Speech Denoising With Deep Feature Losses

François G. Germain[1]*, Qifeng Chen*[2]*, and Vladlen Koltun*[3]

[1]CCRMA, Stanford University, Stanford, CA, USA
[2]Department of Computer Science and Engineering, HKUST, Kowloon, Hong Kong
[3]Intelligent Systems Lab, Intel Labs, Santa Clara, CA, USA

francois@ccrma.stanford.edu, cqf@ust.hk, vladlen.koltun@intel.com

Abstract

We present an end-to-end deep learning approach to denoising speech signals by processing the raw waveform directly. Given input audio containing speech corrupted by an additive background signal, the system aims to produce a processed signal that contains only the speech content. Recent approaches have shown promising results using various deep network architectures. In this paper, we propose to train a fully-convolutional context aggregation network using a deep feature loss. That loss is based on comparing the internal feature activations in a different network, trained for audio classification. Our approach outperforms the state of the art in objective speech quality metrics and in large-scale perceptual experiments with human listeners. It also outperforms an identical network trained using traditional regression losses. The advantage of the new approach is particularly pronounced for the hardest data with the most intrusive background noise, for which denoising is most needed and most challenging.

Index Terms: Speech denoising, speech enhancement, deep learning, context aggregation network, deep feature loss

1. Introduction

Speech denoising (or enhancement) refers to the removal of background content from speech signals [1]. Due to the ubiquity of this audio degradation, denoising has a key role in improving human-to-human (e.g., hearing aids) and human-to-machine (e.g., automatic speech recognition) communication. A particularly challenging but common problem is the underdetermined case of single-channel speech denoising, due to the complexity of speech processes and the unknown nature of the non-speech material. The complexity is further compounded by the nature of audio data which contains a high density of data samples (e.g., 16,000 samples per second). Challenges also arise in mediated human-to-human communication, as perception mechanisms can make small errors still noticeable by the average user [2]. Earlier denoising systems relied on spectrogram-domain signal processing methods [1], followed more recently by methods based on spectrogram factorization [3]. Current denoising pipelines instead rely on deep networks for state-of-the-art performance. However, most pipelines still operate in the spectrogram domain [4, 5, 6, 7, 8, 9, 10, 11]. With such solutions, signal artifacts can arise due to time aliasing when using the inverse short-time Fourier transform to produce the time-domain enhanced signal. This issue can be alleviated somewhat, but with increased computational cost and system complexity [12, 13, 14, 15, 16, 17, 18].

Increasingly, methods optimized end-to-end and operating directly on the raw waveform have achieved state-of-the-art results in speech processing [19, 20] including speech denoising [21, 22, 23, 24]. Such approaches aim at fully leveraging the expressive power of deep networks while avoiding expensive time-frequency transformations or loss of phase information. These approaches typically use simple regression losses for training the network [21, 22] (e.g., L^1 loss on the raw waveform); pipelines with more advanced loss functions have shown limited gains in mismatched conditions [23, 24]. In this work, we present an end-to-end deep learning approach to speech denoising. Our approach trains a fully-convolutional denoising network using a deep feature loss. This loss function is inspired by computer vision research, where feature activation patterns in pretrained classification networks were found to yield effective loss functions for complex computer vision tasks (e.g., image stylization, synthesis and reflection separation [25, 26, 27]). To compute the loss between two images, these approaches apply a pretrained general-purpose image classification network to both. Each image induces a pattern of internal activations in the network to be compared, and the loss is defined in terms of their dissimilarity. Such training losses have been shown to yield state-of-the-art results without the need for prior expert knowledge or added complexity for the processing network itself. In particular, increased performance can be achieved without specializing the loss networks to the target task [28]. Our work develops this idea in the context of speech processing.

To compute our deep feature loss between two audio waveforms, we apply a pretrained general-purpose audio classification network to each waveform and compare the internal activation patterns they induce in the network. This compares a multitude of features at different scales in the two waveforms. We perform extensive experiments that compare the presented approach to recent state-of-the-art end-to-end deep learning techniques for denoising. Our approach outperforms them in both objective speech quality metrics and large-scale perceptual experiments with human listeners. The advantages of the presented approach are particularly pronounced for the hardest, noisiest inputs, for which denoising is most challenging. Our code, trained model and audio examples are available online.

