A COMPLEX SPECTRAL MAPPING WITH INPLACE CONVOLUTION RECURRENT NEURAL NETWORKS FOR ACOUSTIC ECHO CANCELLATION

Chenggang Zhang, Jinjiang Liu, Xueliang Zhang

Department of Computer Science, Inner Mongolia University, China

cgzhang@mail.imu.edu.cn, jetliu1994@foxmail.com, cszxl@imu.edu.cn

ABSTRACT

Recently, deep learning is introduced in acoustic echo cancellation (AEC) and achieves remarkable performance. For deep learning-based AEC, the most important problem is generalization ability in diversity scenarios. Different from most methods which process the entire frequency band, we propose inplace convolution recurrent neural networks (ICRN) for end-to-end AEC, which utilizes inplace convolution and channel-wise temporal modeling to ensure the near-end signal information being preserved. In addition, we employ complex spectral mapping with a multi-task learning strategy for better generalization capability. Experiments conducted on various unmatched scenarios show that the proposed method outperforms previous methods. Moreover, the system has 210K parameters and 1.76G MACs, which is suitable for real-time applications.

Index Terms— Acoustic echo cancellation, inplace convolutional recurrent network, deep learning

1. INTRODUCTION

Due to acoustic coupling between the loudspeaker and the microphone of hands-free voice communication systems, the far-end talker receives its own voice that disturbs normal communication. Acoustic echo cancellation (AEC) is a technique to solve this problem [1, 2, 3]. Adaptive filtering is the main approach for AEC problem, which attempts to model the echo path from far-end (reference) signal to microphone. [4, 5]. However, the performance is still deteriorated in some cases, such as the double-talk problem and the nonlinear distortion introduced by power amplifier and loudspeaker.

Recently, deep learning methods without any prior assumptions of the signal model have been introduced to solve the AEC problem [6, 7, 8, 9]. Zhang and Wang [6] proposed a deep bidirectional long short-term memory (BiLSTM) to predict an ideal ratio mask from combined magnitude spectra of microphone and far-end signals, in order to recover the near-end signal. Zhang et al. [7] used convolutional recurrent networks (CRN) and LSTM to separate the near-end speech from the microphone signal. Fazel et al. [8] proposed deep recurrent neural networks with multi-task learning to estimate the near-end speech. More recently, the authors [10, 11] proposed hybrid methods that integrate adaptive filtering and deep learning for canceling the linear and nonlinear echoes, respectively.

Although showing impressive echo suppression performance, the learning-based method often degrades significantly in totally different conditions other than the ones they were trained on [8]. These methods have two shortcomings. 1) They take entire frequency band as the networks input for feature extraction, which ignore the independence of each individual frequency bin. 2) The hidden features are extracted by downsampling operation along the frequency dimension, e.g. convolution with stride larger than one, which potentially lead to the target information lost.

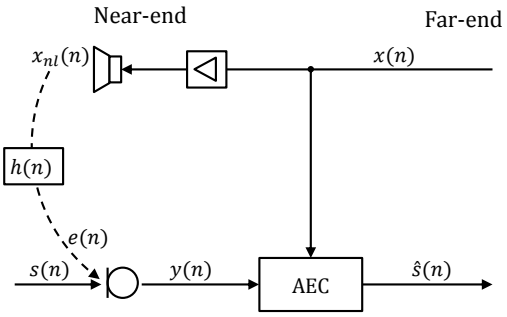


Fig. 1. Diagram of the single channel AEC.

In contrast, conventional method for AEC, e.g. recursive least squares algorithm (RLS) [4], treats each frequency band independently. This motivates us to design a novel algorithm to improve near-end signal quality and generation capability. In our recent work [12], we proposed the inplace convolution for frequency pattern extraction and reconstruction for multi-channel speech enhancement, which effectively improve speech quality. In this paper, we propose inplace convolution recurrent neural networks (ICRN) for AEC, which utilizes inplace convolution and channel-wise temporal modeling to make sure that information of the near-end signal in each frequency can be preserved efficiently. In addition, we also employ the complex spectral mapping with a multi-task learning strategy for better generalization.

