Speaker-Aware Speech Enhancement with Self-Attention

Ju Lin
Clemson University
Clemson, SC 29634
Email: jul@clemson.edu

Adriaan J. van Wijngaarden
Nokia Bell Labs
Murray Hill, NJ 07974
Email: avw@ieee.org

Melissa C. Smith
Clemson University
Clemson, SC 29634
Email: smithmc@clemson.edu

Kuang-Ching Wang
Clemson University
Clemson, SC 29634
Email: kwang@clemson.edu

***Abstract*—Speech enhancement aims to improve the intelligibility and quality of speech that is affected by noise. In this paper, we propose a novel speaker-aware speech enhancement (SASE) method that extracts speaker information using long short-term memory (LSTM) layers, and then uses a convolutional recurrent neural network (CRN) to embed the extracted speaker information. It is shown in a series of comprehensive experiments that only a few seconds of reference audio suffice for the proposed SASE method to perform better than LSTM and CRN baseline systems. The addition of a self-attention mechanism can further boost relevant speech-quality metrics.**

I. Introduction

Speech enhancement aims to improve the quality and the intelligibility of a speech signal that is degraded by ambient noise. Speech enhancement algorithms are used extensively in many audio- and communication systems, including mobile handsets, speaker verification systems and hearing aids. Speech enhancement methods have been developed and refined during the last several decades. Popular classic techniques include spectral-subtraction algorithms, statistical model-based methods that use maximum-likelihood (ML) estimators, Bayesian estimators, minimum mean squared error (MMSE) methods, subspace algorithms based on single value decomposition and noise-estimation algorithms (see [1] and references therein). Modern speech enhancement techniques often use deep learning, which typically outperform classic methods. Early methods include a recurrent neural network (RNN) that was used to model long-term acoustic characteristics [2], a deep auto-encoder to denoise the signal using greedy layer-oriented pre-training [3], and a deep neural (DNN) that was used as a non-linear regression function [4]. In [5], a convolutional recurrent neural network (CRN) was used, consisting of a convolutional encoder-decoder architecture and multiple long short-term memory (LSTM) layers. Generative adversarial networks (GANs), which are known for their ability to generate natural-looking signals in the time or frequency domain have also been applied successfully for speech enhancement [6]–[10]. Recent studies [11]–[17] consider the use of an attention mechanism for speech enhancement. In [15], self-attention [18] is combined with a dense convolutional neural network. A time-frequency (T-F) attention method, proposed in [16], combines time- and frequency-based attention for noisy reverberant speech enhancement.

Recently, modeling to learn the acoustic noisy-clean speech mapping has been enhanced by including auxiliary information such as visual cues [19], phonetic and linguistic information [20], [21], and speaker information [22]. In particular, the utilization of three kinds of broad phonetic class (BPC) information for speech enhancement can achieve notable improvements [21]. In [22], a speaker-aware deep denoising auto-encoder (SaDAE) extracts speaker representation from the noisy input using a DNN model. Target speaker extraction was investigated in [23]–[26].

In this paper, we first visualize the impact of the quality of a clean speech reference signal on speaker representation. Given that it is generally possible to collect a few seconds of clean reference speech in applications, e.g., similar to a smart virtual assistant that needs a few-second clean speech record during its setup stage, or extracted from (prior) high-SNR recordings, it is worthwhile investigating how a few seconds of clean reference can be best used to improve speech enhancement performance. The paper proposes a novel speaker-aware speech enhancement (SASE) method that extracts speaker information from a clean reference using long short-term memory (LSTM) layers, and then uses a convolutional recurrent neural network (CRN) to embed the extracted speaker information. The SASE framework is extended with a self-attention mechanism. Extensive simulations are performed using the Valentini-Botinhao corpus [27] to determine the performance of the proposed SASE method. It will be shown that a few seconds of clean reference speech is sufficient, and that the proposed SASE method performs well for a wide range of scenarios.