2. Method

2.1. Denoising Network

Let $\mathbf{x}$ be an audio signal corresponding to speech $\mathbf{s}$ that is corrupted by an additive background signal $\mathbf{n}$ so that $\mathbf{x} = \mathbf{s} + \mathbf{n}$. Our goal is to find a denoising operator g such that $g(\mathbf{x}) \approx \mathbf{s}$. To do so, we train a fully-convolutional network architecture based on context aggregation networks [29] as shown in Figure 1. In our preliminary experiments, we found no advantage with inserting time-frequency transforms in the pipeline so our network directly processes and outputs audio waveforms. The output signal is synthesized sample by sample as we slide the

This work was performed while F. Germain was interning at Intel.

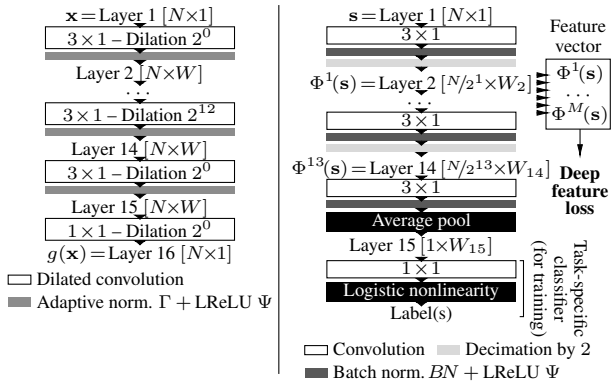

Figure 1: *Architectures of the denoising network (left) and the classification/feature loss network (right) used to form the feature vector fed to the deep feature loss function.*

network along the input. Context aggregation networks have been previously used in the WaveNet architecture for speech synthesis [19]. Our architecture is simpler – no skip connections across layers, no conditioning, no gated activations – while our loss function is more advanced (see Section 2.2).

Our context aggregation network consists of a stack of 16 convolutional layers. Each intermediary layer (i.e., layers 2 through 15) is computed from the previous one via a dilated convolution with 3×1 learned kernels [29] followed by an adaptive normalization (see below) and a pointwise leaky rectified linear unit (LReLU) [30] $\Psi(x) = \max(0.2x, x)$. The dilation operator aggregates long-range contextual information without changing sampling frequency across layers [29, 19]. We increase the dilation factor exponentially with depth from 2^0 for layer 2 to 2^{12} for layer 14. The factor for layer 15 is 2^0. Layer 16 is computed from layer 15 via an affine transformation (i.e., a convolution with 1×1 learned kernels plus bias). Layers 1 and 16 respectively correspond to the degraded input signal and the enhanced output signal, and have width 1 as they correspond to monochannel audio waveforms. The intermediary layers have identical width (i.e., number of feature maps) $W = 64$. All convolutions use zero-padding so that all layer lengths are equal to the input signal length N (i.e., its number of samples). As a consequence, our network is trained to handle the beginning and end of audio files even when speech content is near the sequence boundary. Due to its fully convolutional structure, the input signal length can vary and does not need to be known in advance. Only the signal sampling frequency f_s is known (here 16 kHz). The receptive field of the entire network is $2^{14} + 1$ samples, i.e., about 1 s of audio. We thus expect the system to capture context on the time scales of spoken words. A similar network architecture was also shown to be advantageous in terms of compactness and runtime for image processing [31].

The adaptive normalization operator Γ in our network matches the one proposed in [31] and improves performance and training speed. It adaptively combines batch normalization and identity mapping of the input x as the weighted sum $\Gamma(x) = \alpha_k x + \beta_k BN(x)$ (where $\alpha_k, \beta_k \in \mathbb{R}$ are learned scalar weights for the k-th layer and BN is the batch normalization operator [32]). The weights α_k, β_k are learned by backpropagation as network parameters.

2.2. Feature Loss

In our experiments, simple training losses (e.g., L^1) led to noticeably degraded output quality at lower signal-to-noise ratios (SNRs). The network appeared to improperly process low-energy speech information of perceptual importance. Instead, we train the denoising network using a deep feature loss that penalizes differences in the internal activations of a pretrained deep network that is applied to the signals being compared. By the nature of layered networks, feature activations at different depths in the loss network correspond to different time scales in the signal. Penalizing differences in these activations thus compares many features at different audio scales.

