A Novel Low-Complexity Attention-Driven Composite Model for Speech Enhancement

Mojtaba Hasannezhad, Wei-Ping Zhu
Electrical and Computer Engineering
Concordia University, Montreal, Canada
Emails: m_hasann@encs.concordia.ca, weiping@ece.concordia.ca

Benoit Champagne
Electrical and Computer Engineering
McGill University, Montreal, Canada
Email: benoit.champagne@mcgill.ca

***Abstract*—Speech exhibits strong dependencies among its samples in both time and frequency domains. In this paper, we propose a low-complexity composite model for speech enhancement (SE) that integrates a convolutional neural network (CNN) and a long short-term memory (LSTM) network. These two modules take full advantage of the spectral and temporal information of input speech and extract in parallel a complementary set of features. The CNN is enabled to capture non-local spectral information via dilated frequency convolutions. It also incorporates an attention mechanism to recalibrate its weights without imposing considerable additional complexity. A grouping strategy is adopted for LSTM implementation to reduce its complexity while keeping performance almost unchanged. Our composite model is carefully designed to address concerns in real-time applications including limited computational resources, low-latency processing, and causal architecture. Through extensive and comparative simulation studies, it is shown that the proposed model significantly outperforms some other DNN-based SE methods in the recent literature.**

Index Terms — speech enhancement, dilated convolution, grouping strategy, attention technique, low complexity.

I. Introduction

In real-world environments, clean speech is often corrupted by ambient noise. Speech enhancement (SE) as a key area of speech processing focuses on removing the noise component from the noisy speech to improve user experience. SE is essential yet highly demanding for numerous applications such as mobile speech communication, hearing prosthesis, robust speech recognition, etc. [1].

In the last decade, most SE studies have been undertaken based on data-driven approaches that are often developed using deep learning. Deep learning can deal with highly complex acoustic scenarios and offer real-time processing. To shed a light on the necessity of causal SE, we can invoke many applications such as real-time speech communication and hearing aids, where in the latter case, even a very short latency as low as three to five milliseconds can be noticeable to a wearer [2].

Some of the most widely used deep learning-based SE methods have relied on the fully-connected deep neural network (DNN). Estimating an ideal ratio mask (IRM) by DNN for SE was one of the very first approaches of this type, which achieved notable improvement over traditional unsupervised SE methods [3]. DNN was also used to map noisy speech log power spectral magnitudes to the clean ones in [4] , which led to significant improvement in terms of speech quality and intelligibility. However, the DNN architecture involves a large number of model parameters. It also fails to model numerous speakers and noise types since it processes input samples independently and does not consider long-term temporal information of input speech [5].

Recently, LSTM networks have been introduced as a natural means to model temporal dependencies of speech. Chen *et al.* in [6] employed an LSTM network for SE and showed its advantage in speaker generalization over DNN. More recently, a time-frequency LSTM was introduced in [7] and [8] to take advantage of temporal and spatial information in the input speech spectrogram. Since the LSTM network has a large number of parameters, gated recurrent unit (GRU) and simple recurrent unit (SRU) networks were adopted in [9] and [10] as efficient implementations of LSTM for SE.

Besides, CNN has been also employed for complex spectrogram estimation for SE [11]. It was shown in [12] that the same SE results can be achieved by CNN with much smaller number of model parameters than DNN and LSTM, while the memory footprint for CNN is higher than DNN and LSTM, as will be discussed in Section III-C. Another limitation of CNN is the involved max pooling operation which only retains rough information [13]. Moreover, the receptive field of convolutional layers is limited, which means that only local correlations of input can be considered. Ouyang *et al.* in [14] applied a pooling layer-free network with dilated convolutional layers to tackle these CNN problems and showed promising SE results. Combinations of CNN and LSTM were also studied in [15]–[18] showing convincing SE results, but these methods are mainly non-causal and highly complex. In [19] and [20], an attention-driven CNN-BLSTM network and a time-frequency model were introduced, respectively, for artificial bandwidth extension, where the advantage of attention mechanism in SE is emphasized.

