INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Efficient Low-Latency Speech Enhancement with Mobile Audio Streaming Networks

Michal Romaniuk, Piotr Masztalski, Karol Piaskowski, Mateusz Matuszewski

Samsung R&D Institute Poland
{m.romaniuk2, p.masztalski, k.piaskowski, m.matuszews2}@samsung.com

Abstract

We propose Mobile Audio Streaming Networks (MASnet) for efficient low-latency speech enhancement, which is particularly suitable for mobile devices and other applications where computational capacity is a limitation. MASnet processes linear-scale spectrograms, transforming successive noisy frames into complex-valued ratio masks which are then applied to the respective noisy frames. MASnet can operate in a low-latency incremental inference mode which matches the complexity of layer-by-layer batch mode. Compared to a similar fully-convolutional architecture, MASnet incorporates depthwise and pointwise convolutions for a large reduction in fused multiply-accumulate operations per second (FMA/s), at the cost of some reduction in SNR.

Index Terms: low-latency speech enhancement, deep learning

1. Introduction

Speech enhancement systems are an important component of modern communication tools such as mobile telephony, VoIP and smart earphones (hearables). They can also be used as a preprocessing step in speech recognition systems [1]. With recent advancements in deep learning, there has been renewed interest in the speech enhancement problem, bringing quality and intelligibility improvements over prior methods [2]. In this work we introduce Mobile Audio Streaming Networks (MASnet), an efficient convolutional neural network architecture for speech enhancement that can be implemented in a low-latency inference mode.

Our contributions are as follows:

- We propose MASnet, an efficient low-latency convolutional architecture for spectrogram-based audio processing;
- We evaluate MASnet in several size variations on the standard Valentini-Botinhao noisy speech database;
- We show that further performance gains can be achieved by adding residual connections.

2. Background

The problem of speech enhancement can be approached algorithmically in several ways. The first imporant choice is that of signal representation. Some algorithms work directly with waveforms [3, 4, 5, 6, 7], while others are based on spectrogram representations computed with the short-time Fourier transform (STFT) [8, 9]. Some other methods work with Mel-scale spectrograms [1]. For processing algorithms based on deep learning, the spectrogram representation has the advantage of extracting meaningful features and effectively increasing the temporal receptive fields of neural network cells. Due to the efficiency of the FFT algorithm, this is achieved with only a small cost.

In the context of STFT-based methods, a direct approach consists of noisy spectrogram analysis followed by re-synthesis of denoised spectrograms [10]. In practice, a more common method is noisy spectrogram analysis followed by synthesis of a *mask* which is then multiplied with the noisy spectrogram to produce the denoised version. This approach has the advantage that the masks are thought to be easier to estimate than detailed spectrograms [11] and this approach has been found to work better experimentally with multilayer perceptrons [10].

The spectrogram masking approach has several further variations: the masks can be binary-valued (0 or 1), real-valued or complex-valued [10, 11, 12]. With binary-valued and real-valued masks it is common to reconstruct waveforms from denoised STFT magnitude and (noisy) phase of the input, but it is also possible to use the Griffin-Lim phase reconstruction algorithm [13] or train a phase model in addition to the amplitude model [14]. Complex-valued masks have the advantage of being able to adjust STFT phase in conjunction with magnitude. The method proposed in this paper is based on the complex ratio masking approach.

2.1. Low-latency speech enhancement

Spectrogram-based speech enhancement methods often treat STFT representations as images [9, 8]. This is fine at training-time, but at test-time and especially in low-latency applications such as telephony it is important to be able to output processed audio immediately, without having to record several seconds that could be transformed into a spectrogram.

Some speech enhancement methods achieve low latency with causal recurrent neural net (RNN) models [5]. Low-latency convolutional models such as Wavenet [15] and TCN [16] have also been used for speech enhancement [6, 17]. However, these models have to learn their own feature representations and compute them at inference time, which tends to require more processing power than STFT. Causal convolutional processing of STFT spectrograms was studied by Wilson *et al.* [18] and this idea was also combined with complex ratio masking as part of an audio-visual processing pipeline [19]. Their network architecture is a starting point for our work which aims to reduce its computational complexity.