The rest of this paper is organized as follows. In Section 2, the problem is formulated. In Section 3, we describe the proposed method. The experimental setup is shown in Section 4. In Section 5, we demonstrate performance of the proposed system. We conclude the paper in Section 6.

2. PROBLEM FORMULATION

The diagram of single channel AEC system is illustrated in Fig.1. The microphone signal $y(n)$ is a mixture of echo signal $e(n)$ and near-end signal $s(n)$, as follows:

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

The speaker out signal $x_{nl}(n)$ is a nonlinear distortion of the far-end signal $x(n)$. The acoustic echo $e(n)$ is $x_{nl}(n)$ convolving with a room impulse response (RIR) [13] $h(n)$. It should be mentioned that we focus on the main challenges of AEC, e.g. the double-talk and nonlinearity. Other issues like background noise are not covered.

3. PROPOSED METHOD

3.1. Inplace convolution

Different from the classic CRN [14, 7], which downsample the features along the frequency dimension, we proposed *inplace convolution* where the stride of convolution is set to one that maintains the number of features unchanged [12]. This simple yet effective operation not only retains key information for each target frequency bin but also keeps the correlation of adjacent channels. For the AEC task, feature extraction by inplace convolution potentially preserve the near-end signal information optimally.

3.2. Encoder and decoders

As shown in Fig.2, the model has one encoder and two same structure decoders (denote as *amplitude decoder* and *phase decoder*). The encoder and decoders consist of 6 cascaded Inplace convolution and deconvolution layers, respectively. Skip connections are used to concatenate the output of each Inplace-Conv2d block to the input of the corresponding Inplace-Deconv2d block. Each Inplace-Conv2d and Inplace-Deconv2d block is successively followed by a batch normalization [15] operation and an exponential linear unit (ELU) [16] activation function. Two linear layers are parallel at the end of each decoder to project the learned features.

3.3. Channel-wise temporal features modeling

After inplace convolution encoding, the extracted frequency bin only contain frequency dimension features without channel-wise features. However, it is crucial to exploit temporal correlations between different channels for each frequency bin, so we employ two LSTM layers to model the temporal characteristics of each frequency bin along the channel dimension. Since the context information are similar among different frequency bins, we share the same LSTM blocks. This channel-wise temporal features modeling allows the framework have lower parameters at the same time. A channel-wise linear layer is followed to keep the shape of LSTM output the same as input.

A more detailed description of our proposed network hyperparameters is provided in Table 2. The input size and the output size of encoder (decoder) layer are given in the $[Batchsize \times frequencyChannels \times Featuremaps \times Timesteps]$ format. In addition, the layer hyperparameters are specified in the ($kernelsize$, $strides$, $outchannels$) format. Note that the number of frequency channels in each decoder layer is doubled by skip connections.

3.4. Multi-task learning strategy (MTLS)

Inspired by multi-task learning [17, 18], in which multiple related tasks are jointly learned can achieve a regularization effect for improving the generalization, we adopt two decoders to predict the phase Ψ, masking M and mapping $|\hat{S}|_{map}$ of clean near-end speech, respectively. As depicted in Fig 2. M and $|\hat{S}|_{map}$ are two outputs of