In this paper, we propose a low-complexity composite model for IRM estimation in which carefully designed LSTM and CNN are integrated to extract a complementary set of features by taking full advantage of the temporal and spectral dependencies of input speech. LSTM and CNN perform independently and in parallel to speed up the computation, thereby addressing fundamental concerns of real-time processing, limited latency and low complexity in SE applications. While having very low complexity, the proposed hybrid model outperforms some other DNN-based SE methods from the

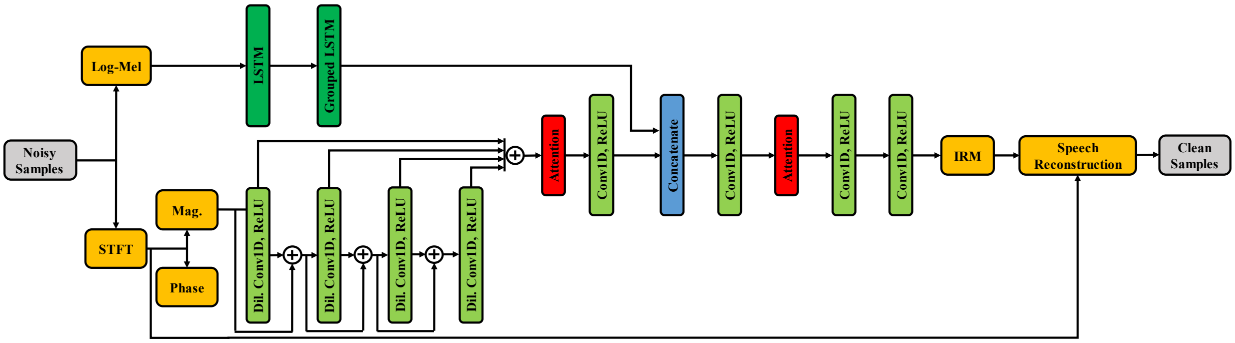


Fig. 1: Proposed composite model integrating grouped LSTM and attention-driven CNNs.

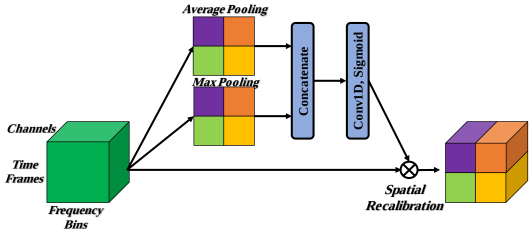


Fig. 2: Applied spatial SAE attention.

recent literature.

II. Proposed System Description

The proposed SE system is shown in Fig. 1, where CNN and LSTM networks operate in parallel to exploit spatial and temporal information of input speech. An attention mechanism is embedded in dilated frequency CNN to emphasize the valuable information of CNN feature maps while suppressing remaining details. A grouping strategy is adopted in LSTM implementation to reduce its complexity. The features extracted by both paths are mapped to IRM via another low-complexity CNN empowered with an attention mechanism.

A. Dilated 1D causal frequency convolution

The input of CNN is taken as the short-time Fourier transform (SFTT) of the input signal. CNN filters capture merely local information while the speech STFT exhibits non-local correlations along the frequency axis [21]. To benefit from such spectral correlations, Fu et al. [11] increased the size of CNN kernels, but this alternative leads to higher complexity and lower speed. To tackle this problem, dilated convolution has been introduced in [22], which exponentially expands the receptive field while keeping the kernel size small. The paradigm of Dilated convolution has been already employed in different contexts including SE [9], [14].

In our system in Fig. 1, we employ stacked dilated convolution CNNs in the lower path, to exploit the non-local spectral correlations of speech STFT. Further, 1D causal convolution along the frequency axis is applied. Skip connection and residual learning techniques are also adopted to facilitate training and accelerate convergence.