Tan and Wang [20] proposed a causal encoder-core-decoder type network where the core is an LSTM stack, with additional skip connections between the encoder and the decoder. Their network operates on STFT magnitude spectra and uses striding to increase receptive field size in the frequency dimension while the LSTM provides temporal memory. In contrast, our model is based on complex-valued STFT spectra, enabling it take advantage of phase information and adjust the phase of the output spectra. Pandey and Wang [17] have developed an encoder-core-decoder type architecture where the core is a sequence of temporal convolutional blocks and there are skip con-

http://dx.doi.org/10.21437/Interspeech.2020-2443

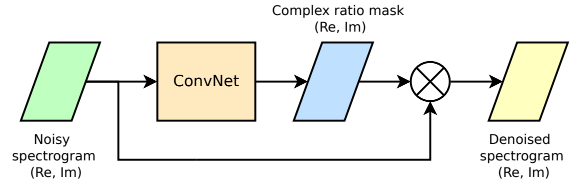

Figure 1: *Model overview*

nections from the encoder and the decoder. They have also used combined depthwise and pointwise convolutions in their core blocks. However, their model operates directly on frames extracted from the waveform, rather than on spectrograms.

2.2. Efficient convolutional architectures

Convolutional neural networks often require high-performance hardware for fast inference, so finding network architectures that come close to state-of-the-art prediction accuracy while taking less processing power is an important research topic [21, 22, 23, 24, 25]. Flattened ConvNets [21] are based on factored convolutions. Factorized ConvNets [23] combine intra-channel convolutions with topological subdivisioning. Squeezenets [22] are built from Fire modules, each consisting of a 1x1 convolution that squeezes the number of channels and a bank of 1x1 and 3x3 convolutions that expand it again. MobileNets [24] replace standard convolutions with pairs of depthwise and pointwise (1x1) convolutions. Our proposed MASnet architecture borrows from MobileNets [24], by replacing standard convolutions with pairs of depthwise and pointwise convolutions. We chose MobileNets because of their popularity in the computer vision community.

3. Model details

3.1. Feature representation

The models considered in this paper are all designed to process spectrograms represented as two-channel (real and imaginary) images and output two-channel images which are interpreted as real and imaginary parts of complex-valued ratio masks and then multiplied with the input spectrograms to produce denoised output spectrograms. The output spectrograms are then turned into output waveforms with inverse STFT. The spectrogram representations are computed with the (one-sided) STFT, using a Hann window with size 256 and stride 128. This gives 129 frequency bins. For input sampled at 16kHz, the frame rate is 125 per second.

3.2. LLASnet

We adopt the causal convolutional stack from [18] (without the LSTM and FC layers) and refer to it as Low-Latency Audio Streaming network (LLASnet). Detailed layer configurations of the LLASnet layers in two size versions: LLASnet-8 and LLASnet-15, are shown in table 1. All convolutions are computed with stride 1 and dilations are used to increase the receptive fields. Zero-padding is added in the convolutions as necessary to ensure that feature maps all have the same dimensions as the input frequency grid. At training time subsequent layers are computed in the conventional layer-wise way and causality is enforced by adding all the necessary zero-padding at the start of the time axis, with no padding at the end. The frequency axis is zero-padded equally on both ends. Hidden layers all use batchnorm [26] followed by ReLU activations. The output layers use a linear activation without batchnorm. These models are the baseline for our further investigations. We show that LLASnet can approach the state-of-the-art CRMRN U-net [9] in terms of speech quality metrics while achieving low latency, but at the cost of more computation.

Table 1: *LLASnet-15 layer configuration (kernel shape and dilation in [time × frequency] format). Layers marked with an asterisk form the LLASnet-8 version.*

Type	Channels	Kernel	Dilation
Conv2D*	32	1x7	1x1
Conv2D*	32	7x1	1x1
Conv2D*	32	5x5	1x1
Conv2D*	32	5x5	2x1
Conv2D*	32	5x5	4x1
Conv2D*	32	5x5	8x1
Conv2D*	32	5x5	16x1
Conv2D	32	5x5	32x1
Conv2D	32	5x5	1x1
Conv2D	32	5x5	2x2
Conv2D	32	5x5	4x4
Conv2D	32	5x5	8x8
Conv2D	32	5x5	16x16
Conv2D	32	5x5	32x32
Conv2D*	2	1x1	1x1

3.3. MASnet

LLASnet can be implemented in a low-latency inference mode but it requires too much processing power to be practical in mobile or low-power applications. Our proposed model, the Mobile Audio Streaming Network (MASnet), is a further development of LLASNet, reducing the computation required. We incorporate the ideas from MobileNets [24] to reduce the number of fused multiply-accumulates per second (FMA/s) by a factor of roughly 10.

