SERGAN: SPEECH ENHANCEMENT USING RELATIVISTIC GENERATIVE ADVERSARIAL NETWORKS WITH GRADIENT PENALTY

Deepak Baby and Sarah Verhulst

Dept. of Information Technology, Ghent University, Belgium
{deepak.baby, s.verhulst}@ugent.be

ABSTRACT

Popular neural network-based speech enhancement systems operate on the magnitude spectrogram and ignore the phase mismatch between the noisy and clean speech signals. Recently, conditional generative adversarial networks (cGANs) have shown promise in addressing the phase mismatch problem by directly mapping the raw noisy speech waveform to the underlying clean speech signal. However, stabilizing and training cGAN systems is difficult and they still fall short of the performance achieved by spectral enhancement approaches. This paper introduces relativistic GANs with a relativistic cost function at its discriminator and gradient penalty to improve time-domain speech enhancement. Simulation results show that relativistic discriminators provide a more stable training of cGANs and yield a better generator network for improved speech enhancement performance.

***Index Terms*—** speech enhancement, relativistic GAN, convolutional neural networks

1. INTRODUCTION

Speech enhancement systems aim to improve the quality and intelligibility of acquired speech signals by removing artefacts caused by background noise or other interferences such as room reverberation. Recently, deep neural network (DNN)-based approaches gained success in speech enhancement due to their powerful modeling capabilities [1–5].

DNN-based systems are typically trained to estimate a time-frequency (T-F) mask in the range $[0,\ 1]$, which provides the relative amplitudes of the underlying clean speech and noise signals at every T-F point [1,6]. However, these masks only modify the magnitude spectra of the input signal and ignore the phase mismatch between the noisy and clean speech signals [1,7]. Since speech quality can be significantly improved when the clean phase spectrum is known [8], it is worthwhile exploring speech enhancement techniques which preserve phase information. To remedy this phase mismatch problem, this paper investigates the use of generative neural networks which can directly map the raw noisy speech waveform to the underlying clean speech waveform.

Recently, generative adversarial network (GAN)-based models [9] have been explored for raw speech waveform enhancement [10–14]. GAN consists of a generative model or *generator network* (G) and a *discriminator network* (D) that play a min-max game between each other. [12] demonstrated that the generator part G alone with an $L1$ loss can yield similar performance as GANs that adversarially train G to fool D. Therefore, there is a growing debate on the suitability of GANs for speech enhancement.

A part of this concern is attributed to their complex training which requires finding a Nash equilibrium of a non-convex game between G and D [9,15], and the quality of the generated samples critically depends on this achieved equilibrium. This paper investigates whether an improved discriminator could lead to a better generator to yield a *cleaner* speech signal. We introduce SERGANs: speech enhancement systems that make use of relativistic GANs (RGANs) [16]. RGANs use a relativistic loss function at the discriminator and are shown successful in image generation [16]. This paper investigates whether RGANs can yield a better generator network for speech enhancement. We also investigate the use of gradient penalty in D [17] for stabilizing such systems.

This paper evaluates and compares several relativistic GAN models such as relativistic GANs and relativistic average GANs with mean-square error and binary cross-entropy loss functions with gradient penalty in the discriminator. In addition, we also introduce Wasserstein GANs [18] for speech enhancement. Simulation results show that SERGAN models with gradient penalty improve the speech enhancement performance in addition to yielding a more stable GAN training. To the best of our knowledge, it is the first time the standard binary cross-entropy loss has been shown successful for GAN-based speech enhancement.

2. SPEECH ENHANCEMENT USING GAN

Speech enhancement systems aim to estimate the clean speech signal $\mathbf{x}$ from the noisy mixture $\mathbf{y} = \mathbf{x} + \mathbf{w}$, where $\mathbf{w}$ is the added background noise.

This work was funded with support from the EU Horizon 2020 programme under grant agreement No 678120 (RobSpear). We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

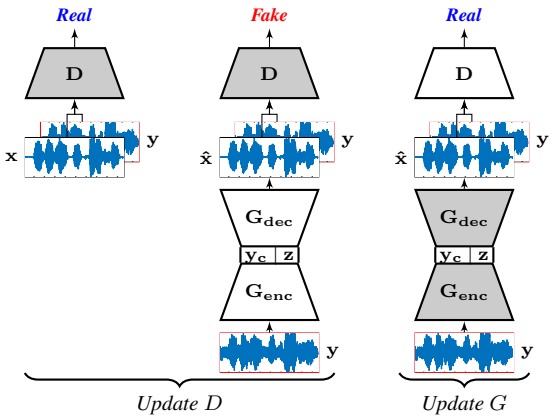

Fig. 1. *Training a conventional cGAN-based speech enhancement system. The updates for D and G are alternated over several epochs. $\mathbf{y}$, $\mathbf{x}$ and $\hat{\mathbf{x}}$ are the noisy speech, clean speech and the clean speech estimate generated by G, respectively. $\mathbf{y_c}$ is the encoder output of noisy speech and $\mathbf{z}$ are samples from the prior distribution $\mathcal{Z}$.*