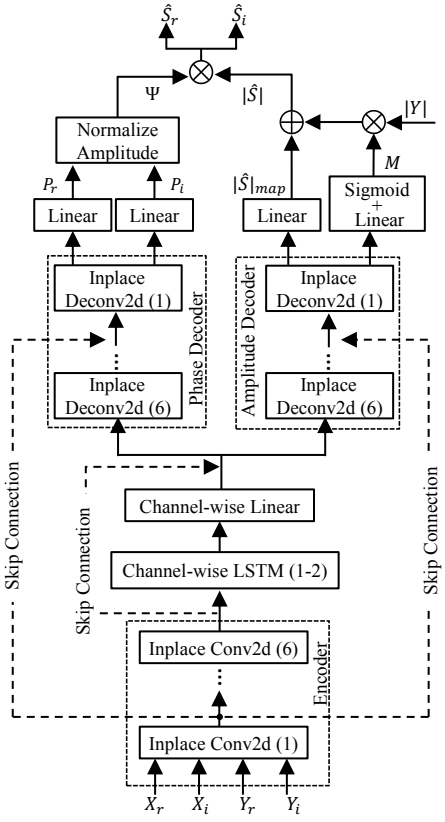


Fig. 2. Network architecture of the proposed ICRN. Inplace-Conv2d and Inplace-Deconv2d denote the inplace convolution and deconvolution operations, respectively. The numbers in parenthesis represent the i-th layer. Ψ denotes the phase, and $|\cdot|$ denotes the amplitude. $(\cdot)_r$ and $(\cdot)_i$ are the real and imaginary spectra, respectively.

amplitude decoder. We generate the amplitude spectrum as follows:

$$|\hat{S}| = |Y| \otimes M + |\hat{S}|_{map} \tag{2}$$

where $\otimes$ denotes element-wise multiplication. $|Y|$ is the amplitude of $y(n)$.

Meanwhile, the estimate phase Ψ is obtained by normalizing the outputs of phase decoder, seen Eq.(3).

$$\Psi = \frac{P_r + jP_i}{\sqrt{{P_r}^2 + {P_i}^2}} \tag{3}$$

where P_i and P_r are two outputs of phase decoder.

Accordingly, the predicted complex spectrogram $\hat{S}$ can be computed by $\hat{S} = |\hat{S}| \otimes \Psi$. In the Cartesian coordinates:

$$\hat{S} = \hat{S}_r + j\hat{S}_i \tag{4}$$

We utilize the real spectrum S_r and imaginary spectrum S_i of the near-end signal as the training target. For inference stage, the inverse STFT is used to obtain the estimated waveform of near-end signal $\hat{s}(n)$. For complex spectral mapping, recent studies [19, 20] demonstrate that combine complex spectrum domain loss with a magnitude domain loss could improve the speech quality, so we use the follow-

Table 1. Comparisons of different methods with real RIR, unseen nonlinearity distortion and music echo scenarios. The best scores in each case are highlighted by **boldface**.

	Real RIR						Unseen Nonlinear Distortion						Music echo					
Metrics	ERLE			PESQ			ERLE			PESQ			ERLE			PESQ		
SER(dB)	-5	0	5	-5	0	5	-5	0	5	-5	0	5	-5	0	5	-5	0	5
WebRTC-AEC3	30.21	30.16	29.04	1.47	1.60	1.74	22.35	22.49	21.89	1.28	1.48	1.68	24.25	21.56	19.09	1.85	2.14	2.34
BLSTM [6]	**56.89**	**58.03**	58.52	1.94	2.27	2.61	24.24	27.09	30.95	1.91	2.26	2.60	43.89	42.21	40.78	1.96	2.31	2.65
CRN [7]	35.33	35.20	35.17	2.41	2.80	3.15	26.42	26.47	26.79	2.10	2.53	2.90	28.34	26.22	22.95	2.24	2.57	2.83
CRN+MTLS	51.17	52.35	53.67	2.48	2.91	3.26	29.44	28.59	28.17	2.14	2.57	2.95	44.04	41.96	40.54	2.33	2.65	2.89
ICRN-MTLS	34.53	35.03	35.42	2.54	2.93	3.27	26.35	26.36	26.81	2.12	2.54	2.92	33.04	30.70	27.24	2.64	2.92	3.14
ICRN	54.56	56.12	**58.95**	**2.67**	**3.10**	**3.47**	**32.36**	**32.17**	**32.71**	**2.21**	**2.58**	**2.98**	**47.95**	**45.74**	**41.90**	**2.68**	**2.97**	**3.15**

Table 2. The proposed ICRN architecture, where B, T and F represents the number of batch size, frame and frequency bin, respectively.