II. Speaker Embedding

The need for accurate speaker information is visualized by an experiment with fifteen speakers from the Valentini-Botinhao corpus [27], where two noise sources from the DEMAND corpus were added at an SNR of -5 dB, 0 dB, and 5 dB. Fig. 1 shows the t-distributed stochastic neighbor embedding (t-SNE) [28] of the speaker embedding information affected by noise. One clearly sees that speaker embedding information is very sensitive to noise. To mitigate the effects of noise, we propose to use clean reference speech, and show that a few seconds suffice to properly extract speaker embedding information. Given that it is often feasible to use a

few seconds of clean reference speech in real applications, e.g., from pre-recorded training samples or from prior high SNR recordings, it is worth investigating how the availability of a few seconds of clean reference can be best used to improve speech enhancement performance.

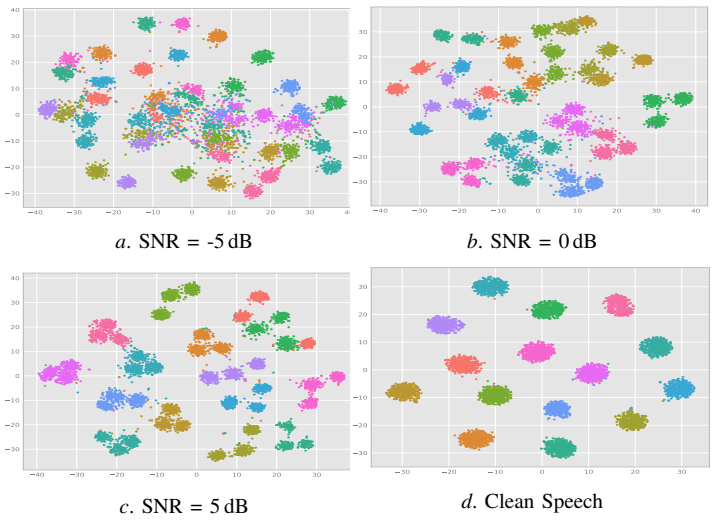

Fig. 1. Example of t-SNE visualization for speaker embedding of 15 speakers for various SNR conditions using two noise types from the DEMAND corpus [29].

III. Proposed SASE System

We propose a novel speaker-aware speech enhancement (SASE) system that uses a short clean-speech reference. The system consists of three components: a pre-trained speaker embedding extractor to process the reference clean speech, a CRN-based speech enhancement module, and a self-attention module. The CRN comprises a convolutional encoder-decoder structure which extracts high-level features with a 2-D convolution, and a long short-term memory (LSTM) layers to capture long-span dependencies in temporal sequences. A block diagram is shown in Fig. 2.

A. Speaker embedding extractor

The speaker embedding extractor, proposed in [30], is shown to perform well and is used here. It consists of three LSTM layers with 768 nodes in each layer and one linear layer with a 256-dimensional output.

The pre-trained model [1] is trained using the VoxCeleb2 data set [31], which comprises records of thousands of speakers. The model takes as input a Mel-spectrogram, which is extracted using a Short Time Fourier Transform (STFT) with an 80 ms window and a 40 ms hop size. The model achieves a 7.4 % equal error rate on the VoxCeleb1 test data set (first eight speakers of the data set).

[1] https://github.com/mindslab-ai/voicefilter

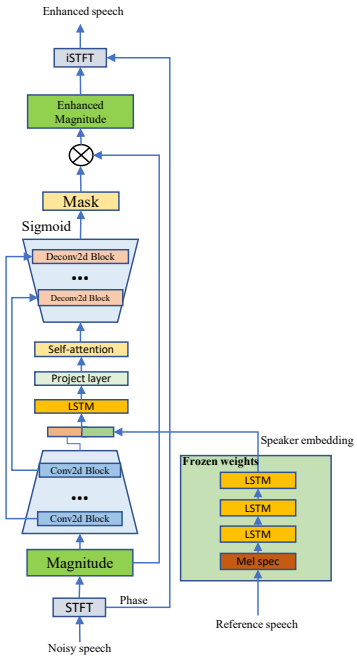

Fig. 2. Proposed SASE framework.