B. Attention Techniques

The spatial and channel squeeze and excitation (SAE) attention operations, as introduced in [23] and [24], respectively, recalibrate the feature maps in CNN through emphasizing the significant features and suppressing the rest. The spatial SAE squeezes the information of different channels pixel[1]-wise and excites spatially, while the channel SAE squeezes features spatially, via a global average pooling, into a channel descriptor and then excites along the channels. Various combinations of spatial and channel SAE have been actively studied in the literature [23].

Fig. 2 shows the spatial SAE employed in the CNNs of our model. Consider the input feature map $U \in \mathbb{R}^{T \times F \times C}$ where T, F, and C indicate the total number of time frames, frequency bins, and CNN channels, respectively. Slicing U over time-frequency bins gives a tensor $U = [u^{1,1}, u^{1,2}, ..., u^{t,f}, ..., u^{T,F}]$, where the dimension of $u^{t,f}$ is $1 \times 1 \times C$. In the squeezing phase, both average and max pooling operations over channels are applied for each $u^{t,f}$ to reflect valuable information of the feature map. These calculated matrices of dimension $T \times F$ are fed into a convolutional layer. Finally, the output of the convolutional layer is element-wise multiplied with the input feature map to re-weight its information pixel-wise. It is worth mentioning that the implementation of this operation does not entail a considerable number of parameters, unlike channel SAE.

C. Modeling Temporal Dependencies via Grouped LSTM

Speech signal exhibits strong temporal dependencies. LSTM with an embedded memory cell can appropriately leverage long-term context to boost SE performance. LSTM facilitates information flow over time using its internal gates. Considering $x^t \in \mathbb{R}^{M \times 1}$ and $h^{t-1} \in \mathbb{R}^{N \times 1}$ as the input and hidden state at times t and $t-1$, an LSTM is implemented using the following set of equations,

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

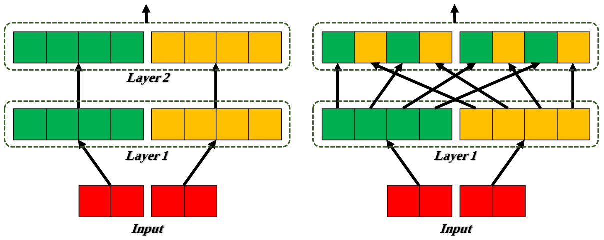
$$f^t = \sigma(W_f x^t + U_f h^{t-1} + b_f) \tag{2}$$

$$o^t = \sigma(W_o x^t + U_o h^{t-1} + b_o) \tag{3}$$

$$\hat{c}^t = \tanh(W_c x^t + U_c h^{t-1} + b_c) \tag{4}$$

$$c^t = f^t \odot c^{t-1} + i^t \odot \hat{c}^t \tag{5}$$

[1] An elementary time-frequency cell within the spectrogram



(a) (b)

Fig. 3: A two-layer grouped LSTM network: (a) Grouped with K=2; (a) Grouped with K=2 and a parameter-free representation rearrangement.

$$h^t = o^t \odot tanh(c^t) \tag{6}$$

where (i^t, f^t, and o^t) $\in \mathbb{R}^{N\times 1}$ denote input, forget, and output gates at time t, respectively. $W's \in \mathbb{R}^{N\times M}$, $U's \in \mathbb{R}^{N\times N}$, and $b's \in \mathbb{R}^{N\times 1}$ indicate weight and bias matrices. $\odot$ and σ denote element-wise multiplication and sigmoid activation function. Considering the above dimensions, the number of parameters to implement each gate is $N^2 + NM + N$.

