INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

A Two-stage Approach to Speech Bandwidth Extension

Ju Lin [1,*], *Yun Wang* [2], *Kaustubh Kalgaonkar* [2], *Gil Keren* [2], *Didi Zhang* [2], *Christian Fuegen* [2]

[1] Clemson University, USA
[2] Facebook AI, USA

jul@clemson.edu, {yunwang, kaustubhk, gilkeren, didizbq, fuegen}@fb.com

Abstract

Algorithms for speech bandwidth extension (BWE) may work in either the time domain or the frequency domain. Time-domain methods often do not sufficiently recover the high-frequency content of speech signals; frequency-domain methods are better at recovering the spectral envelope, but have difficulty reconstructing the details of the waveform. In this paper, we propose a two-stage approach for BWE, which enjoys the advantages of both time- and frequency-domain methods. The first stage is a frequency-domain neural network, which predicts the high-frequency part of the wide-band spectrogram from the narrow-band input spectrogram. The wide-band spectrogram is then converted into a time-domain waveform, and passed through the second stage to refine the temporal details. For the first stage, we compare a convolutional recurrent network (CRN) with a temporal convolutional network (TCN), and find that the latter is able to capture long-span dependencies equally well as the former while using a lot fewer parameters. For the second stage, we enhance the Wave-U-Net architecture with a multi-resolution short-time Fourier transform (MSTFT) loss function. A series of comprehensive experiments show that the proposed system achieves superior performance in speech enhancement (measured by both time- and frequency-domain metrics) as well as speech recognition.

Index Terms: speech bandwidth extension, speech enhancement, speech recognition

1. Introduction

Speech bandwidth extension (BWE) aims to increase the sampling rate of a given low-resolution speech signal. It is used in many applications, including speech enhancement [1], speech synthesis [2] and speaker identification [3]. Early work focused on estimating the spectral envelope of speech signals and modeling the mapping from narrow-band to wide-band signals, and employed models such as Gaussian mixture models (GMMs) [1, 4, 5] and hidden Markov models (HMMs) [6, 7]. However, these models suffered from the over-smoothing problem due to their inadequate modeling abilities [8], limiting the quality of reconstructed speech signals. Recently, deep learning based BWE methods have achieved great success compared with conventional BWE approaches. In general, these methods can be divided into two categories: frequency-domain and time-domain. Frequency-domain methods typically learn a mapping from the narrow-band spectrogram to the wide-band spectrogram, or the high-frequency part of the latter. The mapping is usually implemented as a neural network, such as a deep neural networks (DNN) [9–11], or a long short-term memory (LSTM) network [12]. For example, Li *et al.* [10] proposed a DNN to predict the wide-band log-power spectrogram (LPS) from the narrow-band LPS. To artificially create the missing phase information in the high-frequency band, the phase spectrogram of the low-frequency band was flipped into the high-frequency band, and this symmetric phase spectrogram was used to reconstruct the time-domain signal. This method was shown to outperform GMM-based methods. Time-domain approaches, on the other hand, process raw time-domain signals directly with neural networks [13, 14]. For instance, Kuleshov *et al.* [14] proposed an end-to-end convolutional auto-encoder network, trained with the mean squared error (MSE) objective function to reconstruct time-domain signals. Several studies have also investigated the possibility to combine the advantages of both the time and frequency domains. A time-frequency networks (TFNet) was proposed in [15], which contain two jointly optimized branches that reconstruct the time- and frequency-domain representations of a signal respectively. Wang *et al.* [16] proposed to use a time-frequency loss to ensure the reconstructed signal is close to the target in both domains.

