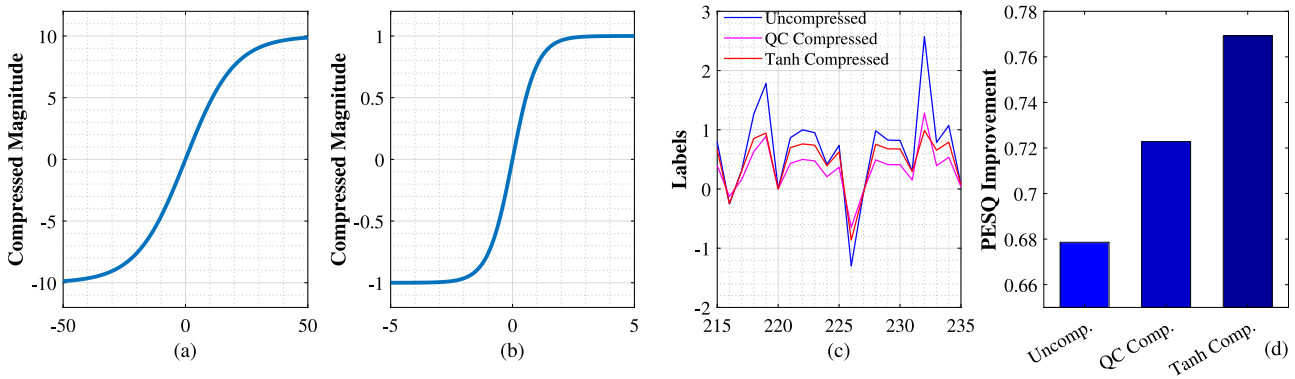


FIGURE 14. Label compression. (a) QC compression methods, (b) hyperbolic tangent, (c) a cut of mask values, (d) Average PESQ score improvement of different compression methods.

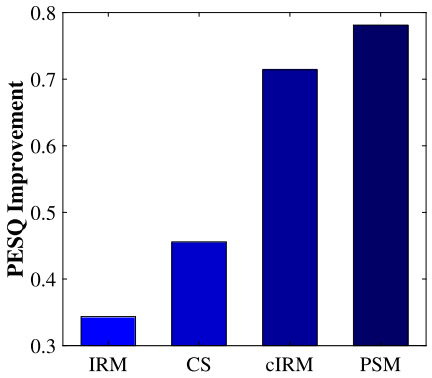


FIGURE 15. Comparison of training targets with hybrid model.

TABLE 2. The Number of Trainable Parameters in Each Method (In Million)

Method	FFT-Mag	TMS	IRM	SMM	PSSA	cIRMC	cIRM	Proposed
No. of Parameters	2.66	12.35	2.66	2.66	0.91	0.99	2.82	1.00

5 frames are used to capture temporal contextual information, and the input is a complementary set of features. cIRM estimation is also performed using a composite model in [19]. The model uses Mel-frequency cepstral coefficients (MFCCs) features and STFT of the noisy speech as input, and a parallel model. We call this model cIRMC in comparison tables. Also, a two-layer LSTM is used for phase-sensitive spectrum approximation (PSSA) in [18]. This network's input is 100-bin log-Mel filterbank features, and a sigmoid function is used as the activation function of the output FC layer. IRM, SMM, and PSSA resynthesize the estimated speech signal with the noisy phase. All networks are evaluated with the same datasets, noises, and SNR levels for a fair comparison.[2]

The number of trainable parameters in each method is presented in this Table 2. As illustrated, the number of trainable parameters of the proposed framework is less than that of

[2]Some demos are [Online]. Available: https://drive.google.com/file/d/11mqds55i7KV5-8aPFFqjQpgFCrv4p2Zi/view?usp=sharing