In computer vision, there are standard well-established classification networks such as VGG-19 [33], pretrained on standard classification datasets such as ImageNet [34]. Such standard general-purpose classification networks do not exist in the audio processing field yet, so we design and train our own feature loss network. As feature loss network, we use a simple audio classification network (see Figure 1) inspired by the VGG architecture used in computer vision [33], since it is known as a particularly effective feature loss architecture [28]. The network consists of a stack of 15 convolutional layers. Each feature layer (i.e., layers 2 through 15) is computed from the previous layer via a convolution with 3×1 learned kernels, followed by batch normalization, a pointwise LReLU, and decimation by 2. Layer 15 uses average-pooling instead of decimation. Layer 1 corresponds to the input signal with length equal to the signal length N. All convolutions use zero-padding so that all other layers are half the length of the previous layer, except for layer 15 (the output vector) whose length is 1. Layer 1 has width 1 while layer l (for $l = 2 \ldots 15$) has width $W_l = 32 \times 2^{\lfloor \frac{l-2}{5} \rfloor}$ (i.e., starting at 32 feature maps for layer 2 and doubling every 5 layers). We train the network using backpropagation by feeding its output vector as features for logistic classifiers with a cross-entropy loss assigned to one or more classification tasks of interest. The classifier parameters are task-specific, while the loss network parameters are shared.

We can then form the denoising loss function from that network. Let Φ^m be the m-th feature layer of the pretrained feature loss network, with layers at different depths corresponding to features with various time resolutions. The loss function is defined as a weighted L^1 loss on the difference between the feature activations induced in the M lower feature layers of the network by the clean reference signal $\mathbf{s}$ and the output $g(\mathbf{x})$ of the denoising network being trained:

$$\mathcal{L}_{\mathbf{s},\mathbf{x}}(\theta) = \sum_{m=1}^{M} \lambda_m \left\| \Phi^m(\mathbf{s}) - \Phi^m(g(\mathbf{x};\theta)) \right\|_1, \qquad (1)$$

where θ are the parameters of the denoising network. The weights λ_m are set to balance the contribution of each layer to the loss. λ_m is set as the inverse of the quantity $\left\| \Phi^m(\mathbf{s}) - \Phi^m(g(\mathbf{x};\theta)) \right\|_1$ as calculated after 10 training epochs. (For these first 10 epochs, the weights are set to 1.)

3. Training

3.1. Feature Loss

To generate a general-purpose feature loss network, we train it jointly on multiple audio classification tasks. (Again, only the logistic classifier parameters are task-specific.) We use two tasks from the DCASE 2016 challenge [35]: the acoustic scene classification task and the domestic audio tagging task. In the first task, we are provided with audio files featuring various scenes (e.g., beach); the goal is to determine the scene type for each file. In the second task, we are given audio files featuring events of interest (e.g., child speaking); the goal is to determine which events took place in each file (with possibly multiple events in one file). Both tasks involve mixes of speech

and natural background, and the second task involves several speech-specific labels. While we use only these two tasks in the context of this paper, our flexible architecture allows for the future addition of more classification tasks (e.g., Audioset [36]).

For the scene classification task, the training set [37] consists of 30-second-long audio files sampled at 44.1kHz, split among 15 different scenes (i.e., classes). As we need to develop a feature loss for the reduced sampling frequency of 16 kHz, we resample the data. The audio files are stereo, so we split them into two mono files. The resulting training set contains 2,340 files (19.5 h). For the tagging task, the training set *CHiME-Home-refine* [38] consists of 4-second-long mono audio files sampled at 16 kHz, with 7 different tags (i.e., labels). The training set contains 1,946 files (2.16 h).

Network weights are initialized using the Xavier initialization [39] and network biases are initialized to zero. We use the Adam optimizer [40] with learning rate 10^{-4}. The network is trained for 2,500 epochs. In each epoch, we iterate over the training data for each task, alternating between files from each task. The order of the files is randomized independently for each epoch. The dataset for the first task is larger than the one for the second task, so we present some of the files in the second dataset (chosen at random) a second time to preserve strict alternation between tasks. As a result, 1 epoch consists of 4,680 iterations (1 file per iteration). As data augmentation procedure, we do not present the entire clip, but instead present a continuous section of length at least 2^{15} samples (i.e., the network receptive field), culled at random for each iteration.