In the generic GAN model, G acts as a generative model that learns to map samples $\mathbf{z}$ from some prior distribution $\mathcal{Z}$ to samples $\mathbf{x}$ that belong to a data distribution of interest $\mathcal{X}$ (i.e., the distribution of the clean speech samples, in our case). D is a binary classifier that is trained to classify samples from the true data distribution as real and the generated samples from G as fake. Since G is trained to fool D so that D classifies G's output as real, G will in turn learn to generate samples that are closer to the real data manifold. With cGANs, we direct this data generation process based on the input noisy speech $\mathbf{y}$ such that G generates an estimate that is closer to the underlying clean speech signal $\mathbf{x}$ (denoted as $\hat{\mathbf{x}} \triangleq G(\mathbf{y}, \mathbf{z})$).

The training phases of a conventional cGAN-based speech enhancement system are depicted in Fig. 1. Notice that D is conditioned using the noisy speech signal $\mathbf{y}$ and G makes use of an encoder-decoder structure. The encoder (G_{enc}) projects the input noisy signal into a condensed representation $\mathbf{y_c} = G_{\text{enc}}(\mathbf{y})$, which is concatenated with the latent samples $\mathbf{z}$. The decoder (G_{dec}) then reconstructs the signal such that its output $\hat{\mathbf{x}} = G_{\text{dec}}(\mathbf{y_c}, \mathbf{z})$ fools D into classifying it as real. As can be seen from Fig. 1, training a cGAN-based speech enhancement setting is comprised of repeating the following three updates for every mini-batch over several epochs (encoding real as 1 and fake as 0):

1. Update D such that $\mathbf{x}$ and $\mathbf{y}$ pairs are classified as real, i.e., $D(\mathbf{x}, \mathbf{y}) \rightarrow 1$
2. Update D such that the generated samples $\hat{\mathbf{x}}$ and $\mathbf{y}$ pairs are classified as fake, i.e., $D(\hat{\mathbf{x}}, \mathbf{y}) \rightarrow 0$
3. Freeze D and update G such that D classifies $\hat{\mathbf{x}}$ and $\mathbf{y}$ pairs as real, i.e., $D(\hat{\mathbf{x}}, \mathbf{y}) \rightarrow 1$

The updates for G and D depends on the output loss function. Popular loss functions for training conventional cGAN models are listed below. For brevity, we define the data-pairs as $\mathbf{x_r} \triangleq (\mathbf{x}, \mathbf{y}) \sim \mathcal{P}$ and $\mathbf{x_f} \triangleq (\hat{\mathbf{x}}, \mathbf{y}) \sim \mathcal{Q}$. Let σ be the sigmoid non-linearity and $C(\mathbf{x})$ be the discriminator network without the final sigmoid layer $\Rightarrow D(\mathbf{x}) = \sigma\left(C(\mathbf{x})\right)$.

1. Standard GAN (SGAN) [9,19]: Binary cross-entropy

$$\mathcal{L}_D = -\mathbb{E}_{\mathbf{x_r}\sim\mathcal{P}}\left[\log\left(D(\mathbf{x_r})\right)\right] - \mathbb{E}_{\mathbf{x_f}\sim\mathcal{Q}}\left[\log\left(1 - D(\mathbf{x_f})\right)\right]$$
$$\mathcal{L}_G = -\mathbb{E}_{\mathbf{x_f}\sim\mathcal{Q}}\left[D(\mathbf{x_f})\right].$$

2. Least-square GAN (LSGAN) [10,20]: Mean-squared error

$$\mathcal{L}_D = \mathbb{E}_{\mathbf{x_r}\sim\mathcal{P}}\left[\left(C(\mathbf{x_r})\right)^2\right] + \mathbb{E}_{\mathbf{x_f}\sim\mathcal{Q}}\left[\left(1 - C(\mathbf{x_f})\right)^2\right]$$
$$\mathcal{L}_G = \mathbb{E}_{\mathbf{x_f}\sim\mathcal{Q}}\left[C(\mathbf{x_f})^2\right].$$

3. Wasserstein GAN (WGAN) [18]: Wasserstein loss

$$\mathcal{L}_D = -\mathbb{E}_{\mathbf{x_r}\sim\mathcal{P}}\left[C(\mathbf{x_r})\right] + \mathbb{E}_{\mathbf{x_f}\sim\mathcal{Q}}\left[C(\mathbf{x_f})\right]$$
$$\mathcal{L}_G = -\mathbb{E}_{\mathbf{x_f}\sim\mathcal{Q}}\left[C(\mathbf{x_f})\right].$$

However, almost all prior works on GAN-based speech enhancement are based only on LSGANs since SGANs were found to be unstable [10–12], where G or D becomes more powerful and the loss function of the other diverges. In addition, WGAN for speech enhancement has not been explored yet.