B. Self-Attention Module

Self-attention [18] is an efficient context information aggregation mechanism that operates on the input sequence itself and that can be utilized for any task that has a sequential input and output. Consider an 4-dimensional input $\mathbf{X}$ of shape $[B, C, T, F]$, where B, C, T, and F denote the batch size, number of channels, and the time and frequency dimensions, respectively. The self-attention layer takes $\mathbf{X}$ as input and uses three 1×1-convolutions to form the query $\mathbf{Q}$ and the key-value pair $(\mathbf{K}, \mathbf{V})$, where $\mathbf{Q}$ and $\mathbf{K}$ have shape $[B, C', T, F]$, and $\mathbf{V}$ has shape $[B, C, T, F]$. To reduce memory requirements, we use $C' = C/8$. Next, $\mathbf{Q}$, $\mathbf{K}$ and $\mathbf{V}$ are reshaped to form 3D matrices.

In order to compute the attention component $\mathbf{A}$, we first compute the weight $\mathbf{W}$, given by

$$\mathbf{W} = \mathbf{Q}^{\mathsf{T}}\mathbf{K}, \tag{1}$$

and then use the soft-max function $\sigma(\cdot)$ to obtain $\widehat{\mathbf{W}} = \{\widehat{W}_{i,j}\} = \sigma(\mathbf{W})$, i.e.,

$$\widehat{W}_{i,j} = \frac{\exp(W_{i,j})}{w_j}, \text{ where } w_j = \sum_{i=1}^{T \cdot F} \exp(W_{i,j}). \tag{2}$$

The attention component $\mathbf{A} \in \mathbb{R}^{B \times C \times T \times F}$ is now determined using

$$\mathbf{A} = \widehat{\mathbf{W}}\mathbf{V}^{\mathsf{T}}, \tag{3}$$

The attention module outputs $\widehat{\mathbf{X}} = \mathbf{X} + \delta\mathbf{A}$, where δ is a learnable scalar with initial value zero.

C. Proposed SASE framework

The SASE framework, shown in Fig. 2, has three main components: an encoder-decoder based CRN, a LSTM-based speaker embedding extractor and a self-attention module. The encoder of the CRN consists of five 2-D convolutional blocks, each of which includes a 2-D convolutional layer, a batch normalization layer [32], and exponential linear units (ELUs) [33]. The decoder uses five 2-D deconvolutional blocks to convert the low-resolution features into high-resolution spectrograms. Each deconvolutional block consists of a 2-D transposed convolutional layer, followed by batch normalization and the ELU activation. We include skip connections from each encoder layer to its corresponding decoder layer, in order to avoid losing fine-resolution details and to facilitate optimization. There are two LSTM layers between the encoder and decoder to capture long-term temporal dependencies.

Training Flow. The proposed SASE method takes noisy speech and reference clean speech as input. The reference clean speech is fed into the speaker embedding extractor to obtain speaker information. The noisy speech is fed into the encoder to determine the low-resolution features. The concatenation of the speaker representation and the encoder output are then fed into LSTM layers. The LSTM output is also fed into the self-attention module. The attention output is then followed by the encoder. We apply a sigmoid at the encoder output to generate a [0-1] mask. The following loss function, referred to as SA-MSE, is used during the training stage:

$$\mathcal{L} = \|\mathbf{M} \odot \mathbf{X} - \hat{\mathbf{X}}\|_{2.} \tag{4}$$

where $\mathbf{X}$ and $\hat{\mathbf{X}}$ denote the magnitude of the noisy speech and clean speech signals, respectively, and the operator $\odot$ denotes the Hadamard product. The mean squared error (MSE) loss function that is determined using the clean and predicted magnitude directly is referred to as SM-MSE. When performing the inverse STFT to reconstruct the waveform, we use the phase of the original noisy speech.

IV. Experiments and Results

In the following, the data set, model set up and the evaluation metrics are detailed. The results will be discussed at the end of this section.