3.2. Speech Denoising

We use the noisy dataset from [41]. To our knowledge, it is the largest available dataset for denoising that provides pre-mixed data with a clearly documented mixing procedure, and with the ground truth signal needed for evaluation (see Section 4.3). It also has the benefit of being the dataset used in 2 recent works that we use as baselines. All details concerning the data can be found in [41]. The training set is generated from the speech data of 28 speakers (14 male/14 female) and the background data of 10 unique background types. Each noise segment is used to generate four files with 0, 5, 10, and 15 dB SNR. The original files are sampled at 48 kHz and normalized so that the clean speech files have a maximum amplitude of 0.5. We resample them to 16 kHz. The full dataset comprises 11,572 files (9.39 h).

Network weights are initialized using the Xavier initialization and network biases are initialized to zero. The adaptive normalization parameters are initialized at $\alpha_k = 1$ and $\beta_k = 0$. The feature loss is computed using the first $M = 6$ layers. We use the Adam optimizer with learning rate 10^{-4}. The network is trained for 320 epochs (80 h) on a Titan X GPU. In each epoch, we present the entire dataset in randomized order (1 file per iteration) and files are presented in their entirety.

4. Experimental Setup

4.1. Baselines

As baselines, we use a Wiener filtering pipeline with a priori noise SNR estimation (as implemented in [42]), and two recent state-of-the-art methods that use deep networks to perform end-to-end denoising directly on the raw waveform: the Speech Enhancement Generative Adversarial Network (SEGAN) [23] and a WaveNet-based network [21]. The former is designed around an encoder-decoder processing architecture, and uses a discriminator network identical to the encoder for adversarial training.

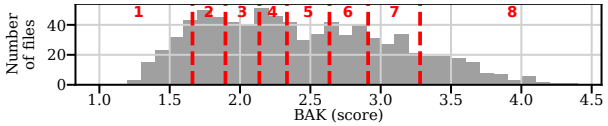

Figure 2: *Distribution of the test set in terms of composite background score. The test set was partitioned into 8 tranches, demarcated by red dashed lines and labeled with red numbers.*

The latter is designed around minor modifications to the architecture in [19]. It uses stacked context aggregation modules with gated activation units, skip connections, and a conditioning mechanism. The modifications include training with a regression loss (L^1 on the raw waveform) rather than a classification loss. The number of layers is larger than in our network (30), while the receptive field is smaller ($3 \cdot 2^{11}$ samples), capturing contextual information on more limited time scales. The network architecture is also distinctly more complex than ours. For both deep learning baselines, we use the code and models published and optimized by their respective authors. To the best of our knowledge, they also present the distinct advantage of being 2 published state-of-the-art approaches with publicly available implementations and pretrained models on the exact same training dataset as ours, which is essential for a fair comparison.

4.2. Data

All our testing is done in mismatched conditions. The data source is the same as in Section 3.2. The speech is obtained from 2 speakers (1 male/1 female). The background data is obtained from 5 distinct background types. Neither the speakers nor the backgrounds used at test time were seen during training. Each background segment is used to generate four files with 2.5, 7.5, 12.5, and 17.5 dB SNR. The full test set comprises 824 files (0.58 h). Our pipeline needs about 12 ms to process every 1 s of audio in our configuration.

4.3. Quantitative Measures

To evaluate each system, we compare its output to the ground-truth speech signal (i.e., the clean speech alone) as done in [21] and [23]. The common metrics to measure speech quality given ground truth are compared in [1]. We use here the composite scores from [42] that were found to be best correlated with human listener ratings. These consist of the overall (OVL), the signal (SIG), and the background (BAK) scores, each on a scale from 1.0 to 5.0, and corresponding respectively to the measure of overall signal quality, the measure of quality when considering speech signal degradation alone, and the measure of quality when considering background signal intrusiveness alone [43]. We also report the SNR [44], as a raw measure of the relative energies of the residual background and the speech in a given signal, quantified in decibel (dB). We use the implementations in [1]. For all metrics, higher scores denote better performance.