TABLE 3. Performance of Different Methods at −6 dB

Method	PESQ				SSNR			
	BBE	FTRY	STRT	RTRT	BBE	FTRY	STRT	RTRT
Unprocessed	1.24	1.10	1.22	1.29	-10.26	-9.65	-9.48	-9.57
	0.94	0.83	0.98	0.90	-9.50	-9.14	-9.09	-8.97
FFT-Mag	1.57	1.78	2.10	1.72	-0.25	0.38	1.39	0.34
	1.22	1.34	1.54	1.22	0.03	0.20	1.35	0.46
TMS	1.49	1.62	1.88	1.61	0.05	0.27	1.33	0.38
	1.29	1.37	1.67	1.31	0.45	0.60	1.77	0.80
IRM	1.61	1.73	2.11	1.91	-3.25	-1.95	1.05	0.06
	1.29	1.34	1.66	1.46	-2.60	-1.77	1.21	0.67
SMM	1.61	1.72	2.05	1.86	-2.70	-1.79	-0.19	-1.25
	1.23	1.32	1.66	1.37	-2.36	-1.71	0.42	-0.85
PSSA	1.56	1.73	2.06	1.74	-1.94	-0.47	1.75	0.43
	1.22	1.42	1.73	1.35	-1.59	-0.22	2.37	0.78
cIRMC	1.73	1.84	2.28	**1.89**	-0.29	0.20	2.53	0.74
	1.53	1.62	2.05	1.61	0.07	0.42	2.90	1.48
cIRM	1.60	1.81	2.30	1.94	-0.14	0.79	2.19	0.66
	1.35	1.50	1.90	1.60	-0.16	0.66	2.51	1.05
Proposed	1.69	**1.85**	**2.34**	**2.03**	**0.47**	**0.80**	**2.94**	**1.44**
	1.59	**1.75**	**2.11**	**1.76**	**1.03**	**1.25**	**3.46**	**2.10**

other models, except for PSSA and cIRMC. The methods are evaluated on the TIMIT dataset and four noises, as mentioned in Section IV. Tables 3, 4, 5, and 6 present performance scores of the mentioned methods for different noises and SNR levels where BBE, FTRY, STRT, and RTRT denote babble, factory, street, and restaurant, respectively. The top number in each table cell represents the average PESQ score with all aforementioned noises at different SNR levels for males and the bottom one for females. As shown, the proposed framework prioritizes other models for every noise at almost all SNR levels regarding PESQ score. With reference to SSNR, the proposed model outperforms other models at SNR levels of −6 and 0 dB, while IRM shows higher scores at SNR 6 dB and IRM and PSSA yield slightly better results at 12 dB SNR levels.

We also evaluated the aforementioned methods using the IEEE corpus where they are trained with TIMIT dataset at unmatched SNR levels. Results can be seen in Table 7, where PESQ and SSNR scores of the proposed model are higher than others, except for SMM that outperforms others at SNR level of −6 dB.

TABLE 4. Performance of Different Methods at 0 dB

Method	PESQ				SSNR			
	BBE	FTRY	STRT	RTRT	BBE	FTRY	STRT	RTRT
Unprocessed	1.64	1.52	1.74	1.64	-5.33	-5.20	-4.90	-4.82
	1.38	1.30	1.48	1.32	-4.95	-5.01	-4.77	-4.71
FFT-Mag	2.10	2.16	2.43	2.19	1.63	1.99	2.54	1.81
	1.61	1.73	1.86	1.66	1.40	1.71	2.40	1.88
TMS	1.92	2.06	2.28	2.04	1.90	2.16	2.85	1.89
	1.72	1.86	2.07	1.77	2.17	2.55	3.39	2.53
IRM	2.22	2.30	2.66	2.42	1.59	2.66	4.80	3.49
	1.86	1.95	2.29	2.01	2.07	2.82	5.45	4.11
SMM	2.13	2.20	2.43	2.27	0.89	1.78	2.68	1.72
	1.79	1.86	2.13	1.87	1.20	1.93	3.69	2.32
PSSA	2.12	2.27	2.56	2.27	1.93	2.97	4.51	3.12
	1.86	1.98	2.25	1.91	2.50	3.14	5.25	3.72
cIRMC	**2.30**	**2.40**	2.77	2.41	2.84	3.12	4.91	3.52
	2.08	2.15	2.48	2.12	3.12	3.32	5.33	4.15
cIRM	2.21	2.28	2.70	2.45	2.95	3.34	4.74	3.51
	1.90	2.02	2.35	2.06	3.04	3.34	4.93	3.73
Proposed	**2.30**	**2.40**	**2.83**	**2.50**	**3.46**	**3.68**	**5.36**	**4.01**
	2.12	**2.21**	**2.54**	**2.26**	**3.71**	**4.06**	**5.93**	**4.38**