In [25], a grouping strategy was proposed to reduce LSTM network complexity by splitting the LSTM input and middle layers into K independent groups. The grouping strategy for a two-layer LSTM network with K=2 is shown in Fig. 3 (a). Ignoring the bias vector, grouping strategy decreases the number of parameters by the factor of K as follows,

$$K\left((\frac{N}{K})^2 + (\frac{N}{K})(\frac{M}{K})\right) = \frac{N^2 + NM}{K} \tag{7}$$

However, the inter-group dependencies are lost due to the independent operation of each group. Hence, a parameter-free representation rearrangement was adopted, as shown in Fig 3 (b), to give subsequent layers access to the output of other groups. As such, the complexity of the LSTM network is reduced while keeping the performance almost at the same level.

D. Network Architecture

Referring to Fig. 1, the magnitude of noisy speech STFT is computed and then fed to the CNN. Four 1D causal CNNs with dilated frequency at the rate of 1, 2, 4, and 8 are stacked to exponentially enlarge the receptive field of CNN filters. The number of channels in these layers is 16, 32, 16, and 8, and the kernel size is (1×7). For residual layers, identity mapping with kernel size (1×1) is employed. The number of layers of skip connections is 32 and their kernel size is again (1×1). ReLU is adopted as the activation function. The skip outputs are then input to the attention layer which computes average and max pooling. The two matrices resulting from these operations are next combined using a convolutional layer with kernel size (1×7) and a sigmoid activation function. The output of the CNN path will be the point-wise product of CNN output and attention weights.

The input of the LSTM network consists of conventional acoustic features [18], namely, low-dimensional log melfilterbank energy features (Log-Mel) concatenated with their delta and acceleration. The grouped LSTM network is made up of two layers each comprising 128 units, wherein dropout at the rate of 0.3 is applied. Empirically, we found that the best results are achieved if a grouping strategy with K=2 is embedded just in the second layer only.

In the regression stage, the outputs of CNN and LSTM paths are concatenated and input to a 1D CNN network. The kernel size of this three-layer network is (1×3): the number of channels for the first two layers is 32 and 16, respectively, with ReLU activation function, and this number is 1 for the last layer, with a linear activation function A further attention mechanism is adopted between these two layers to further improve the SE results. The output of this network is an IRM mask, which will be multiplied with noisy speech STFT and combined by the noisy phase to reconstruct the clean speech.

III. Experiments

A. Experimental Setup

TIMIT dataset [26] is used in our experiment, from which 6300 utterances spoken by male and female are chosen for training. These utterances are mixed additively with random sections of 20 different noises [2] form NOISEX-92 [27] at SNR levels of -5, 0, 5, and 10 dB. For the testing stage, 60 unseen utterances spoken by male and female are mixed with two unseen noises, specifically, *Coffee Shop* (CS) and *Busy City Street* (BCS) from [28] at unmatched SNR levels of -6, 0, 6, and 12 dB. The sampling rate is set to 16kHZ and the STFT coefficients are computed by means of a 320-point DFT with Hanning window and %50 overlap. Perceptual evaluation of speech quality (PESQ) and segmental signal-to-noise ratio (SSNR) are used as performance metrics [29].

B. Comparison with Related Work

We compare the proposed model with three other DNN-based methods: DNN-cIRM [30], CNN-GRU MCRM [18], and CNN-RI [14]. DNN-cIRM and CNN-GRU MCRM are masking-based methods where a three-layer DNN and a combination of CNN and LSTM are respectively used to estimate a complex IRM. For CNN-RI, a fully convolutional neural network is employed to estimate the complex spectrogram of clean speech. All the networks are trained and tested with the same datasets, as described above, for a fair comparison.

Table I shows the PESQ and SSNR results obtained from each method along with the number of model parameters (in million). Top and bottom numbers in each cell of the table show the results for males and females, respectively. As shown, DNN-cIRM contains quite a large number of model parameters, CNN-GRU MCRM has a reduced number of model parameters, and CNN-RI shows a much lower model complexity. The results clearly show that the proposed delicately designed composite model achieves the best PESQ and SSNR results at all SNR levels with the lowest number

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

TABLE I: Evaluation of different methods for unseen utterances mixed with unseen noises at unmatched SNR levels.