A. Data Set

The database used here is derived from the Valentini-Botinhao corpus [27]: 84 speakers and two speakers in the original data set are used for training and test, respectively. Each speaker fragment consist of about 10 different sentences. The noisy training set used here considers 40 conditions: 10 noise types (two artificial noise types and eight noise types selected from the Demand database [29]), where each noise type is considered at an SNR of 0 dB, 5 dB, 10 dB, 15 dB. For the test set, a total of 20 different conditions are considered: five types of noise (all from the Demand database) with four SNRs each (2.5 dB, 7.5 dB, 12.5 dB and 17.5 dB). There are around 20 different sentences in each condition for each test speaker. The test set condition is totally different with the training set, as it uses different speakers and conditions. For each speaker, we generate a 60-second segment as clean reference speech. The clean reference is processed by removing the silence part. After holding out the utterance for clean reference, there are 722 sentences in total for testing. During the training stage or testing stage, we randomly choose a small segment from the clean reference for the given segment size, e.g., 2 s, 4 s, 6 s, and 8 s.

B. Model Setup

The baseline systems considered here are the LSTM- and CRN-based speech enhancement methods. The LSTM baseline model consists of two LSTM layers with 768 nodes each, followed by a fully-connected output layer that reduces the dimension to 161. The CRN-based method consists of five conv2d blocks with filters of size 3×2 each and [16, 32, 64, 128, 128] output channels, respectively. This output is post-processed by two LSTM layers with 512 nodes each, followed by five deconv2d blocks with filter size 3×2 each and output channels [128, 64, 32, 16, 1], respectively.

The proposed SASE method has a similar encoder-decoder as the CRN-based method. The speaker representation (256-D) and the encoder output (512-D) are concatenated and then fed into two LSTM layers of 768 nodes each. The output is then projected onto 512 feature dimensions and reshaped to match the encoder output, and then post-processed by the self-attention module and the decoder.

The feature input for all models is a spectral magnitude vector of length 161 of the noisy speech signal, which is computed using a STFT with a 20 ms Hamming window and a 10 ms window shift. All models are trained using the Adam optimizer [34] with an initial learning rate of 0.0006. A mini-batch size of 32 utterances is used for all models except for SASE with attention. SASE with attention uses mini-batch size of 16 utterances. We zero-pad all utterances to have the same length as the longest utterance within a mini-batch.

C. Evaluation Metrics

The speech enhancement systems are evaluated using the commonly used *perceptual evaluation of speech quality* (PESQ) score [35]–[37], the *short-time objective intelligibility* (STOI) score [38], and the scale-invariant signal-to-distortion ratio (SI-SDR) [39]. Three subjective scores that measure signal distortion CSIG, background intrusiveness CBAK and overall quality COVL scores [40] are used as well.

D. Experiments and Results

We first investigate the performance of all models by determining the SM-MSE loss. The results are provided in Table I. The performance metrics for the CRN-based method are better than the LSTM-based method, except in terms of PESQ. The proposed SASE approach outperforms the CRN baseline system, even with only 2 s reference clean speech. The best performance when applying the SM-MSE loss function

TABLE I
PERFORMANCE SCORES FOR THE PROPOSED SASE AND BASELINE SYSTEMS

	Loss	Model size	PESQ	STOI	SI-SDR	CSIG	CBAK	COVL
Noisy Speech	–	–	1.970	92.06	8.51	3.35	2.45	2.63
LSTM	SM-MSE	7.71 M	2.608	93.44	16.36	2.91	3.10	2.74
CRN		4.69 M	2.598	93.49	16.52	3.31	3.14	2.94
SASE (2s)		10.33 M (12.13 M)	2.636	93.67	16.80	3.42	3.18	3.02
SASE (4s)			2.627	93.72	16.73	3.44	3.18	3.02
SASE (6s)			2.649	93.80	16.93	3.48	3.20	3.05
SASE (8s)			2.651	93.72	16.84	3.52	3.19	3.07
LSTM	SA-MSE	7.71 M	2.614	93.65	16.70	3.96	3.19	3.29
CRN		4.69 M	2.658	93.87	16.67	4.02	3.22	3.34
SASE (2s)		10.33M (12.13 M)	2.702	93.95	16.86	4.08	3.26	3.40
SASE (4s)			2.699	**94.07**	16.97	**4.09**	3.27	3.40
SASE (6s)			2.696	94.00	16.92	4.08	3.26	3.40
SASE (8s)			2.693	93.98	17.05	4.08	3.27	3.39
SASE (2s) + attn	SA-MSE	10.35M (12.13 M)	2.670	93.92	17.14	4.05	3.26	3.36
SASE (4s) + attn			2.706	94.02	17.34	4.05	3.29	3.38
SASE (6s) + attn			**2.756**	94.05	**17.35**	**4.09**	**3.32**	**3.43**
SASE (8s) + attn			2.703	93.97	17.23	4.05	3.28	3.38