The main change is to replace each convolution (including its batchnorm and ReLU) with a MAS block (called *depthwise separable convolution* in [24]), consisting of a depthwise convolution followed by a pointwise (1x1) convolution, each of them followed by batchnorm and ReLU activations. Before the first layer of the network, an additional 1x1 convolution is added to expand input channel count from 2 to 32 so that the first depthwise convolution can run on 32-channel input. Applying these changes to LLASnet-8 and LLASnet-15 produces MASnet-9 and MASnet-16, respectively. These new networks are summarized in table 2. We also explore deeper MASnet versions by appending further layers to MASnet-16. The configuration of these extra layers is a repetition of the last 6 layers of MASnet-16 (marked with asterisks in table 2). By repeating this sequence 1, 2 or 3 times, we get MASnet-22, MASnet-28 and MASnet-34 respectively.

Table 2: *MASnet-16 layer configuration (kernel shape and dilation in [time × frequency] format). Layers marked with † form the MASnet-9 version. By repeating the sequence of layers marked with asterisks we get MASnet-22, MASnet-28, MASnet-34 etc.*

Type	Channels	Kernel	Dilation
Conv2D†	32	1x1	1x1
MAS block†	32	1x7	1x1
MAS block†	32	7x1	1x1
MAS block†	32	5x5	1x1
MAS block†	32	5x5	2x1
MAS block†	32	5x5	4x1
MAS block†	32	5x5	8x1
MAS block†	32	5x5	16x1
MAS block	32	5x5	32x1
MAS block*	32	5x5	1x1
MAS block*	32	5x5	2x2
MAS block*	32	5x5	4x4
MAS block*	32	5x5	8x8
MAS block*	32	5x5	16x16
MAS block*	32	5x5	32x32
Conv2D†	2	1x1	1x1

3.4. Incremental low-latency inference

Both LLASnet and MASnet can be implemented in an incremental low-latency inference mode, taking one spectrogram frame at a time and returning output immediately. This process is shown in *fig.* 2. Each time a new spectrogram frame arrives, we compute the corresponding frame of successive feature maps, using the cached frames from the most recent steps to reduce computational complexity. Due to having the same feature map dimensions for all layers, this means that we end up doing only as much computation as in layer-by-layer inference mode. While incremental inference is not very common for image processing, it has been studied before in the context of sequence models [16, 27].

3.5. Training

The network is trained by backpropagation, using the Adam optimizer [28]. As the loss function, we use the Mean Squared Error (MSE) between the denoised spectrogram (after masking) and the clean spectrogram:

$$L(\hat{x}, x) = \frac{1}{TF} \sum_{t,f} \left(Re\left(\hat{x} - x\right)^2 + Im\left(\hat{x} - x\right)^2 \right) \qquad (1)$$

where x is the clean reference spectrogram and $\hat{x}$ is the denoised estimate (noisy input multiplied by the predicted mask). The summation variables t and f are time and frequency respectively, while T is the number of time steps (frames) and F is the number of frequency bins.

4. Experiments

To evaluate MASnet, we use a standard dataset [29, 30] and compare with LLASnet-8, LLASnet-15 and the CRMRN U-net from [9, 8]. All of our experiments are implemented with PyTorch [31].