Method	PESQ								SSNR								Number of Parameters (Million))
	-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	1.25	1.24	1.68	1.74	2.10	2.24	2.56	2.68	-10.77	-9.69	-5.61	-5.01	0.23	5.57	5.57	6.11	-
	0.96	0.93	1.34	1.48	1.87	2.00	2.37	2.51	-9.93	-8.95	-5.59	-5.15	0.19	5.67	5.67	6.21	
DNN-cIRM	1.51	1.98	2.14	2.48	2.72	2.97	3.21	3.38	-1.39	1.25	2.48	4.05	6.63	8.56	8.56	9.32	2.82 M
	1.24	1.68	1.83	2.21	2.40	2.65	2.94	3.12	-1.22	1.56	2.38	4.17	6.31	7.97	7.97	8.64	
CNN-GRU MCRM	1.66	1.99	2.20	2.52	2.73	2.94	3.26	3.42	-0.77	1.28	2.39	4.07	6.77	8.49	8.49	9.55	0.99 M
	1.40	1.70	1.98	2.24	2.51	2.69	2.98	3.14	-0.87	1.46	2.50	4.43	6.88	8.08	8.08	9.05	
CNN-RI	1.54	1.71	2.00	2.18	2.48	2.59	2.91	2.98	-5.36	-1.10	-0.41	2.27	4.96	7.53	7.53	8.00	0.24 M
	1.22	1.45	1.69	1.91	2.18	2.38	2.64	2.77	-4.99	-0.55	-0.02	2.69	5.56	7.94	7.94	8.64	
Proposed	**1.84**	**2.37**	**2.45**	**2.81**	**2.94**	**3.21**	**3.32**	**3.50**	**-0.26**	**3.72**	**4.62**	**6.46**	**8.22**	**8.82**	**8.82**	**9.32**	**0.21 M**
	1.75	**2.20**	**2.36**	**2.76**	**2.83**	**3.14**	**3.28**	**3.41**	**0.07**	**3.84**	**5.08**	**6.93**	**8.51**	**9.42**	**9.42**	**9.88**	

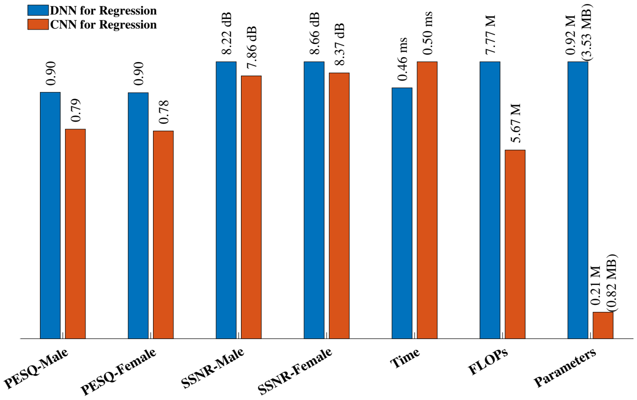


Fig. 4: Comparison of employing DNN or CNN for regression. PESQ and SSNR amounts indicate the average improvement values over all noises and SNR levels. The number of parameters is in million.

of model parameters in comparison with the aforementioned methods.

C. CNN vs. DNN for Regression

DNN and CNN have different attributes in terms of modeling high and low-frequency signals as demonstrated in [31], and their respective advantage depends on the application of interest. Apart from that, Park *et al.* in [12] showed that CNN and DNN can yield almost the same SE performance although CNN requires a much smaller number of model parameters. While they discussed the memory needed to store model parameters, they did not consider the required memory for computations.