The values in parenthesis specify the duration of the reference speech signals. The best score in a column is **bold-faced**, the second best is navy blue and the third best is dark pink.

is achieved by SASE with 8 s reference speech. This indicates that additional speaker information is useful to further improve speech enhancement performance. Next, we replace the SM-MSE loss function by SA-MSE at the training stage. Table I shows that all SA-MSE-based loss models perform better than the models that use the SM-MSE loss function, in particular for the PESQ, CSIG and COVL scores. For instance, relative to SASE with 2 s reference speech using SM-MSE loss, the PESQ score improves from 2.636 to 2.702 and the COVL score improves from 3.02 to 3.40.

Further adding self-attention can boost the performance as well in terms of most metrics. We observe that adding self-attention improves the SI-SDR consistently for all SASE-based approaches. The best PESQ, SI-SDR, CISG, CBAK, and COVL scores are achieved by SASE with a 6-second reference speech signal.

Fig. 3 shows the PESQ, STOI and SI-SDR scores for both the proposed SASE system (with 6-s self-attention) and the baseline systems as a function of SNR. One can clearly see that the proposed SASE method is more effective at lower SNR. This suggests that additional speaker information provides important cues to distinguish the speech and noise when there is a lot of noise.

Model complexity. The proposed SASE method has more parameters than the baseline systems, because of the three LSTM layers of the speaker embedding extractor. It is planned to replace the LSTM-based speaker embedding extractor with an extractor that uses CNN models with fewer parameters.

V. CONCLUSIONS

In this paper, we presented and validated a novel speaker-aware speech enhancement method that uses a few seconds of reference clean speech. We first compared the proposed

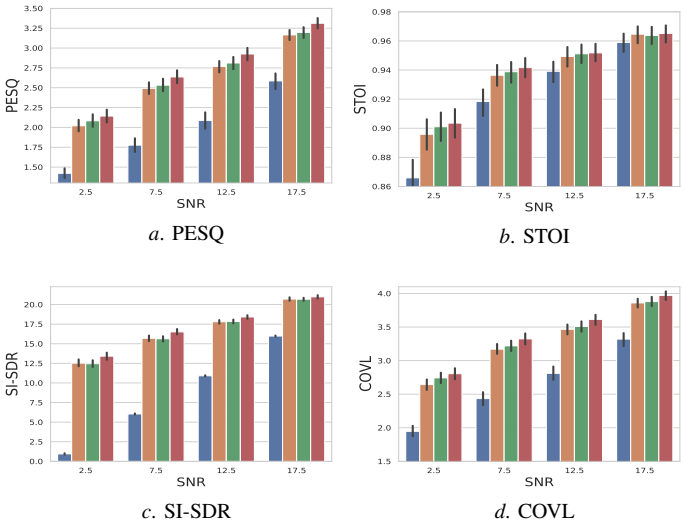

Fig. 3. PESQ, STOI, SI-SDR and COVL scores for noisy speech (blue), the LSTM-method (orange), the CRN-based method (green) and the proposed SASE system with 6-s self-attention (red) when the SNR is 2.5, 7.5, 12.5, and 17.5 dB.

SASE with baseline systems using spectral mapping-MSE and mask-based signal approximation-MSE loss, respectively. The experimental results indicate that the proposed SASE system outperforms the baseline systems using both loss functions. The results also show that using mask-based signal approximation loss is better than spectral mapping-MSE loss. Adding self-attention achieves the best performance in terms of most metrics, especially for SI-SDR metric. We tested the proposed SASE approach with four different reference-speech durations. All achieved better performance in comparison with the CRN baseline, which demonstrates the effectiveness of the

proposed method.

ACKNOWLEDGMENT

This work is supported by the US Army Medical Research and Materiel Command under Contract No. W81XWH-17-C-0238.