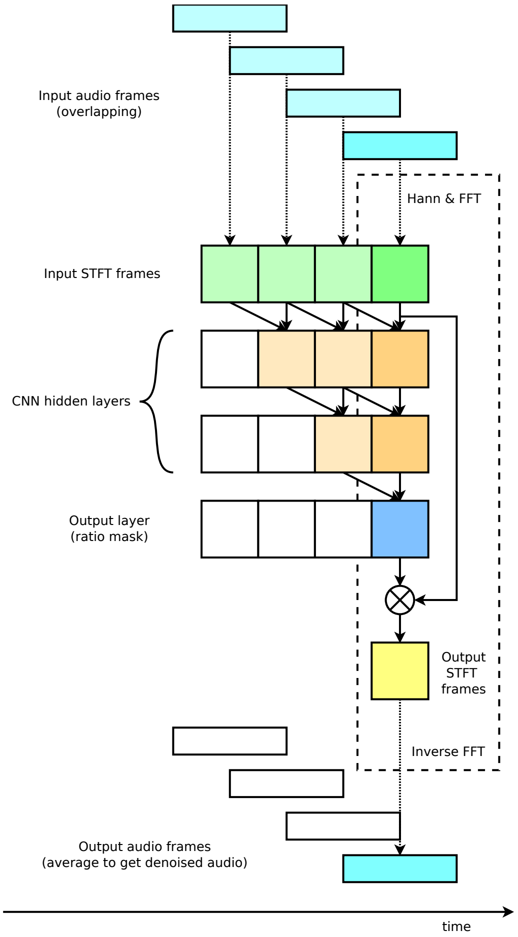

Figure 2: *Low-latency inference. The dashed box contains the computations that have to be done for each new audio frame*

4.1. Training on the Valentini-Botinhao database

We evaluate the proposed network designs on a noisy speech database published by Valentini-Botinhao [29, 30], which consists of recorded spoken phrases in two training sets: 28 speakers from England and 56 from other regions. The recordings are sampled at 48 kHz, with a balance between male and female speakers. Each of the clean recordings has a time-aligned noisy version with noise mixed in at SNR of 15 dB, 10 dB, 5 dB or 0 dB. The test set is based on two other speakers from England (one male and one female) and has other types of noise mixed in, with higher SNRs: 17.5 dB, 12.5 dB, 7.5 dB or 2.5 dB.

We combined the 28-speaker and 56-speaker training sets into one large training set. All training and test recordings were resampled to 16kHz. To make it possible to train in batched mode, we restricted utterance length to 3 × 16384 samples (slightly over 3 seconds), by either truncating the recordings at the end or zero-padding them equally at the beginning and end. In order to select a well-performing training checkpoint for evaluation, we created a validation set from the first male and first female speakers in the training set (p226 and p228). These recordings were only used for validation and not for training.

We trained all of the models for 200 epochs with batch size 16, using the Adam optimizer [28] with learning rate 10^{-4}, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. Then for each model we selected the

Table 3: *(Table 5) Model performance comparison on Valentini-Botinhao database. SNR is in dB. All models were trained for up to 200 epochs. (Checkpoint was selected by best result on validation data.)*

Model	**FMA/s**	**SNR**	**STOI**	**PESQ**
(Noisy data)	—	8.45	0.9210	1.972
LLASnet-8	2240M	16.36	0.9216	2.383
LLASnet-15	5199M	17.26	0.9298	2.364
MASnet-9	224M	15.40	0.9155	2.193
MASnet-16	404M	15.85	0.9192	2.213
MASnet-22	584M	16.18	0.9219	2.113
MASnet-28	765M	15.70	0.9207	2.035
MASnet-34	945M	15.80	0.9192	2.124
CRMRN U-net	403M	18.38	0.9333	2.559

best checkpoint with respect to the validation set and we report its performance on the test set.

We evaluate our models using the following metrics: Signal-to-Noise Ratio (SNR) computed on whole waveforms without segmentation[1], Short-Time Objective Intelligibility (STOI) [32] and Perceptual Speech Quality (PESQ) [33]. Table 3 shows aggregate results.

CRMRN U-net has the most favorable performance figures and reasonable computational complexity, however it is not straightforward to adapt it for low-latency inference due to non-causal and temporally strided convolutions. In terms of SNR, LLASnet-15 comes within 1.12 dB of CRMRN U-net but its FMA/s requirement is too large for low-resource applications. LLASnet-8 requires less than half the computations of LLASNnet-15, but at the cost of a further decline in SNR of 0.9 dB. Meanwhile, MASnet-16 requires less than a tenth of the computations of LLASnet-15, with an SNR reduction of 1.41 dB. MASnet-22 requires slightly over 11% of the FMA/s of LLASnet-15 at a cost of 1.08 dB reduction in SNR. Deeper MASnet versions seem to do worse than MASnet-22 in terms of SNR.