The test dataset is divided into 4 mixing SNR subgroups (see Section 4.2). We argue that the dataset should be rather considered as a continuous distribution of degradation, since

Table 1: *Performance for different approaches according to objective quality measures. (Higher is better.)*

	SNR (dB)	SIG (score)	BAK (score)	OVL (score)
Noisy	8.45	3.34	2.44	2.63
Wiener	12.28	3.23	2.68	2.67
SEGAN	14.82	3.21	2.76	2.56
WaveNet	18.18	2.87	3.08	2.43
Ours	**19.00**	**3.86**	**3.33**	**3.22**

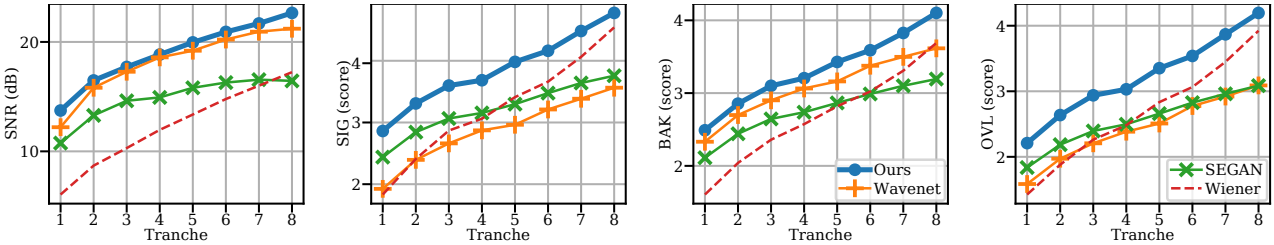

Figure 3: *Performance of different denoising approaches according to 4 objective quality measures (SNR, SIG, BAK, and OVL), plotted for each tranche in the test set (as indexed in Figure 2). For all measures, higher is better.*

Table 2: *Training the same network with different loss functions. For all metrics, higher is better.*

	SNR (dB)	SIG (score)	BAK (score)	OVL (score)
Noisy	8.45	3.34	2.44	2.63
L2	18.46	3.70	3.21	3.07
L1	18.98	3.75	3.27	3.11
Feature loss	**19.00**	**3.86**	**3.33**	**3.22**

SNR correlates poorly with human perception of the degradation level [1]. The continuum of degradation levels is better represented in the distribution of the background intrusiveness BAK score. (The SIG score is less informative since the undistorted speech signal is added.) To evaluate performance as a function of input degradation magnitude, we partition the test set into 8 tranches of equal size, corresponding to the 8 octiles of the BAK score distribution as shown in Figure 2, with each tranche representing a different denoising difficulty.

Table 1 reports these metrics for our approach and the baselines, evaluated over the test set. Our method outperforms all the baselines according to all measures by a comfortable margin. The plots in Figure 3 further show that our network yields the best quality for all levels of background intrusiveness separated in tranches, with a particularly significant margin according to perceptually-motivated composite measures. Table 2 shows the benefit of using a feature loss compared to training the same denoising network, by the same procedure on the same data, using an L^1 or an L^2 loss. Training with a feature loss outperforms networks trained with other losses. In particular, while an L^1 loss achieves a similar SNR score as our feature loss, the feature loss shows definite improvement for the BAK and OVL metrics. It also scores well for the SIG metric, especially in the noisier tranches, demonstrating the ability to capture meaningful features when important cues are hidden in the noise. Informal signal analysis suggests that our approach is better for high frequencies and around wideband speech components.

4.4. Perceptual Experiments

Objective metrics are known to only partially correlate with human audio quality ratings [1]. Hence, we also conduct carefully designed perceptual experiments with human listeners. The procedure is based on A/B tests deployed at scale on the Amazon Mechanical Turk platform. The A/B tests are grouped into Human Intelligence Tasks (HITs). Each HIT consists of 100 "ours vs baseline" pairwise comparisons. Each comparison presents two audio clips that can be played in any order by the worker, any number of times. One of the clips is the output of our approach and one is the output of one of the baselines, for the same input from the test set. The files are presented in random order (both within each pair and among pairs), so the worker is given no information as to the provenance of the clips. The worker is asked to select, within each pair, the clip with the cleaner speech. Each HIT also includes 10 additional 'sentry' comparisons in which the right answer is obvious. These sentry pairs are mixed into the HIT in random order. If a worker gives an incorrect answer to two or more sentry pairs, the entire HIT is discarded. Each HIT then contains a total of 110 pairwise comparisons. A worker is given 1 hour to complete a HIT. Each HIT is completed by 10 distinct workers.