TABLE 5. Performance of Different Methods at 6 dB

Method	PESQ				SSNR			
	BBE	FTRY	STRT	RTRT	BBE	FTRY	STRT	RTRT
Unprocessed	2.10	1.99	2.23	2.06	0.34	0.16	0.82	0.65
	1.88	1.78	2.01	1.82	0.30	0.28	0.81	0.67
FFT-Mag	2.50	2.52	2.65	2.56	3.13	3.25	3.50	3.33
	1.97	1.99	2.06	1.96	2.84	2.90	3.26	2.94
TMS	2.31	2.40	2.58	2.40	3.64	3.89	4.22	3.68
	2.17	2.25	2.38	2.17	3.99	4.27	4.52	4.12
IRM	2.79	2.86	3.10	2.95	6.16	**6.71**	7.78	**6.95**
	2.44	2.54	2.83	2.58	6.59	**7.07**	**8.70**	**7.48**
SMM	2.59	2.64	2.84	2.65	4.56	5.32	5.95	5.14
	2.29	2.33	2.53	2.29	5.15	5.59	6.72	5.70
PSSA	2.73	2.78	3.04	2.80	5.97	6.41	7.76	6.43
	2.41	2.49	2.67	2.40	6.47	6.55	8.23	6.84
cIRMC	**2.87**	2.89	3.16	2.90	6.33	6.38	7.32	6.44
	2.61	2.66	2.89	2.61	6.60	6.59	8.08	6.76
cIRM	2.82	2.82	3.10	2.92	6.03	6.24	7.27	6.45
	2.49	2.57	2.79	2.54	5.64	5.95	7.29	6.19
Proposed	2.82	**2.90**	**3.22**	**2.96**	**6.43**	6.63	**7.82**	6.85
	2.66	**2.70**	**3.01**	**2.72**	**6.67**	6.83	8.57	7.32

TABLE 6. Performance of Different Methods at 12 dB

Method	PESQ				SSNR			
	BBE	FTRY	STRT	RTRT	BBE	FTRY	STRT	RTRT
Unprocessed	2.55	2.45	2.71	2.51	6.05	6.09	6.77	6.40
	2.35	2.28	2.51	2.31	6.00	5.95	6.76	6.36
FFT-Mag	2.71	2.74	2.80	2.76	4.13	4.24	4.08	4.24
	2.14	2.14	2.16	2.14	3.59	3.64	3.72	3.75
TMS	2.66	2.66	2.79	2.69	5.23	5.24	5.34	5.21
	2.47	2.55	2.64	2.50	5.28	5.56	5.53	5.19
IRM	3.28	3.33	3.47	3.33	9.35	9.95	10.04	9.40
	3.05	3.13	3.27	3.05	**10.24**	**10.71**	11.02	**10.26**
SMM	3.05	3.04	3.26	3.05	8.90	9.41	9.98	9.28
	2.76	2.82	2.97	2.74	9.21	9.61	10.49	9.55
PSSA	3.23	3.19	3.38	3.21	**9.97**	**10.08**	**10.89**	**9.93**
	2.90	2.96	3.12	2.89	10.07	10.23	**11.47**	10.20
cIRMC	3.30	3.09	3.28	3.03	9.13	9.19	9.78	9.21
	2.02	3.00	3.17	2.98	9.25	9.40	10.46	9.52
cIRM	3.26	3.27	3.48	3.31	8.80	8.81	9.78	9.02
	2.94	3.00	3.17	2.98	8.08	8.23	9.29	8.31
Proposed	**3.31**	**3.35**	**3.55**	**3.40**	9.58	9.66	10.29	9.89
	3.18	**3.16**	**3.35**	**3.14**	9.76	9.63	10.80	9.94

TABLE 7. Average SSNR and PESQ Score of Different Methods Trained With TIMIT Dataset and Tested With IEEE Corpus