4.2. Adding Residual Connections

Deeper versions of our model (MASnet-28 and MASnet-34) have worse results than MASNet-22, so we decided to see if the popular technique of adding residual connections [34, 35], which has also been used for audio [36, 17, 37] would help in this case. We added an identity bypass connection to each MASblock, which resulted in a family of residual networks (MASnet-R) with a negligible increase in FMA/s over their sequential versions. The results for MASnet-R are shown in table 4.

Table 4: *(Table 6) Residual MASnet (MASnet-R) comparison on Valentini-Botinhao database. SNR is in decibels. All models were trained for up to 200 epochs. (Checkpoint was selected by best result on validation data.)*

Model	**FMA/s**	**SNR**	**STOI**	**PESQ**
MASnet-R-9	≈224M	15.62	0.9185	2.306
MASnet-R-16	≈404M	16.20	0.9184	2.257
MASnet-R-22	≈584M	16.21	0.9246	2.273
MASnet-R-28	≈765M	15.83	0.9224	2.331
MASnet-R-34	≈945M	15.32	0.9192	2.378

Smaller MASnet-R versions (up to MASnet-R-22) have slightly higher SNR figures than their respective sequential ancestors, but the reverse is the case for MASnet-R-34. This result is somewhat surprising, since we expected residual connections to be most beneficial for larger networks. However, PESQ improves for all size versions. As for SNR and STOI, MASnet-R-22 is the best model of the MASnet family, with SNR of 16.21 dB 1.05 dB less than LLASnet-15), STOI of 0.9246 (0.0052 less than LLASnet-15) and PESQ of 2.273 (0.091 less than LLASnet-15). At the same time, MASnet-R-22 requires slightly over 11% of the FMA operations of LLASnet-15.

5. Conclusions

We propose MASnet for speech enhancement applications where latency and computational complexity are important factors. Our models achieve a large reduction in FMA/s compared to a similar fully-convolutional architecture, at the cost of some reduction in SNR. This is achieved by replacing standard convolutions with pairs of depthwise and pointwise convolutions, inspired by Mobilenets [24]. It may be possible to make further performance gains by adjusing the shapes of the depthwise kernels or other architecture refinements that were proposed in the literature [38, 39, 40, 25], or replacing dilations in the frequency dimension with striding [20]. Another direction for improvement would be to add an adversarial loss term [41, 42] in the training process, with the goal of improving the quality of denoised speech. Some work was done in this direction by other authors using differeent network architectures, working either with raw waveforms [3] or with spectrograms [43]. However, low-latency inference was not their focus.

6. References

[1] C. Donahue, B. Li, and R. Prabhavalkar, “Exploring speech enhancement with generative adversarial networks for robust speech recognition,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5024–5028.

[2] D. Wang and J. Chen, “Supervised speech separation based on deep learning: An overview,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 10, pp. 1702–1726, 2018.

[3] S. Pascual, A. Bonafonte, and J. Serra, “SEGAN: Speech enhancement generative adversarial network,” *arXiv preprint arXiv:1703.09452*, 2017.

[4] F. G. Germain, Q. Chen, and V. Koltun, “Speech denoising with deep feature losses,” *arXiv preprint arXiv:1806.10522*, 2018.

[5] Y. Luo and N. Mesgarani, “TasNet: time-domain audio separation network for real-time, single-channel speech separation,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 696–700.

[6] D. Rethage, J. Pons, and X. Serra, “A wavenet for speech denoising,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5069–5073.

[7] D. Stoller, S. Ewert, and S. Dixon, “Wave-U-net: A multi-scale neural network for end-to-end audio source separation,” *arXiv preprint arXiv:1806.03185*, 2018.

[1] *N.b.* this is different from segmental SNR (SSNR) which is also commonly used in speech enhancement

[8] J. Yao and A. Al-Dahle, “Coarse-to-fine optimization for speech enhancement,” *arXiv preprint arXiv:1908.08044*, 2019.

[9] H.-S. Choi, J.-H. Kim, J. Huh, A. Kim, J.-W. Ha, and K. Lee, “Phase-aware speech enhancement with deep complex U-net,” *https://openreview.net/forum?id=SkeRTsAcYm, retracted*, 2019.

[10] Y. Wang, A. Narayanan, and D. Wang, “On training targets for supervised speech separation,” *IEEE/ACM transactions on audio, speech, and language processing*, vol. 22, no. 12, pp. 1849–1858, 2014.