Generally, frequency-domain approaches achieve better frequency-domain metrics, *e.g.* log-spectral distance (LSD), while time-domain approaches achieve better time-domain metrics, *e.g.* signal-to-noise ratio (SNR). In this paper, we propose a two-stage approach for BWE to take advantage of both frequency- and time-domain neural mappings. Our proposed model is a cascade of two stages. The first stage is a frequency-domain neural network, which maps the narrow-band log-magnitude spectrogram to the high-frequency part of the target log-magnitude spectrogram. We investigate two architectures: a convolutional recurrent network (CRN) and a temporal convolutional network (TCN). The CRN [17] comprises a convolutional encoder-decoder structure which extracts high-level features with 2-D convolutions, as well as long short-term memory (LSTM) layers which capture long-span temporal dependencies. The TCN [18] consists of dilated 1-D convolutional layers, which create a large temporal receptive field with fewer parameters. The predicted wide-band log-magnitude spectrogram is combined with a mirrored phase spectrogram and converted into a time-domain waveform, which is fed into the second stage to refine the details. We use the Wave-U-Net [19] architecture for the second stage, and train it with a multi-domain loss function to ensure the quality of the reconstructed signal in both the time and the frequency domains. The effectiveness of our proposed method is verified on the Valentini-Botinhao corpus [20], using multiple evaluation metrics for speech enhancement focusing on different aspects.

2. Proposed Method

2.1. Overview and Notations

In this work, we study the bandwidth extension from 8 kHz speech signals to 16 kHz. Let $\boldsymbol{y}$ be the target 16 kHz signal in the time domain. We can perform a short-time Fourier trans-

*Work performed during internship at Facebook AI.

http://dx.doi.org/10.21437/Interspeech.2021-1941

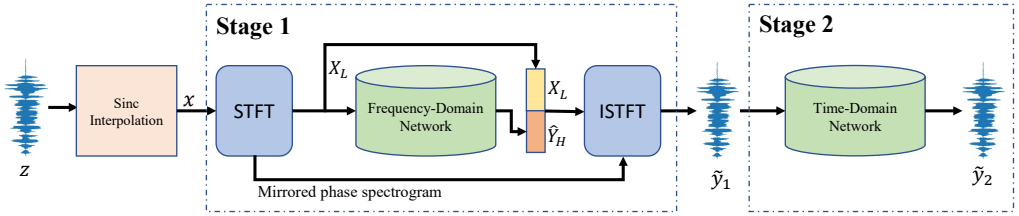


Figure 1: *Block diagram of the proposed two-stage BWE system.*

form (STFT) to obtain its log-magnitude spectrogram $\boldsymbol{Y}$; this can be divided equally into the low-band part $\boldsymbol{Y}_L$ and the high-band part $\boldsymbol{Y}_H$. The time-domain signal $\boldsymbol{y}$ can be downsampled to 8 kHz, and the resultant signal $\boldsymbol{z}$ is the input to our system.

The overall block diagram of our BWE system is shown in Fig. 1. First, we upsample the input signal $\boldsymbol{z}$ back to 16 kHz, filling in the missing samples using the simple method of sinc interpolation. The resultant waveform is denoted by $\boldsymbol{x}$, and its log-magnitude spectrogram is denoted by $\boldsymbol{X}$. The high-band part $\boldsymbol{X}_H$ of this spectrogram will contain infinitesimal numbers, while the low-band part $\boldsymbol{X}_L$ will be nearly identical to $\boldsymbol{Y}_L$.

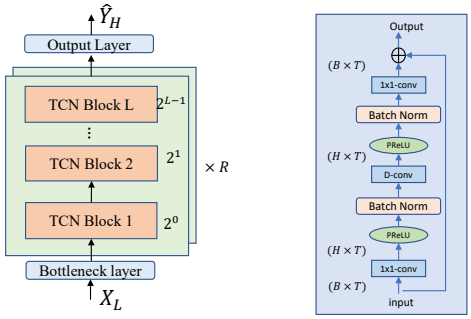
In the first stage, the TCN takes $\boldsymbol{X}_L$ as input, and predicts a high-band log-magnitude spectrogram $\hat{\boldsymbol{Y}}_H$. We concatenate $\boldsymbol{X}_L$ with $\hat{\boldsymbol{Y}}_H$ to form the predicted wide-band log-magnitude spectrogram $\hat{\boldsymbol{Y}}$. The encoder-decoder architecture of the CRN requires its input and output spectograms to be aligned, so it takes the entire $\boldsymbol{X}$ as input and predicts the entire wide-band spectrogram $\hat{\boldsymbol{Y}}$. We then replace its low-band part with the known $\boldsymbol{X}_L$.