Method	PESQ				SSNR			
	-6	0	6	12	-6	0	6	12
Unprocessed	1.36	1.73	2.12	2.51	-9.50	-5.34	0.04	5.83
FFT-Mag	1.64	1.97	2.25	2.45	0.06	0.93	1.53	1.92
TMS	1.62	1.90	2.19	2.43	0.62	1.83	2.86	3.67
IRM	1.78	2.16	2.58	2.98	-0.88	2.24	4.96	6.85
PSSA	1.73	2.12	2.51	2.90	-0.39	2.15	4.59	7.00
SMM	**1.81**	2.12	2.44	2.74	-1.26	1.45	4.33	7.41
cIRMC	1.69	2.13	2.52	2.93	0.30	2.57	4.35	5.96
cIRM	1.80	2.19	2.60	2.96	0.18	2.23	4.15	5.89
Proposed	1.73	**2.21**	**2.68**	**3.10**	**0.86**	**3.08**	**5.31**	**7.47**

TABLE 8. Average SSNR and PESQ Score of Different Methods Evaluated With IEEE Corpus

Method	PESQ				SSNR			
	-6	0	6	12	-6	0	6	12
Unprocessed	1.40	1.74	2.14	2.55	-7.79	-3.90	1.40	7.21
FFT-Mag	1.95	2.39	2.76	3.00	1.49	3.53	5.14	6.18
TMS	1.87	2.34	2.73	3.01	1.47	3.74	5.56	6.80
IRM	1.87	2.45	3.00	3.39	-0.81	4.20	8.46	11.65
PSSA	1.90	2.46	2.97	3.37	0.71	4.84	8.47	**11.86**
SMM	1.84	2.32	2.76	3.14	-0.96	2.82	6.37	10.13
cIRMC	2.03	2.56	3.05	3.44	2.02	5.14	8.07	10.88
cIRM	1.90	2.44	2.95	3.34	1.81	4.66	7.50	10.15
Proposed	**2.13**	**2.66**	**3.11**	**3.48**	**2.95**	**5.90**	**8.93**	11.68

Table 8 shows a comparison of different models trained with IEEE corpus mixed with 20 different noises at unmatched SNR levels. Obviously, the proposed model again outperforms others in almost all the cases except for the SNR level of 12 dB, where PSSA yields marginally better results. Furthermore, the model is evaluated with unmatched utterances mixed with unseen noises, *Coffee Shop* and *Busy City Street* represented by CF and BCS, respectively, at unmatched SNR levels. The results are shown in Table 9. Clearly, the proposed model outperforms other methods in terms of both PESQ and SSNR scores.

V. CONCLUSION

In this paper, a hybrid model based on the integration of CNN and LSTM was proposed for speech enhancement. First, CNN was employed to extract the most appropriate features from the speech spectrogram. An attention technique is adopted to recalibrate the CNN feature maps. A grouped LSTM network structure was then exploited to map the CNN-extracted features to a PSM training target to benefit from strong temporal dependencies of speech while keeping the complexity low. CNN as a feature extractor was compared with some high-quality conventional acoustic features to demonstrate CNN's advantage at feature extraction. Also, the most common RNN variations have been considered for the mapping part in the proposed model, where the LSTM was shown to be the best trade-off in terms of the performance, computational time, memory footprint, and the number of model parameters. We also evaluated different grouping strategies within the LSTM to find the hybrid model's best performance. Moreover, various training targets were compared in the hybrid model to

TABLE 9. SSNR and PESQ Score of Different Methods Where Unseen Utterances are Mixed With Unseen Noises At Unmatched SNR Levels

Method	PESQ								SSNR							
	-6		0		6		12		-6		0		6		12	
	CF	BCS	CF	BCS	CF	BCS	CF	BCS	CF	BCS	CF	BCS	CF	BCS	CF	BCS
Unprocessed	1.36	1.31	1.70	1.80	2.11	2.25	2.54	2.71	-8.24	-8.10	-4.41	-4.10	0.93	1.00	6.60	6.90
FFT-Mag	1.49	1.92	2.04	2.40	2.60	2.78	2.97	3.04	-0.76	0.98	2.17	3.01	4.65	4.92	6.24	6.28
TMS	1.46	1.96	1.97	2.38	2.49	2.77	2.89	3.06	-0.63	0.79	2.28	3.15	4.68	5.18	6.42	6.73
IRM	1.52	1.80	2.09	2.42	2.70	2.99	3.19	3.38	-3.52	-1.72	1.18	2.91	6.34	7.21	10.51	10.77
PSSA	1.46	1.90	2.08	2.49	2.68	2.99	3.21	3.40	-2.03	0.04	2.39	3.61	6.48	7.25	10.34	10.89
SMM	1.52	1.83	2.06	2.29	2.55	2.78	2.99	3.19	-3.15	-1.53	0.77	2.24	4.76	6.08	8.93	10.05
cIRMC	**1.74**	2.01	2.26	2.53	2.82	3.05	3.28	3.45	**0.03**	1.45	3.72	4.51	7.16	7.56	10.30	10.61
cIRM	1.42	1.95	2.03	2.48	2.69	3.01	3.22	3.39	-0.12	0.89	2.97	3.93	6.13	6.85	9.34	9.72
Proposed	1.68	**2.12**	**2.33**	**2.68**	**2.84**	**3.12**	**3.30**	**3.50**	-0.12	**2.31**	**3.75**	**5.09**	**7.22**	**7.90**	**10.55**	**11.07**