Layer name	Input size	Hyperparameters	Output size
Inplace Conv2d(1)	[B, 4, F, T]	5×1, (1, 1), 24	[B, 24, F, T]
Inplace Conv2d(2-6)	[B, 24, F, T]	5×1, (1, 1), 24	[B, 24, F, T]
Reshape	[B, 24, F, T]	-	[B×F, T, 24]
Channel-wise LSTM×2	[B×F, T, 24]	48	[B×F, T, 48]
Channel-wise Linear	[B×F, T, 48]	24	[B×F, T, 24]
Reshape	[B×F, T, 24]	-	[B, 24, F, T]
Inplace Deconv2d(6-2) ×2	[B, 48, F, T]	5×1, (1, 1), 24	[B, 24, F, T]
Inplace Deconv2d(1) ×2	[B, 48, F, T]	5×1, (1, 1), 24	[B, 2, F, T]
Reshape	[B, 2, F, T]	-	[B, 2, T, F]
Linear × 4	[B, 1, T, F]	-	[B, 1, T, F]

Table 3. Types of nonlinear simulation function $f(\cdot)$ for generating training mixtures.

Type	Nonlinear Functions	
I	$x_n(k) = ax(k)/\sqrt{a^2 + x^2(k)}, a = 5/\varepsilon$	$\varepsilon \in [2, 5]$
II	$x_n(k) = 1 - e^{-ax(k)}, a = \varepsilon/10$	
III	$x_n(k) = 2ax(k) + ax^2(k) + x^3(k)$, $a = \log(\varepsilon/10) + 0.1$	

ing loss function.

$$\mathcal{L} = \|\hat{S}_r - S_r\|_1 + \|\hat{S}_i - S_i\|_1 + \|\sqrt{\hat{S}_r^2 + \hat{S}_i^2} - |S|\|_1 \quad (5)$$

where $\|\cdot\|_1$ is the L_1 norm, $|S|$ denotes the magnitude of $s(n)$.

4. EXPERIMENTAL SETUPS

4.1. Datasets preparation

We randomly choose 200 pairs of speakers from the 630 speakers in the TIMIT dataset [21] as the near-end and far-end speakers, respectively. In each far-end speaker, three utterances are randomly selected and concatenated into a long utterance. In each near-end speaker, the utterance is kept the same length as the one in the far-end by adding zeros at both the front and rear. To model the nonlinearity of the echo path, the far-end signal $x(n)$ is further distorted by three types of nonlinear function, as shown in Table 3. After that, the distorted signal $x_{nl}(n)$ is convolved with RIR generated using the Image method [22]. We simulate different rooms of size $l \times w \times h$ m^3 for training mixtures, where l form 4 to 10, w from 5 to 11 with step 1m, and h = [3, 4]m. The microphone-loudspeaker (M-L) distance is randomly selected from [0.5, 0.7, 0.9]m. The reverberation time (T_{60}) is randomly selected from [0.2, 0.3, 0.4, 0.5, 0.6]s to generate RIRs in each room. 10000 RIRs are created for training mixtures. Then $e(n)$ is mixed with near-end speech $s(n)$ at a signal-to-echo ratio (SER) randomly chosen from [-9, -6, -3, 0, 3, 6, 9]dB. The SER level here is evaluated in the double-talk period. Finally, 32000 mixtures are generated, totaling about 71 hours, in which 30000 for training and 2000 for validation. We also randomly selected 100 pairs of speakers from the remaining 230 untrained speakers to generate 100 testing mixtures in the same way.