To reconstruct the time-domain waveform, we also need a phase spectrogram. Following [10], we take the low-band half of $\boldsymbol{x}$'s phase spectrogram, and create the high-band half artificially by mirroring the former about the 4 kHz line and reverting the sign. An inverse STFT (ISTFT) is performed on the combined magnitude and phase spectrograms to reconstruct the time-domain waveform $\tilde{\boldsymbol{y}}_1$. The second stage takes $\tilde{\boldsymbol{y}}_1$ as input, and refine it into a new time-domain signal $\tilde{\boldsymbol{y}}_2$.

2.2. Frequency-domain Networks

CRN structure. The CRN takes $\boldsymbol{X}$ as input and it has an encoder-decoder structure. The encoder consists of six 2-D convolutional blocks, each of which includes a 2-D convolutional layer, a batch normalization layer [21], and the PReLU activation [22]. The output of the encoder is passed through two LSTM layers, which capture long-term temporal dependencies. The decoder consists of six 2-D deconvolutional blocks, and serves to convert the low-resolution features generated by the LSTM layers into high-resolution spectrograms. Each deconvolutional block consists of a 2-D transposed convolutional layer, followed by batch normalization and the PReLU 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. Finally, the output layer uses filters of size 1×1 to generate a wide-band log-magnitude spectrogram $\hat{\boldsymbol{Y}}$ in a single channel.

TCN structure. We adopt a similar architecture to [23], which is shown in Fig. 2. The low-band spectrogram $\boldsymbol{X}_L$ is first passed through a bottleneck layer to reduce the dimensionality from F to B. The trunk of the TCN consists of R identical stacks of L TCN blocks. Each TCN block comprises a 1×1 convolutional layer to increase the dimensionality from B to H, a dilated depth-wise convolutional (D-conv) layer with kernel size P and



(a) TCN (b) TCN Block

Figure 2: *The TCN architecture for speech bandwidth extension.*

varying dilation factors, and another 1×1 convolutional layer to reduce the dimensionality from H back to B. The dilation factor of the D-conv layer in the ℓ-th TCN block is $\Delta_\ell = 2^{\ell-1}$; these exponentially increasing dilation factors have been shown to form such a large receptive field that a TCN can outperform an RNN in temporal sequence modeling [23, 24]. A PReLU activation layer [22] and a batch normalization layer [21] are inserted both before and after each D-conv layer to accelerate training and improve performance. The output of each TCN stack is recombined with the input using a skip connection to avoid losing low-level details. Finally, the output layer uses a 1×1 convolutional layer to convert the dimensionality from B to F', the dimensionality of the high-band spectrogram.

2.3. Time-domain Network

We should note that the first stage is only trained to predict a log-magnitude spectrogram. Studies have shown that phase information is also important for achieving a good perceptual quality [25], and it is not ideal to reconstruct the time-domain waveform simply using a mirrored phase spectrogram. To refine the reconstructed waveform, we use a variant of Wave-U-Net [19] as the second stage of our proposed system. Wave-U-Net consists of downsampling (DS) blocks and upsampling (US) blocks. Skip connections are used between DS and US blocks to generate multi-scale features. Each DS block consists of a 1-D convolutional layer, followed by batch normalization and LeakyReLU activation [26]. To upsample the feature maps in the US blocks, instead of using transposed convolutions with strides, we perform linear interpolation to ensure temporal continuity. This is followed by a 1-D convolutional layer, batch normalization and LeakyReLU activation.

2.4. Network Training

The two stages of the network are trained separately. We first train the first stage using the mean squared error (MSE) loss function. The CRN predicts the entire wide-band log-

magnitude spectrogram $\hat{\boldsymbol{Y}}$, while the TCN only predicts its high-band part $\hat{\boldsymbol{Y}}_H$. Therefore the MSE loss function is also slightly different for the two choices:

$$\mathcal{L}_{\text{CRN}} = ||\hat{\boldsymbol{Y}} - \boldsymbol{Y}||_2, \tag{1}$$

$$\mathcal{L}_{\text{TCN}} = ||\hat{\boldsymbol{Y}}_H - \boldsymbol{Y}_H||_2. \tag{2}$$