In this section, we compare DNN and CNN when used for the regression part of our model, in terms of PESQ, SSNR, computation time, memory footprint, and the number of parameters. The DNN consists of two layers each having 512 nodes with a ReLU activation function, and one affine last layer comprising 161 nodes. The CNN structure is as described in Section II-D. A single NVIDIA GeForce RTX 2080 GPU with 8 GB memory and 2.2 GHz AMD Ryzen Threadripper 2920X 12-Core Processor is used to perform the experiments. The comparison results are shown in Fig. 4. The first four columns show the average PESQ and SSNR improvement over all noises and SNR levels. The next column presents the average processing time for a 1-second audio file. To measure computational complexity for the whole model, we use FLOPs (FLoating-point OPerations) per frame as in [32]. The last column shows the number of model parameters in million (M) and the memory to store them in megabyte (MB). It is worth mentioning that the computational time and the number of FLOPs are measured in the testing stage.

As shown in Fig. 4, using DNN for the final regression yields better results compared to CNN in terms of both PESQ and SSNR. Moreover, the computational time for the model with DNN for the regression is a bit less than that with CNN. However, the number of model parameters using DNN is roughly 4.4 times that of the same model using CNN for regression. Besides, the number of FLOPs for the whole model in the DNN case is almost 1.37 times more than that using CNN. The high number of FLOPs in the DNN case clearly stems from the high number of model parameters since the computations in DNN are simple and straight forward. For the CNN, nonetheless, the memory requirements originate not only from the need to store its model parameters, but also other intermediate activations as well as a workspace for the computations. Also, the high volume of matrix multiplications in CNN explains its relatively large computation time [32]. This explains why CNN and DNN take almost the same computational time and comparable memory while the number of model parameters using CNN is much less than that of a DNN. Consequently, the choice of CNN vs. DNN in our model depends on the application , e.g., whether the computation is to be carried out online or offline.

IV. Conclusion

In this paper, a low-complexity composite model has been proposed for speech enhancement. The new model benefits from an LSTM network that appropriately exploits the strong temporal dependencies of speech and a dilated frequency convolution 1D CNN that captures non-local spectral dependencies in speech spectrograms. The Skip connection and residual learning techniques were embedded in the CNN structure for easier training and faster convergence, while a grouping strategy was adopted to reduce the complexity of the LSTM. Furthermore, an attention technique was adopted in the CNN implementation to emphasize the prominent information. It was shown that the proposed model outperforms some DNN-base SE methods while having very low-complexity.

References

[1] P. C. Loizou, *Speech Enhancement: Theory and Practice*. CRC press, 2013.

[2] J. Agnew and J. M. Thornton, "Just noticeable and objectionable group delays in digital hearing aids," *J. of the American Academy of Audiology*, vol. 11, no. 6, pp. 330–336, 2000.

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

[4] 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. on Audio, Speech, and Language Processing*, vol. 23, no. 1, pp. 7–19, 2014.

[5] M. Kolbæk, Z.-H. Tan, and J. Jensen, "Speech intelligibility potential of general and specialized deep neural network based speech enhancement systems," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 1, pp. 153–167, 2016.

[6] J. Chen and D. Wang, "Long short-term memory for speaker generalization in supervised speech separation," *The J. of the Acoustical Society of America*, vol. 141, no. 6, pp. 4705–4714, 2017.

[7] J. Deng, B. Schuller, F. Eyben, D. Schuller, Z. Zhang, H. Francois, and E. Oh, "Exploiting time-frequency patterns with LSTM-RNNs for low-bitrate audio restoration," *Neural Computing and Applications*, vol. 32, no. 4, pp. 1095–1107, 2020.

[8] T. Grzywalski and S. Drgas, "Using recurrences in time and frequency within u-net architecture for speech enhancement," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, May 2019, pp. 6970–6974.

[9] M. Hasannezhad, Z. Ouyang, W.-P. Zhu, and B. Champagne, "An integrated CNN-GRU framework for complex ratio mask estimation in speech enhancement," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conf. (APSIPA ASC)*. IEEE, Dec. 2020, pp. 764–768.