4.2. Baselines and training details

We compare our proposed approach with three baselines including both conventional and NN-based methods that represent the recent advances in AEC. 1) WebRTC-AEC3[1] 2) BLSTM [6] and 3) CRN [7]. Note that the CRN model without near-end speech detector performs better in terms of PESQ as described in [7]. ICRN is the proposed method. We also evaluate the effectiveness of the proposed MTLS. CRN+MTLS denotes the CRN with MTLS. ICRN-MTLS is the proposed network without MTLS, and its training target is the complex spectrum which is the same as CRN [7].

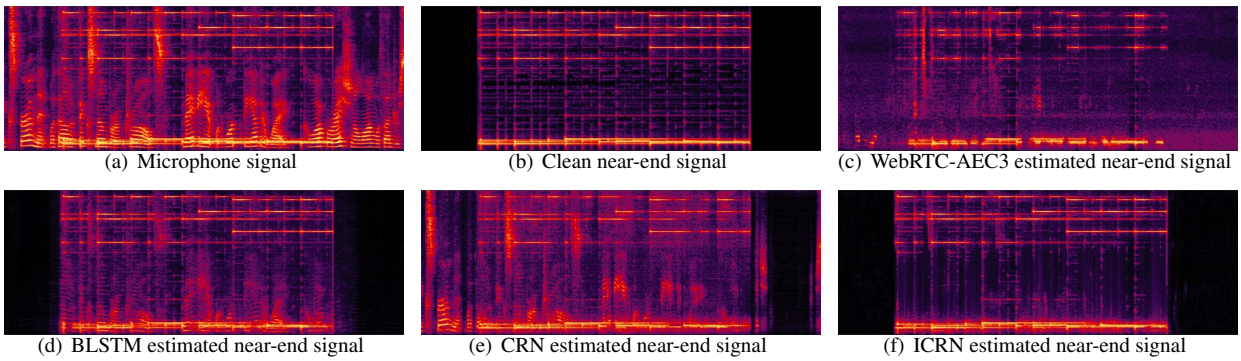
The proposed approach is optimized by the Adam algorithm [23]. The initial learning rate is 0.001, and the model is trained for 100 epochs with the batch size of 8. We reduce 70% learning rate when validation loss does not increase for consecutive 3 epochs. All signals are sampled at 16 kHz. A 20ms Hamming window is used with 10ms frame shift.

5. EXPERIMENTAL RESULTS AND DISCUSSION

To evaluate the performance of the proposed method, we utilize the echo return loss enhancement (ERLE) [3] and the perceptual evaluation of speech quality (PESQ) [24] as the metrics that measure the echo suppression for single-talk and near-end speech quality for double-talk periods, respectively. Higher scores indicate better performance.

We first evaluate the performance in three common unmatched conditions under the same simulation training dataset. 1) Real RIR scenario selected from the Aachen Impulse Response database which are measured in meeting rooms [25]. 2) Unseen nonlinear distortion scenario which utilizes the $hardclipping$ and memoryless

[1]https://github.com/ewan-xu/AEC3



(a) Microphone signal (b) Clean near-end signal (c) WebRTC-AEC3 estimated near-end signal

(d) BLSTM estimated near-end signal (e) CRN estimated near-end signal (f) ICRN estimated near-end signal

Fig. 3. Sample spectrograms of different signals at SER = 0 dB. Note that the near-end is music signal while the far-end is speech signal.

sigmoid functions followed by [7, 8, 10] to simulate the hardware distortions. 3) Music echo scenario where far-end signal is music selected from MUSAN database [26].

Table 1 shows the average results of all methods in terms of ERLE and PESQ at different SERs. From the table, we conclude the following points: 1) Because of radical nonlinearity processing, the WebRTC-AEC3 achieves lower sores in terms of PESQ. 2) Although the BLSTM achieves impressive ERLE scores, it is a non-causal system that utilizes future frames. 3) The CRN+MTLS yields better performance than CRN, and ICRN is better than ICRN-MTLS in all conditions. It implies that the MTLS significantly improve the system's ability. 4) The ICRN-MTLS (without MTLS) significantly outperforms CRN in terms of PESQ, it demonstrates that the near-end information is greatly preserved because of the frequency independence and avoiding downsampling operations. 5) The proposed ICRN consistently outperforms the other methods in terms of PESQ, while slightly worse compare with BLSTM in terms of ERLE at -5 and 0 dB SERs for real RIR conditions.