With the first stage trained and its parameters fixed, we then train the second stage using a combination of an L_1 loss in the time domain and a multi-resolution short-time Fourier transform (MSTFT) loss [27] in the frequency domain:

$$\mathcal{L} = \lambda||\boldsymbol{y} - \tilde{\boldsymbol{y}}_2||_1 + \mathcal{L}_{\text{MSTFT}}(\boldsymbol{y}, \tilde{\boldsymbol{y}}_2), \tag{3}$$

where $\boldsymbol{y}$ and $\tilde{\boldsymbol{y}}_2$ are the target wide-band waveform and the prediction of the Wave-U-Net, respectively. To calculate the second term, we apply multiple STFTs with different parameters to $\boldsymbol{y}$ and $\tilde{\boldsymbol{y}}_2$, and sum up the L_1 losses between each pair of log-magnitude spectrograms. λ is a hyperparameter that controls the balance between the two loss terms. As we shall see in the experiments, the inclusion of the MSTFT loss played a key role in improving the performance of the Wave-U-Net.

3. Experiments and Results

3.1. Dataset

Speech bandwidth extension. We evaluate BWE with the Valentini-Botinhao corpus [20], and follow the official split: 28 speakers for training, and two speakers for testing. The original utterances are sampled at 48 kHz; we downsample them to 16 kHz as the target wide-band signals $\boldsymbol{y}$, and then generate input log-magnitude spectrograms $\boldsymbol{X}$ for the frequency-domain neural networks as described in Sec. 2.1. For the short-term Fourier transform, we use Hanning windows of 32 ms, an FFT length of 32 ms, and a hop size of 16 ms.

Automatic speech recognition. To verify the automatic speech recognition (ASR) performance of the proposed method, we trained ASR systems using data from the LibriSpeech corpus [28], and tested on the Valentini-Botinhao dataset. LibriSpeech is an open-source corpus containing 960 hours of speech derived from audiobooks in the LibriVox project.

3.2. Setup of Speech Bandwidth Extension

The baseline systems used for performance comparison are a DNN system [10] which operates in the frequency domain, a Temporal Feature-Wise Linear Modulation (TFiLM) system [29] and a Wave-U-Net system [19] which operate in the time domain.

DNN: We re-implemented the DNN-based method proposed in [10]. The DNN takes the log-spectrum of the narrow-band signal as input, and the output is the high-frequency log-spectrogram of the wide-band signal. The model has 3 hidden layers, each having 2,048 nodes.

TFiLM: The model has an encoder-decoder architecture including downsampling and upsampling convolutional blocks. In each block, there is a core TFiLM layer, which captures contextual information in sequential inputs by combining elements of convolutional and recurrent approaches.

Wave-U-Net: We use the default parameters in [19]. The default Wave-U-Net includes 12 DS and 12 US blocks.

Below are the details of our proposed model:

Wave-U-Net (SS): We implemented a light version of Wave-U-Net as the second stage (SS) of our proposed model. It contains 6 DS and 6 US blocks, and has only about 15% of the number of parameters of the original Wave-U-Net. We computed the MSTFT loss with three configurations of STFT: window length (240, 600, 1200) samples, FFT length (512, 1024, 2048) samples, hop size (50, 120, 240) samples. The hyperparameter λ in Eq. 3 was set to 10 empirically.

CRN: The CRN consists of 6 Conv2d blocks. The number of output channels is [16, 32, 64, 128, 128, 128] for each layer, and their filter sizes are all 3×3. This is followed by two LSTM layers, each having 640 nodes. The LSTM layers are followed by 6 Deconv2d blocks with [128, 128, 64, 32, 16, 16] output channels and filter size 3×3, and an output layer with a single output channel and filter size 1×1.

TCN: The TCN takes $F = 129$ low-frequency bins as input and predicts $F' = 128$ high-frequency bins. The bottleneck and hidden feature sizes are set to $B = 128$ and $H = 256$. The trunk consists of $R = 3$ stacks of $L = 6$ TCN blocks. The D-conv layers have a kernel size of $P = 3$.

All the proposed models were trained using the Adam optimizer [30] with an initial learning rate of 0.0002. The training speech was cut into segments of 16,384 samples, and each mini-batch contained 32 such segments. The CRN and TCN were trained for 200 epochs; Wave-U-Net (SS) was trained for 500 epochs. The baseline models are implemented by following the setup in their original papers.