demonstrate the advantage of PSM, which takes into account both magnitude and phase information in the enhancement process. Finally, the proposed model is compared with some well-known DNN-based speech enhancement methods, showing significant improvement in speech enhancement in the presence of highly non-stationary noise at different SNR levels. It was also shown that the hybrid model has a smaller number of model parameters as compared to some related models in the literature.

REFERENCES

[1] D. Wang and J. Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 10, pp. 1702–1726, Oct. 2018.

[2] N. Upadhyay and A. Karmakar, "Speech enhancement using spectral subtraction-type algorithms: A comparison and simulation study," *Procedia Comput. Sci.*, vol. 54, pp. 574–584, Aug. 2015.

[3] J. S. Lim and A. V. Oppenheim, "Enhancement and bandwidth compression of noisy speech," *Proc. IEEE*, vol. 67, no. 12, pp. 1586–1604, Dec. 1979.

[4] E. Plourde and B. Champagne, "Auditory-based spectral amplitude estimators for speech enhancement," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 8, pp. 1614–1623, Nov. 2008.

[5] D. Wang, "On ideal binary mask as the computational goal of auditory scene analysis," in *Proc. Speech Separation Hum. Mach.*, 2005, pp. 181–197.

[6] C. Hummersone, T. Stokes, and T. Brookes, "On the ideal ratio mask as the goal of computational auditory scene analysis," in *Proc. Blind Source Separation*, May 2014, pp. 349–368.

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

[8] S. R. Park and J. Lee, "A fully convolutional neural network for speech enhancement," in *Proc. 17th INTERSPEECH*, 2016, pp. 1993–1997.

[9] Y. Wang, J. Du, L.-R. Dai, and C.-H. Lee, "A maximum likelihood approach to deep neural network based nonlinear spectral mapping for single-channel speech separation," in *INTERSPEECH*, 2017, pp. 1178–1182.

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

[11] Z. Ouyang, H. Yu, W.-P. Zhu, and B. Champagne, "A fully convolutional neural network for complex spectrogram processing in speech enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 5756–5760.

[12] Y. Wang and D. Wang, "Towards scaling up classification-based speech separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 7, pp. 1381–1390, Jul. 2013.

[13] Y. Wang, A. Narayanan, and D. Wang, "On training targets for supervised speech separation," *IEEE/ACM Trans. Audio, Speech Lang. Process.*, vol. 22, no. 12, pp. 1849–1858, Dec. 2014.

[14] D. S. Williamson, Y. Wang, and D. Wang, "Complex ratio masking for monaural speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 3, pp. 483–492, Mar. 2015.

[15] K. Paliwal, K. Wójcicki, and B. Shannon, "The importance of phase in speech enhancement," *Speech Commun.*, vol. 53, no. 4, pp. 465–494, 2011.

[16] P. Mowlaee, R. Saeidi, and R. Martin, "Phase estimation for signal reconstruction in single-channel source separation," in *Proc. 13th Conf. Int. Speech Commun. Assoc.*, 2012, pp. 1–4.

[17] M. Krawczyk and T. Gerkmann, "STFT phase reconstruction in voiced speech for an improved single-channel speech enhancement," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 12, pp. 1931–1940, Dec. 2014.

[18] 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.

[19] M. Hasannezhad, Z. Ouyang, W.-P. Zhu, and B. Champagne, "Speech separation using a composite model for complex mask estimation," in *Proc. IEEE 63 rd Int. Midwest Symp. Circuits Syst.*, 2020, pp. 578–581.