[10] X. Cui, Z. Chen, and F. Yin, "Speech enhancement based on simple recurrent unit network," *Applied Acoustics*, vol. 157, p. 107019, 2020.

[11] 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 on Machine Learning for Signal Processing (MLSP)*, Sep. 2017, pp. 1–6.

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

[13] H. Zhang and J. Ma, "Hartley spectral pooling for deep learning," *arXiv preprint arXiv:1810.04028*, 2018.

[14] Z. Ouyang, H. Yu, W.-P. Zhu, and B. Champagne, "A fully convolutional neural network for complex spectrogram processing in speech enhancement," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, May 2019, pp. 5756–5760.

[15] K. Tan and D. Wang, "A convolutional recurrent neural network for real-time speech enhancement." in *INTERSPEECH*, Sep. 2018, pp. 3229–3233.

[16] M. Strake, B. Defraene, K. Fluyt, W. Tirry, and T. Fingscheidt, "Fully convolutional recurrent networks for speech enhancement," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, May 2020, pp. 6674–6678.

[17] H. Zhao, S. Zarar, I. Tashev, and C.-H. Lee, "Convolutional-recurrent neural networks for speech enhancement," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2018, pp. 2401–2405.

[18] M. Hasannezhad, Z. Ouyang, W.-P. Zhu, and B. Champagne, "Speech separation using a composite model for complex mask estimation," in *IEEE 63rd Int. Midwest Symposium on Circuits and Systems (MWSCAS)*, Aug. 2020, pp. 578–581.

[19] M. Ge, L. Wang, N. Li, H. Shi, J. Dang, and X. Li, "Environment-dependent attention-driven recurrent convolutional neural network for robust speech enhancement." in *INTERSPEECH*, Sep. 2019, pp. 3153–3157.

[20] Y. Dong, Y. Li, X. Li, S. Xu, D. Wang, Z. Zhang, and S. Xiong, "A time-frequency network with channel attention and non-local modules for artificial bandwidth extension," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, May 2020, pp. 6954–6958.

[21] D. Yin, C. Luo, Z. Xiong, and W. Zeng, "Phasen: A phase-and-harmonics-aware speech enhancement network." in *AAAI*, 2020, pp. 9458–9465.

[22] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.

[23] A. G. Roy, N. Navab, and C. Wachinger, "Concurrent spatial and channel 'squeeze & excitation' in fully convolutional networks," in *Int. Conf. on medical image computing and computer-assisted intervention*. Springer, 2018, pp. 421–429.

[24] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2018, pp. 7132–7141.

[25] F. Gao, L. Wu, L. Zhao, T. Qin, X. Cheng, and T.-Y. Liu, "Efficient sequence learning with group recurrent networks," in *Proc. of the 2018 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1*, 2018, pp. 799–808.

[26] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett, "Getting started with the DARPA timit cd-rom: An acoustic phonetic continuous speech database," *National Institute of Standards and Technology (NIST), Gaithersburgh, MD*, vol. 107, p. 16, 1988.

[27] 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 Communication*, vol. 12, no. 3, pp. 247–251, 1993.

[28] "Premium beat," www.premiumbeat.com.

[29] Y. Hu and P. C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 16, no. 1, pp. 229–238, 2007.

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

[31] S.-W. Fu, Y. Tsao, X. Lu, and H. Kawai, "Raw waveform-based speech enhancement by fully convolutional networks," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conf. (APSIPA ASC)*. IEEE, Dec. 2017, pp. 006–012.

[32] Z. Lu, S. Rallapalli, K. Chan, and T. La Porta, "Modeling the resource requirements of convolutional neural networks on mobile devices," in *Proc. of the 25th ACM Int. Conf. on Multimedia*, Oct. 2017, pp. 1663–1671.