3.3. Setup of ASR

The automatic speech recognition (ASR) experiments use a time-delay neural network–hidden Markov model (TDNN-HMM) hybrid chain model [31]. This acoustic model is trained using the Kaldi toolkit [32] with the standard recipe[1]. We trained two versions of acoustic models with different data. One was trained using all the 960 hours of original 16 kHz speech in the LibriSpeech corpus, and we refer to it as *TDNN-Default*. The other was trained in a multi-style fashion, using 860 hours of original LibriSpeech data plus 100 hours of speech downsampled and recovered by our proposed system. We refer to this model as *TDNN-MTR*. We measured the word error rate (WER) on the test set (824 sentences) of the Valentini-Botinhao corpus [20], also downsampled and processed by our proposed system.

3.4. Evaluation Metrics

For speech bandwidth extension, we use the following metrics which are often used to evaluate speech enhancement: signal-to-noise ratio (SNR), log-spectral distance (LSD), and the wide-band perceptual evaluation of speech quality (PESQ) score [33]. LSD and SNR measure the similarity of two signals in the frequency and time domains, respectively. Given the target and predicted waveforms $\boldsymbol{y}$ and $\tilde{\boldsymbol{y}}$, and their log-magnitude spectrograms $\boldsymbol{Y}$ and $\tilde{\boldsymbol{Y}}$, the LSD and SNR are calculated as follows:

$$\text{LSD} = \frac{1}{T}\sum_{t=1}^{t=T}\sqrt{\frac{1}{F}\sum_{f=1}^{F}[\tilde{\boldsymbol{Y}}(t,f) - \boldsymbol{Y}(t,f)]^2}, \tag{4}$$

$$\text{SNR} = 10\log_{10}\frac{||\boldsymbol{y}||_2^2}{||\tilde{\boldsymbol{y}} - \boldsymbol{y}||_2^2}. \tag{5}$$

The PESQ score estimates the perceived quality of a speech signal by comparing it against the reference speech, and ranges from -0.5 to 4.5. In addition, we use the word error rate (WER) to evaluate the ASR performance.

[1] https://github.com/kaldi-asr/kaldi/tree/master/egs/librispeech/s5

Table 1: *The LSD, SNR, wide-band PESQ scores, and WER of various networks. WERs are measured with the TDNN-Default acoustic model. The best value in each column is highlighted in bold. * indicates re-implementation.*

Model	#Params	Domain	LSD↓	SNR(dB)↑	WB-PESQ↑	WER(%)↓
***DNN [10]**	15.23M	frequency	1.49	22.68	3.83	14.25
***TFiLM [29]**	68.22M	time	2.35	24.54	4.00	18.66
Wave-U-Net	10.1M	time	2.10	**24.73**	3.82	18.27
Wave-U-Net (SS)	1.49M	time	1.31	23.74	3.91	13.38
CRN	7.74M	frequency	1.34	23.01	4.10	12.93
CRN+SS (Two-stage)	7.74M + 1.49M	frequency + time	**1.24**	24.02	4.07	**12.61**
TCN	1.25M	frequency	1.34	22.89	**4.11**	12.82
TCN+SS (Two-stage)	1.25M + 1.49M	frequency + time	1.26	23.71	4.08	12.67

3.5. Results

Table 1 summarizes the LSD, SNR and wide-band PESQ scores of the various systems. From Table 1, we can see that the time-domain networks TFiLM and Wave-U-Net achieve better SNR scores than the frequency-domain DNN-based approach, while underperforming on LSD metrics. This is because SNR mainly reflects the similarity between the original and reconstructed waveforms in the time domain, while LSD reflects the similarity between the original and reconstructed spectrograms.

Next, we isolate the second stage (SS) of our proposed system – a light version of Wave-U-Net trained with both time- and frequency-domain losses. Even with only 15% of the parameters of the original version, the light version improves LSD and PESQ significantly. This corroborates the finding in [16] that the time- and frequency-domain loss terms are complementary.