Next, we compare the methods generalization performance in a more challenging scenario in which the far-end is speech while the near-end is music signal. The average results of all methods are shown in Table 4. Note that the signal-to-distortion ratio (SDR) is calculated since the PESQ hardly evaluates the near-end music signal quality during the double-talk period. SDR is defined as:

$$SDR = 10\log_{10}\frac{\|s\|_2^2}{\|s-\hat{s}\|_2^2} \tag{6}$$

where s is the clean music signal, $\hat{s}$ is the estimated music signal, $\|\cdot\|_2$ denotes the L_2 norm.

As shown in Table 4, the proposed ICRN outperforms other methods in all conditions, and the inplace convolution is proved effective. For example, the near-end music signal estimated by the ICRN-MTLS improves SDR by 5.81 dB over CRN, and the ICRN inproves SDR by 6.58 dB over CRN+MTLS at 0 dB SER. A clearer spectrograms of different signals are exhibited in Fig. 3. It is observed that the ICRA can suppress more acoustic echoes in the scenario where the near-end signal is extremely unmatched. More audio demos could be found at here[3].

Finally, we compare the number of parameters and computation of the learning-based methods. From Table 5, we can find that the number of trainable parameters of the ICRN is 0.21 million which is 43 times smaller than CRN [7], and the number of multiply-accumulate operations (MACs) is 1.76G per second which is only increased by 11%. In addition, our algorithm is a casual system, and we run our Pytorch code on a laptop computer with Intel(R) Core(TM) i5-6200U CPU @2.3 GHz, the computational time of our model to process a frame is 3.95 ms, which is much less than the stride time 10 ms. The real time factor (RTF) is 0.395 which suitable for real-time system.

[3]https://chenggangzhang.github.io/ICRN-AEC

Table 4. Comparisons of different methods with near-end is music scenario.

	Far-end(speech), Near-end(music)					
Metrics	ERLE			SDR		
SER (dB)	-5	0	5	-5	0	5
WebRTC-AEC3	16.27	19.22	20.69	-2.45	-1.92	-1.99
BLSTM [6]	44.95	41.97	38.80	3.93	6.39	9.43
CRN [7]	28.03	31.55	33.08	2.65	4.54	7.10
CRN+MTLS	47.38	46.61	42.52	2.70	4.64	7.17
ICRN-MTLS	29.40	33.49	34.72	6.74	10.35	13.64
ICRN	**48.85**	**47.25**	**43.28**	**8.05**	**11.22**	**13.73**

Table 5. Comparisons of learning-based methods of trainable parameters (unit is Million) and MACs (unit is Giga).

Method	Parameters (M)	MACs (G)
BLSTM [6]	8.13	0.81
CRN [7]	9.07	1.60
ICRN	0.21	1.76

6. CONCLUSIONS

In this study, we propose ICRN for end-to-end AEC, which utilizes inplace convolution and channel-wise temporal modeling to insure the key information of the target can be preserved. In addition, we employ a multi-task learning strategy for complex spectral mapping. Experimental results show that the proposed method has good generalization performance.

7. ACKNOWLEDGMENT

This research was supported by the National Natural Science Foundation of China (No.61876214).

8. REFERENCES

[1] J. Benesty, T. Gänsler, D.R. Morgan, M.M. Sondhi, and S.L. Gay, *Advances in Network and Acoustic Echo Cancellation*, Springer Science & Business Media, 2001.

[2] E. Hänsler and G. Schmidt, *Acoustic echo and noise control: a practical approach*, vol. 40, John Wiley & Sons, 2005.