Table 3: *Results of perceptual experiments. Each cell lists the fraction of blind randomized pairwise comparisons in which the listener rated the output of our approach as cleaner than the output of a baseline. Each row lists results for a specific baseline. Each column list results for a tranche of the test set. (Chance at 50%, higher is better.)*

Tranche:	1 (hard)	3 (medium)	5 (easy)	7 (very easy)
Ours > Wiener	96.1%	89.4%	81.7%	90.2%
Ours > SEGAN	83.5%	70.5%	64.1%	61.4%
Ours > WaveNet	83.9%	67.0%	61.4%	55.8%

The results are shown in Table 3. This table presents the fraction of blind A/B comparisons in which the listener rated a clip denoised by our network as cleaner than the same clip denoised by a baseline. The preference rates are presented for tranches 1, 3, 5, and 7 (see Figure 2). The most notable results are for the hardest/noisiest tranche, where the output of our approach was rated cleaner than the output of recent state-of-the-art deep networks in more than 83% of the comparisons. This demonstrates that our algorithm is more robust in this regime, in which degradation from the background signal is much more noticeable, and for which denoising is particularly useful. For easier tranches (i.e., lower levels of degradation in the input), both our method and the baselines generally perform satisfactorily and listeners can experience more difficulty distinguishing between the different processed files, but the preference rate for our approach remains well above chance (50%), at statistically significant levels, for all baselines across all tranches.

5. Conclusion

We presented an end-to-end speech denoising pipeline that uses a fully-convolutional network, trained using a deep feature loss network pretrained on generic audio classification tasks. This approach allows the denoising system to capture speech structure at various scales and achieve better denoising performance without added complexity in the system itself or expert knowledge in the loss design. Experiments demonstrate that our approach significantly outperforms recent state-of-the-art baselines according to objective speech quality measures as well as large-scale perceptual experiments with human listeners. In particular, the presented approach is shown to perform much better in the noisiest conditions where speech denoising is most challenging. Our paper validates the combined use of convolutional context aggregation networks and feature losses to achieve state-of-the-art speech denoising performance.

6. References

[1] P. C. Loizou, *Speech Enhancement: Theory and Practice*, 2nd ed. CRC Press, 2013.

[2] M. Bosi and R. E. Goldberg, *Introduction to Digital Audio Coding and Standards*. Springer, 2002.

[3] P. Smaragdis, C. Fevotte, G. J. Mysore, N. Mohammadiha, and M. Hoffman, "Static and dynamic source separation using nonnegative factorizations: A unified view," *IEEE Signal Process. Mag.*, vol. 31, no. 3, 2014.

[4] Y. Wang and D. Wang, "Cocktail party processing via structured prediction," in *Neural Inf. Process. Sys. (NeurIPS)*, 2012.

[5] X. Lu, Y. Tsao, S. Matsuda, , and C. Hori, "Speech enhancement based on deep denoising autoencoder," in *Interspeech*, 2013.

[6] A. Narayanan and D. Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2013.

[7] F. Weninger, J. R. Hershey, J. Le Roux, and B. Schuller, "Discriminatively trained recurrent neural networks for single-channel speech separation," in *IEEE Glob. Conf. Signal Inf. Process.*, 2014.

[8] 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, Lang. Process.*, vol. 23, no. 1, 2015.

[9] A. Kumar and D. Florencio, "Speech enhancement in multiple-noise conditions using deep neural networks," in *Interspeech*, 2016.

[10] X.-L. Zhang and D. Wang, "A deep ensemble learning method for monaural speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 5, 2016.

[11] J. Chen and D. Wang, "Long short-term memory for speaker generalization in supervised speech separation," *J. Acoust. Soc. Am.*, vol. 141, no. 6, 2017.

[12] J. Le Roux and E. Vincent, "Consistent Wiener filtering for audio source separation," *IEEE Signal Process. Lett.*, vol. 20, no. 3, 2013.

[13] F. G. Germain, G. J. Mysore, and T. Fujioka, "Equalization matching of speech recordings in real-world environments," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2016.