REFERENCES

[1] P. C. Loizou, *Speech Enhancement*, 2nd ed. Boca Raton, FL: CRC Press, 2013.

[2] A. L. Maas, Q. V. Le, T. M. O'Neil, O. Vinyals, P. Nguyen, and A. Y. Ng, "Recurrent neural networks for noise reduction in robust ASR," in *Proc. Interspeech*, Portland, OR, Sep. 2012, pp. 22–25.

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

[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. Audio, Speech, Language Process.*, vol. 23, no. 1, pp. 7–19, Jan. 2015.

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

[6] S. Pascual, A. Bonafonte, and J. Serrà, "SEGAN: Speech enhancement generative adversarial network," in *Proc. Interspeech*, Stockholm, Sweden, Aug. 2017, pp. 3642–3646.

[7] S.-W. Fu, C.-F. Liao, Y. Tsao, and S.-D. Lin, "MetricGAN: Generative adversarial networks based black-box metric scores optimization for speech enhancement," in *Proc. Int'l Conf. on Machine Learning*, Long Beach, CA, 2019, pp. 2031–2041.

[8] J. Lin, S. Niu, Z. Wei, X. L. A. J. van Wijngaarden, M. C. Smith, and K.-C. Wang, "Speech enhancement using forked generative adversarial networks with spectral subtraction," in *Proc. Interspeech*, Graz, Austria, Sep. 2019, pp. 3163–3167.

[9] D. Baby and S. Verhulst, "SERGAN: Speech enhancement using relativistic generative adversarial networks with gradient penalty," in *Proc. IEEE Int'l Conf. Acoustics, Speech and Signal Proc.*, Brighton, United Kingdom, May 2019, pp. 106–110.

[10] J. Lin, S. Niu, A. J. van Wijngaarden, J. L. McClendon, M. C. Smith, and K.-C. Wang, "Improved speech enhancement using a time-domain GAN with mask learning," in *Proc. Interspeech*, Shanghai, China, Oct. 2020, pp. 3286–3290.

[11] R. Giri, U. Isik, and A. Krishnaswamy, "Attention Wave-U-Net for speech enhancement," in *Proc. IEEE Workshop on Appl. of Signal Proc. to Audio and Acoustics*, New Paltz, NY, Oct. 2019, pp. 249–253.

[12] J. Kim, M. El-Khamy, and J. Lee, "T-GSA: Transformer with Gaussian-weighted self-attention for speech enhancement," in *Proc. IEEE Int'l Conf. Acoustics, Speech and Signal Proc.*, Barcelona, Spain, May 2020, pp. 6649–6653.

[13] Y. Zhao, D. Wang, B. Xu, and T. Zhang, "Monaural speech dereverberation using temporal convolutional networks with self attention," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 28, p. 1598–1607, 2020.

[14] Y. Koizumi, K. Yatabe, M. Delcroix, Y. Masuyama, and D. Takeuchi, "Speech enhancement using self-adaptation and multi-head self-attention," in *Proc. IEEE Int'l Conf. Acoustics, Speech and Signal Proc.*, Barcelona, Spain, May 2020, pp. 181–185.

[15] A. Pandey and D. Wang, "Dense CNN with self-attention for time-domain speech enhancement," *arXiv:2009.01941*, Sep. 2020.

[16] Y. Zhao and D. Wang, "Noisy-reverberant speech enhancement using DenseUNet with time-frequency attention," in *Proc. Interspeech*, Shanghai, China, Oct. 2020, pp. 3261–3265.

[17] J. Lin, A. J. van Wijngaarden, M. C. Smith, and K.-C. Wang, "Speech enhancement using multi-stage self-attentive temporal convolutional networks," *arXiv:2102.12078*, Feb. 2021.

[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Advances in Neural Information Proc. Sys.*, Long Beach, CA, Dec. 2017, pp. 5998–6008.

[19] J.-C. Hou, S.-S. Wang, Y.-H. Lai, Y. Tsao, H.-W. Chang, and H.-M. Wang, "Audio-visual speech enhancement using multimodal deep convolutional neural networks," *IEEE Trans. Topics Comput. Intell.*, vol. 2, no. 2, pp. 117–128, Apr. 2018.