[11] D. Michelsanti, Z.-H. Tan, S. Sigurdsson, and J. Jensen, “On training targets and objective functions for deep-learning-based audio-visual speech enhancement,” in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 8077–8081.

[12] D. S. Williamson, Y. Wang, and D. Wang, “Complex ratio masking for monaural speech separation,” *IEEE/ACM transactions on audio, speech, and language processing*, vol. 24, no. 3, pp. 483–492, 2015.

[13] D. Griffin and J. Lim, “Signal estimation from modified short-time fourier transform,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, 1984.

[14] T. Afouras, J. S. Chung, and A. Zisserman, “The conversation: Deep audio-visual speech enhancement,” *arXiv preprint arXiv:1804.04121*, 2018.

[15] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, “Wavenet: A generative model for raw audio,” *arXiv preprint arXiv:1609.03499*, 2016.

[16] S. Bai, J. Z. Kolter, and V. Koltun, “An empirical evaluation of generic convolutional and recurrent networks for sequence modeling,” *arXiv preprint arXiv:1803.01271*, 2018.

[17] A. Pandey and D. Wang, “TCNN: Temporal convolutional neural network for real-time speech enhancement in the time domain,” in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6875–6879.

[18] K. Wilson, M. Chinen, J. Thorpe, B. Patton, J. Hershey, R. A. Saurous, J. Skoglund, and R. F. Lyon, “Exploring tradeoffs in models for low-latency speech enhancement,” in *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*. IEEE, 2018, pp. 366–370.

[19] A. Ephrat, I. Mosseri, O. Lang, T. Dekel, K. Wilson, A. Hassidim, W. T. Freeman, and M. Rubinstein, “Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation,” *arXiv preprint arXiv:1804.03619*, 2018.

[20] K. Tan and D. Wang, “A convolutional recurrent neural network for real-time speech enhancement.” in *Interspeech*, 2018, pp. 3229–3233.

[21] J. Jin, A. Dundar, and E. Culurciello, “Flattened convolutional neural networks for feedforward acceleration,” *arXiv preprint arXiv:1412.5474*, 2014.

[22] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and ¡ 0.5 mb model size,” *arXiv preprint arXiv:1602.07360*, 2016.

[23] M. Wang, B. Liu, and H. Foroosh, “Factorized convolutional neural networks,” in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pp. 545–553.

[24] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *arXiv preprint arXiv:1704.04861*, 2017.

[25] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted residuals and linear bottlenecks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[26] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” *arXiv preprint arXiv:1502.03167*, 2015.

[27] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, “Phoneme recognition using time-delay neural networks,” *IEEE transactions on acoustics, speech, and signal processing*, vol. 37, no. 3, pp. 328–339, 1989.

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

[29] C. Valentini-Botinhao, X. Wang, S. Takaki, and J. Yamagishi, “Investigating RNN-based speech enhancement methods for noise-robust text-to-speech.” in *SSW*, 2016, pp. 146–152.

[30] ——, “Speech enhancement for a noise-robust text-to-speech synthesis system using deep recurrent neural networks.” in *Interspeech*, 2016, pp. 352–356.

[31] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, “PyTorch: An imperative style, high-performance deep learning library,” in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alche Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 8024–8035.

[32] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, “An algorithm for intelligibility prediction of time-frequency weighted noisy speech,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011.

[33] ITU, “P. 862.2: Wideband extension to recommendation p. 862 for the assessment of wideband telephone networks and speech codecs,” *International Telecommunication Union, CH–Geneva*, 2005.

[34] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[35] ——, “Identity mappings in deep residual networks,” in *European conference on computer vision*. Springer, 2016, pp. 630–645.

[36] N. Takahashi and Y. Mitsufuji, “Multi-scale multi-band densenets for audio source separation,” in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2017, pp. 21–25.

[37] F. Li, K. Qian, M. Hasegawa-Johnson, and M. Akagi, “Monaural singing voice separation using fusion-net with time-frequency masking,” in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2019, pp. 1239–1243.

[38] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[39] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[40] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, “Inception-v4, inception-resnet and the impact of residual connections on learning,” in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[41] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[42] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[43] 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,” *arXiv preprint arXiv:1905.04874*, 2019.