If we evaluate the first stage of our proposed system (CRN or TCN) alone, we see they achieve much better LSD, PESQ and WER compared with the baselines with fewer parameters. This is typical of frequency-domain methods. There is no significant difference between the performance of the CRN and the TCN, even though the latter has only 16% of the number parameters of the former. This illustrates the power of the large receptive field brought by the exponentially increasing dilation factors. The CRN or TCN alone, however, fails to match the SNR of the time-domain baselines. This can be made up by the addition of the second stage: not only does it increase the SNR by about 1 dB, but it also further reduces the LSD. This demonstrates that a time-domain second stage can improve the reconstructed signal by refining its temporal details.

Table 1 also lists the WER of the *TDNN-Default* acoustic model measured on speech signals reconstructed by the various systems. The oracle WER, obtained by evaluating *TDNN-Default* on the original test signals , is 11.19%. We find the WER to be highly correlated with the LSD, but less correlated with PESQ and SNR. In other words, a more faithful reconstruction of the log-magnitude spectrogram leads to better ASR performance, but better perceived quality of the waveforms may not. The best WER (12.61%) is achieved by the two-stage system CRN+SS. In addition, we also investigate the *TDNN-MTR* acoustic model, which is trained in a multi-style fashion using some reconstructed data. Multi-style training is known to reduce the mismatch between training and inference. From the results in Table 2, we can see that multi-style training is still helpful for speech recognition even with BWE in place.

We would like to emphasize that, while achieving good performance in all metrics, our proposed model is very light-weight compared to the baseline systems. For example, the TCN+SS system has only about 18%, 4% and 27% of the number of parameters of DNN, TFiLM and Wave-U-Net, respectively.

Table 2: *Word error rates achieved by combining multi-style training with bandwidth extension.*

Acoustic Model	CRN+SS	TCN+SS
TDNN-Default	12.61	12.67
TDNN-MTR	11.14	11.29

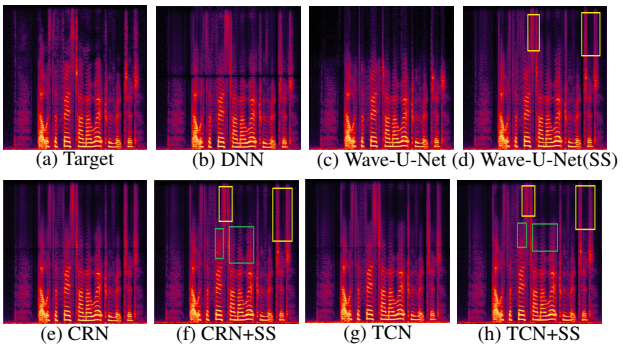


Figure 3: *Log-magnitude spectrograms of predicted wide-band signal by various approaches.*

Finally, in Fig. 3, we visualize the log-magnitude spectrograms of the signals reconstructed by the various systems. Comparing the parts highlighted with yellow boxes in (d, f, h) with (c), we can see that the introduction of the MSTFT loss ensures a sufficient recovery of the high-frequency content of the signal. Comparing the parts hightlighted with green boxes in (f, h) with (e, g), we can see that the second stage eliminates some artifacts caused by using an artificial phase spectrogram.

4. Conclusion

In this paper, we have presented a two-stage approach for speech bandwidth extension (BWE) which combines the advantages of both frequency- and time-domain methods. The first stage of the proposed system is a frequency-domain CRN or TCN which recovers the high-frequency log-magnitude spectrogram; the second stage is a time-domain Wave-U-Net which refines the temporal details of the reconstructed signal. We have found that the TCN performs equally well as the CRN without recurrent layers thanks to the large receptive field created by exponentially increasing dilation factors. We have also found it essential to use loss functions in both the time and the frequency domains during training. Our proposed system achieves better LSD, PESQ and WER metrics and a competitive SNR compared with baseline systems, while having significantly fewer parameters. We have also investigated multi-style training, and found it still helpful for speech recognition even with BWE in place.

5. References

[1] S. Chennoukh, A. Gerrits, G. Miet, and R. Sluijter, "Speech enhancement via frequency bandwidth extension using line spectral frequencies," in *ICASSP*, vol. 1. IEEE, 2001, pp. 665–668.