[20] C.-F. Liao, Y. Tsao, X. Lu, and H. Kawai, "Incorporating symbolic sequential modeling for speech enhancement," in *Proc. Interspeech*, Graz, Austria, Sep. 2019, pp. 2733–2737.

[21] Y.-J. Lu, C.-F. Liao, X. Lu, J. weih Hung, and Y. Tsao, "Incorporating broad phonetic information for speech enhancement," in *Proc. Interspeech*, Shanghai, China, Oct. 2020, pp. 2417–2421.

[22] F.-K. Chuang, S.-S. Wang, J. weih Hung, Y. Tsao, and S.-H. Fang, "Speaker-aware deep denoising autoencoder with embedded speaker identity for speech enhancement," in *Proc. Interspeech*, Graz, Austria, Sep. 2019, pp. 3173–3177.

[23] Q. Wang, H. Muckenhirn, K. Wilson, P. Sridhar, Z. Wu, J. Hershey, R. A. Saurous, R. J. Weiss, Y. Jia, and I. Lopez Moreno, "Voicefilter: Targeted voice separation by speaker-conditioned spectrogram masking," in *Proc. Interspeech*, Graz, Austria, Sep. 2019, pp. 2728–2732.

[24] K. Žmolíková, M. Delcroix, K. Kinoshita, T. Ochiai, T. Nakatani, L. Burget, and J. Černockỳ, "SpeakerBeam: Speaker aware neural network for target speaker extraction in speech mixtures," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 4, pp. 800–814, Aug. 2019.

[25] X. Ji, M. Yu, C. Zhang, D. Su, T. Yu, X. Liu, and D. Yu, "Speaker-aware target speaker enhancement by jointly learning with speaker embedding extraction," in *Proc. IEEE Int'l Conf. Acoustics, Speech and Signal Proc.*, Barcelona, Spain, May 2020, pp. 7294–7298.

[26] L. Qu, C. Weber, and S. Wermter, "Multimodal target speech separation with voice and face references," in *Proc. Interspeech*, Shanghai, China, Oct. 2020, pp. 1416–1420.

[27] C. Valentini-Botinhao, X. Wang, S. Takaki, and J. Yamagishi, "Investigating RNN-based speech enhancement methods for noise-robust Text-to-Speech," in *Proc. ISCA Speech Synthesis Workshop*, Sunnyvale, CA, Sep. 2016, pp. 146–152.

[28] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Machine Learning Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

[29] J. Thiemann, N. Ito, and E. Vincent, "The diverse environments multi-channel acoustic noise database: A database of multichannel environmental noise recordings," *J. Acoustical Soc. of America*, vol. 133, no. 5, pp. 3591–3591, 2013.

[30] L. Wan, Q. Wang, A. Papir, and I. Lopez Moreno, "Generalized end-to-end loss for speaker verification," in *Proc. IEEE Int'l Conf. Acoustics, Speech and Signal Proc.*, Calgary, AB, Apr. 2018, pp. 4879–4883.

[31] J. S. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep speaker recognition," in *Proc. Interspeech*, Hyderabad, India, Sep. 2018, pp. 1086–1096.

[32] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int'l Conf. on Machine Learning*, Lille, France, Jul. 2015, pp. 448–456.

[33] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," in *Proc. Int'l Conf. on Learning Representations*, San Juan, Puerto Rico, May 2016, pp. 1–14.

[34] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. Int'l Conf. on Learning Representations*, San Diego, CA, May 2015, pp. 1–15.

[35] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ) – a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int'l Conf. Acoustics, Speech and Signal Proc.*, Salt Lake City, UT, May 2001, pp. 749–752.

[36] *Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs*, ITU-P recommendation P.862, Feb. 2001.

[37] *Perceptual objective listening quality prediction*, ITU-P recommendation P.863, Mar. 2018.

[38] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 19, no. 7, pp. 2125–2136, Sep. 2011.

[39] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "SDR – half-baked or well done?" in *Proc. IEEE Int'l Conf. Acoustics, Speech and Signal Proc.*, 2019, pp. 626–630.

[40] Y. Hu and P. C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 16, no. 1, pp. 229–238, Jan. 2008.