[3] G. Enzner, H. Buchner, A. Favrot, and F. Kuech, "Acoustic echo control," in *Academic Press Library in Signal Processing*, vol. 4, pp. 807–877. Elsevier, 2014.

[4] S.S. Haykin, *Adaptive filter theory*, Pearson Education India, 2005.

[5] C. Paleologu, S. Ciochină, J. Benesty, and S.L. Grant, "An overview on optimized nlms algorithms for acoustic echo cancellation," *EURASIP Journal on Advances in Signal Processing*, vol. 2015, no. 1, pp. 97, 2015.

[6] H. Zhang and D.L. Wang, "Deep Learning for Acoustic Echo Cancellation in Noisy and Double-Talk Scenarios," in *Proc. Interspeech 2018*, 2018, pp. 3239–3243.

[7] H. Zhang, K. Tan, and D.L. Wang, "Deep Learning for Joint Acoustic Echo and Noise Cancellation with Nonlinear Distortions," in *Proc. Interspeech 2019*, 2019, pp. 4255–4259.

[8] A. Fazel, M. El-Khamy, and J. Lee, "CAD-AEC: Context-aware deep acoustic echo cancellation," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6919–6923.

[9] J. Franzen, E. Seidel, and T. Fingscheidt, "AEC in A Netshell: on target and topology choices for fcrn acoustic echo cancellation," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 156–160.

[10] C. Zhang and X. Zhang, "A Robust and Cascaded Acoustic Echo Cancellation Based on Deep Learning," in *Proc. Interspeech 2020*, 2020, pp. 3940–3944.

[11] M.M. Halimeh, T. Haubner, A. Briegleb, A. Schmidt, and W. Kellermann, "Combining adaptive filtering and complex-valued deep postfiltering for acoustic echo cancellation," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 121–125.

[12] J. Liu and X. Zhang, "Inplace Gated Convolutional Recurrent Neural Network for Dual-Channel Speech Enhancement," in *Proc. Interspeech 2021*, 2021, pp. 1852–1856.

[13] E.AP. Habets, "Room impulse response generator," *Technische Universiteit Eindhoven, Tech. Rep*, vol. 2, no. 2.4, pp. 1, 2006.

[14] K. Tan and D.L. Wang, "Complex spectral mapping with a convolutional recurrent network for monaural speech enhancement," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6865–6869.

[15] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*. PMLR, 2015, pp. 448–456.

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

[17] Y. Zhang and Q. Yang, "An overview of multi-task learning," *National Science Review*, vol. 5, no. 1, pp. 30–43, 2018.

[18] A. Kumar and H. Daume III, "Learning task grouping and overlap in multi-task learning," *arXiv preprint arXiv:1206.6417*, 2012.

[19] S.W. Fu, T.Y. Hu, Y. Tsao, and X. Lu, "Complex spectrogram enhancement by convolutional neural network with multi-metrics learning," in *2017 IEEE 27th International Workshop on Machine Learning for Signal Processing (MLSP)*, 2017, pp. 1–6.

[20] Z.Q. Wang and D.L. Wang, "Deep learning based target cancellation for speech dereverberation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 941–950, 2020.

[21] L.F. Lamel, R.H. Kassel, and S. Seneff, "Speech database development: Design and analysis of the acoustic-phonetic corpus," in *Speech Input/Output Assessment and Speech Databases*, 1989.

[22] J.B. Allen and D.A. Berkley, "Image method for efficiently simulating small-room acoustics," *The Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943–950, 1979.

[23] D.P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[24] A.W. Rix, J.G. Beerends, M.P. Hollier, and A.P. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing*. IEEE, 2001, vol. 2, pp. 749–752.

[25] M. Jeub, M. Schafer, and P. Vary, "A binaural room impulse response database for the evaluation of dereverberation algorithms," in *International Conference on Digital Signal Processing*. IEEE, 2009, pp. 1–5.

[26] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.