[20] S.-W. Fu, T.-y. Hu, Y. Tsao, and X. Lu, "Complex spectrogram enhancement by convolutional neural network with multi-metrics learning," in *IEEE 27th Int. Workshop Mach. Learn. Signal Proc. (MLSP)*, 2017, pp. 1–6.

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

[22] A. Al-Dulaimi, S. Zabihi, A. Asif, and A. Mohammed, "NBLSTM: Noisy and hybrid convolutional neural network and BLSTM-Based deep architecture for remaining useful life estimation," *J. Comput. Inf. Sci. Eng.*, vol. 20, no. 2, pp. 1–12, 2020.

[23] M. Delfarah and D. Wang, "Features for masking-based monaural speech separation in reverberant conditions," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 5, pp. 1085–1094, May 2017.

[24] D. Amodei *et al.*, "Deep speech 2: End-to-end speech recognition in english and mandarin," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 173–182.

[25] J. Guo, N. Xu, L.-J. Li, and A. Alwan, "Attention based CLDNNs for short-duration acoustic scene classification," in *Proc. Interspeech*, 2017, pp. 469–473.

[26] K. Tan and D. Wang, "Complex spectral mapping with a convolutional recurrent network for monaural speech enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 6865–6869.

[27] S. Xia, H. Li, and X. Zhang, "Using optimal ratio mask as training target for supervised speech separation," in *Proc. IEEE Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, 2017, pp. 163–166.

[28] D. Yin, C. Luo, Z. Xiong, and W. Zeng, "PHASEN: A phase-and-harmonics-aware speech enhancement network," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 9458–9465.

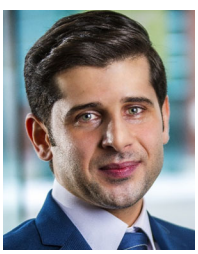
[29] M. Hasannezhad, Z. Ouyang, W.-P. Zhu, and B. Champagne, "A novel integrated CNN-GRU framework for complex ratio mask estimation in speech separation," in *Proc. IEEE 12th Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, 2020, pp. 764–768.

[30] H. Zhang and J. Ma, "Hartley spectral pooling for deep learning," *Comput. Res. Repository (CoRR)*, vol. abs/1810.04028, 2018.

[31] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," in *Workshop Contribution 3rd Int. Conf. Learn. Representations*, 2015.

[32] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. 4th Int. Conf. Learn. Representations*, 2016.

[33] A. v. d. Oord *et al.*, "Wavenet: A generative model for raw audio," *Comput. Res. Repository*, vol. abs/1609.03499, 2016.
[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[35] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7132–7141.
[36] A. G. Roy, N. Navab, and C. Wachinger, "Concurrent spatial and channel 'squeeze & excitation' in fully convolutional networks," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Interv.*, 2018, pp. 421–429.
[37] K.-L. Du and M. Swamy, "Recurrent neural networks," in *Proc. Neural Netw. Stat. Learn.*, 2019, pp. 351–371.
[38] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," 2014, *arXiv:1412.3555*.
[39] R. Dey and F. M. Salemt, "Gate-variants of gated recurrent unit (GRU) neural networks," in *Proc. IEEE 60th Int. Midwest Symp. Circuits Syst.*, 2017, pp. 1597–1600.
[40] F. Gao, L. Wu, L. Zhao, T. Qin, X. Cheng, and T.-Y. Liu, "Efficient sequence learning with group recurrent networks," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2018, pp. 799–808.
[41] K. Tan and D. Wang, "Learning complex spectral mapping with gated convolutional recurrent networks for monaural speech enhancement," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 380–390, Nov. 2019.
[42] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett, "DARPA TIMIT acoustic-phonetic continuous speech corpus CD-ROM. NIST speech disc 1-1.1," Nat. Inst. Standards Technol., Tech. Rep. NISTIR 4930, 1993, p. 27403.
[43] 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.
[44] E. Rothauser, "IEEE recommended practice for speech quality measurements," *IEEE Trans. Audio Electroacoustics*, vol. 17, no. 3, pp. 225–246, Sep. 1969.
[45] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 2nd Int. Conf. Learn. Representations*, 2014.
[46] J. H. Hansen and B. L. Pellom, "An effective quality evaluation protocol for speech enhancement algorithms," in *Proc. 5th Int. Conf. Spoken Lang. Process.*, 1998, pp. 2819–2822.
[47] Y. Hu and P. C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 1, pp. 229–238, Jan. 2007.
[48] J. Chen, Y. Wang, and D. Wang, "A feature study for classification-based speech separation at low signal-to-noise ratios," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 12, pp. 1993–2002, Dec. 2014.
[49] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Joint optimization of masks and deep recurrent neural networks for monaural source separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 12, pp. 2136–2147, Dec. 2015.
[50] X. Xiao *et al.*, "Speech dereverberation for enhancement and recognition using dynamic features constrained deep neural networks and feature adaptation," *EURASIP J. Adv. Signal Process.*, vol. 2016, no. 1, pp. 1–18, 2016.
[51] D. S. Williamson and D. Wang, "Time-frequency masking in the complex domain for speech dereverberation and denoising," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 7, pp. 1492–1501, Jul. 2017.