[2] K. Nakamura, K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda, "A mel-cepstral analysis technique restoring high frequency components from low-sampling-rate speech," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[3] P. S. Nidadavolu, C.-I. Lai, J. Villalba, and N. Dehak, "Investigation on bandwidth extension for speaker recognition," in *INTERSPEECH*, 2018, pp. 1111–1115.

[4] K.-Y. Park and H. S. Kim, "Narrowband to wideband conversion of speech using GMM based transformation," in *ICASSP*, vol. 3. IEEE, 2000, pp. 1843–1846.

[5] H. Seo, H.-G. Kang, and F. Soong, "A maximum a posterior-based reconstruction approach to speech bandwidth expansion in noise," in *ICASSP*. IEEE, 2014, pp. 6087–6091.

[6] P. Jax and P. Vary, "Artificial bandwidth extension of speech signals using MMSE estimation based on a hidden Markov model," in *ICASSP*, vol. 1. IEEE, 2003, pp. I–681–I–683.

[7] G. Chen and V. Parsa, "HMM-based frequency bandwidth extension for speech enhancement using line spectral frequencies," in *ICASSP*, vol. 1. IEEE, 2004, pp. I–709–I–712.

[8] Z.-H. Ling, S.-Y. Kang, H. Zen, A. Senior, M. Schuster, X.-J. Qian, H. M. Meng, and L. Deng, "Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and future trends," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 35–52, 2015.

[9] B. Liu, J. Tao, Z. Wen, Y. Li, and D. Bukhari, "A novel method of artificial bandwidth extension using deep architecture," in *INTERSPEECH*, 2015.

[10] K. Li and C.-H. Lee, "A deep neural network approach to speech bandwidth expansion," in *ICASSP*. IEEE, 2015, pp. 4395–4399.

[11] K. Li, Z. Huang, Y. Xu, and C.-H. Lee, "DNN-based speech bandwidth expansion and its application to adding high-frequency missing features for automatic speech recognition of narrowband speech," in *INTERSPEECH*, 2015.

[12] Y. Gu, Z.-H. Ling, and L.-R. Dai, "Speech bandwidth extension using bottleneck features and deep recurrent neural networks," in *INTERSPEECH*, 2016, pp. 297–301.

[13] Z.-H. Ling, Y. Ai, Y. Gu, and L.-R. Dai, "Waveform modeling and generation using hierarchical recurrent neural networks for speech bandwidth extension," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 5, pp. 883–894, 2018.

[14] V. Kuleshov, S. Z. Enam, and S. Ermon, "Audio super-resolution using neural nets," in *ICLR (Workshop Track)*, 2017.

[15] T. Y. Lim, R. A. Yeh, Y. Xu, M. N. Do, and M. Hasegawa-Johnson, "Time-frequency networks for audio super-resolution," in *ICASSP*. IEEE, 2018, pp. 646–650.

[16] H. Wang and D. Wang, "Time-frequency loss for CNN based speech super-resolution," in *ICASSP*. IEEE, 2020, pp. 861–865.

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

[18] 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.

[19] 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.

[20] 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.

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

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *IEEE International Conference on Computer Vision*, 2015, pp. 1026–1034.

[23] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing ideal time-frequency magnitude masking for speech separation," *IEEE/ACM Trans. Audio, Speech, Language Process*, vol. 27, no. 8, pp. 1256–1266, Aug. 2019.

[24] 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.

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

[26] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical evaluation of rectified activations in convolutional network," *arXiv preprint arXiv:1505.00853*, 2015.

[27] A. Defossez, G. Synnaeve, and Y. Adi, "Real time speech enhancement in the waveform domain," *arXiv preprint arXiv:2006.12847*, 2020.

[28] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *ICASSP*. IEEE, 2015, pp. 5206–5210.

[29] S. Birnbaum, V. Kuleshov, Z. Enam, P. W. Koh, and S. Ermon, "Temporal film: Capturing long-range sequence dependencies with feature-wise modulations," *arXiv preprint arXiv:1909.06628*, 2019.

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

[31] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *INTERSPEECH*, 2015.

[32] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The Kaldi speech recognition toolkit," in *Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE Signal Processing Society, 2011.

[33] I.-T. Recommendation, "Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," *Rec. ITU-T P. 862*, 2001.