[14] T. Gerkmann, M. Krawczyk-Becker, and J. Le Roux, "Phase processing for single-channel speech enhancement: History and recent advances," *IEEE Signal Process. Mag.*, vol. 32, no. 2, 2015.

[15] Y. Wang and D. Wang, "A deep neural network for time-domain signal reconstruction," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2015.

[16] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2015.

[17] D. S. Williamson and D. Wang, "Time-frequency masking in the complex domain for speech dereverberation and denoising," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 7, 2017.

[18] J. A. Moorer, "A note on the implementation of audio processing by short-term Fourier transform," in *IEEE Workshop Appl. Signal Process. Audio Acoust. (WASPAA)*, 2017.

[19] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," *arXiv:1609.03499*, 2016.

[20] Y. Luo and N. Mesgarani, "TaSNet: Time-domain audio separation network for real-time, single-channel speech separation," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2018.

[21] D. Rethage, J. Pons, and X. Serra, "A WaveNet for speech denoising," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2018.

[22] S.-W. Fu, Y. Tsao, X. Lu, and H. Kawai, "Raw waveform-based speech enhancement by fully convolutional networks," in *Asia-Pac. Signal Inf. Process. Assoc. Annu. Summit Conf.*, 2017.

[23] S. Pascual, A. Bonafonte, and J. Serrà, "SEGAN: Speech enhancement generative adversarial network," in *Interspeech*, 2017.

[24] K. Qian, Y. Zhang, S. Chang, X. Yang, D. Florencio, and M. Hasegawa-Johnson, "Speech enhancement using Bayesian WaveNet," in *Interspeech*, 2017.

[25] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Eur. Conf. Comput. Vis. (ECCV)*, 2016.

[26] Q. Chen and V. Koltun, "Photographic image synthesis with cascaded refinement networks," in *Int. Conf. Comput. Vis. (ICCV)*, 2017.

[27] X. Zhang, R. Ng, and Q. Chen, "Single image reflection separation with perceptual losses," in *Comput. Vis. Pattern Recognit. (CVPR)*, 2018.

[28] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Comput. Vis. Pattern Recognit. (CVPR)*, 2018.

[29] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Int. Conf. Learn. Represent. (ICLR)*, 2016.

[30] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *ICML Workshop Deep Learn. Audio, Speech, Lang. Process.*, 2013.

[31] Q. Chen, J. Xu, and V. Koltun, "Fast image processing with fully-convolutional networks," in *Int. Conf. Comput. Vis. (ICCV)*, 2017.

[32] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Int. Conf. on Mach. Learn. (ICML)*, 2015.

[33] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Int. Conf. Learn. Represent. (ICLR)*, 2015.

[34] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and F. Li, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis. (IJCV)*, vol. 115, no. 3, 2015.

[35] A. Mesaros, T. Heittola, E. Benetos, P. Foster, M. Lagrange, T. Virtanen, and M. D. Plumbley, "Detection and classification of acoustic scenes and events: Outcome of the DCASE 2016 challenge," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 2, 2018.

[36] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2017.

[37] A. Mesaros, T. Heittola, and T. Virtanen, "TUT database for acoustic scene classification and sound event detection," in *Eur. Signal Process. Conf. (EUSIPCO)*, 2016.

[38] P. Foster, S. Sigtia, S. Krstulovic, J. Barker, and M. D. Plumbley, "CHiMe-Home: A dataset for sound source recognition in a domestic environment," in *IEEE Workshop Appl. Signal Process. Audio Acoust. (WASPAA)*, 2015.

[39] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Int. Conf. AI Stat. (AISTATS)*, 2010.

[40] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Int. Conf. Learn. Represent. (ICLR)*, 2015.

[41] C. Valentini-Botinhao, X. Wang, S. Takaki, and J. Yamagishi, "Investigating RNN-based speech enhancement methods for noise-robust text-to-speech," in *ISCA Speech Synthesis Workshop*, 2016.

[42] Y. Hu and P. C. Loizou, "Subjective comparison of speech enhancement algorithms," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2006.

[43] ITU-T, "Subjective test methodology for evaluating speech communication systems that include noise suppression algorithm," ITU-T Recomm. P.835, Tech. Rep., 2003.

[44] S. R. Quackenbush, T. P. Barnwell, and M. A. Clements, *Objective Measures of Speech Quality*. Prentice Hall, 1988.