MOJTABA HASANNEZHAD (Student Member, IEEE) received the M.Sc. degree from Tarbiat Modares University, Tehran, Iran, in 2015. He is currently working toward the Ph.D. degree with the Department of Electrical and Computer Engineering, Concordia University, Montreal, QC, Canada. His research interests include speech enhancement, ultra wide-band radar signal processing, and machine learning.

ZHIHENG OUYANG received the B.E. degree from Hangzhou Dianzi University, Hangzhou, China, in 2017 and the M.Sc. degree from Concordia University, Montreal, QC, Canada. His research interests include machine learning for speech and audio processing.

WEI-PING ZHU (Senior Member, IEEE) received the B.E. and M.E. degrees in electrical engineering from the Nanjing University of Posts and Telecommunications, Nanjing, China, and the Ph.D. degree in electrical engineering from Southeast University, Nanjing, China, in 1982, 1985, and 1991, respectively. From 1991 to 1992, he was a Postdoctoral Fellow and from 1996 to 1998, a Research Associate with the Department of Electrical and Computer Engineering, Concordia University, Montreal, QC, Canada. During 1993–1996, he was an Associate Professor with the Department of Information Engineering, Nanjing University of Posts and Telecommunications. From 1998 to 2001, he was with hi-tech companies in Ottawa, Canada, including Nortel Networks and SR Telecom Inc. Since July 2001, he has been a full-time Faculty Member with Concordia's Electrical and Computer Engineering Department, where he is currently a Full Professor. His research interests include digital signal processing and machine learning, speech and statistical signal processing, and signal processing for wireless communication with a particular focus on MIMO systems and cooperative communication.

Dr. Zhu was an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS PART I: FUNDAMENTAL THEORY AND APPLICATIONS during 2001-2003, an Associate Editor for the *Circuits, Systems and Signal Processing* during 2006–2009, and an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS PART II: TRANSACTIONS BRIEFS during 2011–2015. He was also the Guest Editor of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS for the special issues of: Broadband Wireless Communications for High Speed Vehicles, and Virtual MIMO during 2011–2013. He was an Associate Editor for the *Journal of The Franklin Institute (JFI)* during 2015–2019. Since January 2020, he has has been a Subject Editor for JFI. He was the Secretary of Digital Signal Processing Technical Committee of the IEEE Circuits and System Society during June 2012–May 2014, and the Chair of the DSPTC during June 2014–May 2016.

BENOIT CHAMPAGNE (Senior Member, IEEE) received the B.Ing. degree in engineering physics from the Ecole Polytechnique of Montreal, Montreal, QC, Canada, in 1983, the M.Sc. degree in physics from the University of Montreal, Montreal, QC, Canada, in 1985, and the Ph.D. degree in electrical engineering from the University of Toronto, Toronto, ON, Canada, in 1990. From 1990 to 1999, he was an Assistant and Associate Professor with INRS-Telecommunications, Montreal, QC, Canada. In 1999, he joined McGill University, Montreal, QC, Canada, where he is currently a Full Professor ECE Department. He has coauthored more than 300 publications in his areas of research, which include statistical signal processing and wireless communications. He has been Associate Editor for the IEEE SIGNAL PROCESSING LETTERS and the IEEE TRANSACTION ON SIGNAL PROCESSING.