3. RELATIVISTIC GAN

The relativistic GAN approach presented in [16] argues that conventional GAN training misses the key property that the probability of real data being real (i.e., $D(\mathbf{x_r})$) should decrease as the probability of fake data being real (i.e., $D(\mathbf{x_f})$) increases. However, the conventional GAN models cannot incorporate this since G does not influence $D(\mathbf{x_r})$ (ref. Fig. 1). To circumvent this, the discriminator was made relativistic by sampling the $\mathbf{x_r}$/$\mathbf{x_f}$ data-pairs and define it as $D_{rel}(\mathbf{x_r}, \mathbf{x_f}) = \sigma\left(C(\mathbf{x_r}) - C(\mathbf{x_f})\right)$ [16]. The proposed discriminator estimates the probability that the given real data is more realistic than the sampled fake data [16]. Simultaneously, the model also looks at $D'_{rel}(\mathbf{x_r}, \mathbf{x_f}) = \sigma\left(C(\mathbf{x_f}) - C(\mathbf{x_r})\right)$ which is the probability that the fake data is more realistic than the real data.

For binary cross-entropy loss, the second term in the cost function $\log\left(1 - D'_{rel}(\mathbf{x_r}, \mathbf{x_f})\right)$ can be omitted as $1 - D'_{rel}(\mathbf{x_r}, \mathbf{x_f}) = 1 - \sigma\left(C(\mathbf{x_f}) - C(\mathbf{x_r})\right) = \sigma\left(C(\mathbf{x_r}) - C(\mathbf{x_f})\right) = D_{rel}(\mathbf{x_r}, \mathbf{x_f})$. The relativistic SGAN (RSGAN) loss function thus becomes [16];

$$\mathcal{L}_D = -\mathbb{E}_{(\mathbf{x_r},\mathbf{x_f})\sim(\mathcal{P},\mathcal{Q})}\left[\log\left(\sigma\left(C(\mathbf{x_r}) - C(\mathbf{x_f})\right)\right)\right]$$
$$\mathcal{L}_G = -\mathbb{E}_{(\mathbf{x_r},\mathbf{x_f})\sim(\mathcal{P},\mathcal{Q})}\left[\log\left(\sigma\left(C(\mathbf{x_f}) - C(\mathbf{x_r})\right)\right)\right].$$

3.1. Relativistic average GAN

Although the relativistic GAN approach incorporates the ability that G influences $D(\mathbf{x_r})$, it has a high variance. Alternatively, a relativistic average GAN [16] compares the discriminator output with the average of the opposite type, i.e.,

$C(\mathbf{x_r}) - \mathbb{E}_{\mathbf{x_f}\sim\mathcal{Q}}[C(\mathbf{x_f})]$ instead of $C(\mathbf{x_r}) - C(\mathbf{x_f})$. The resulting loss functions for SGAN and LSGAN are:

1. Relativistic average SGAN (RaSGAN) [16]

$$\mathcal{L}_D = -\mathbb{E}_{\mathbf{x_r}\sim\mathcal{P}}\left[\log\left(\bar{D}_{\mathbf{x_r}}\right)\right] - \mathbb{E}_{\mathbf{x_f}\sim\mathcal{Q}}\left[\log\left(1-\bar{D}_{\mathbf{x_f}}\right)\right]$$
$$\mathcal{L}_G = -\mathbb{E}_{\mathbf{x_f}\sim\mathcal{Q}}\left[\log\left(\bar{D}_{\mathbf{x_f}}\right)\right] - \mathbb{E}_{\mathbf{x_r}\sim\mathcal{P}}\left[\log\left(1-\bar{D}_{\mathbf{x_r}}\right)\right],$$

2. Relativistic average LSGAN (RaLSGAN) [16]

$$\mathcal{L}_D = \mathbb{E}_{\mathbf{x_r}\sim\mathcal{P}}\left[\left(\bar{C}_{\mathbf{x_r}}-1\right)^2\right] + \mathbb{E}_{\mathbf{x_f}\sim\mathcal{Q}}\left[\left(\bar{C}_{\mathbf{x_f}}+1\right)^2\right]$$
$$\mathcal{L}_G = \mathbb{E}_{\mathbf{x_f}\sim\mathcal{Q}}\left[\left(\bar{C}_{\mathbf{x_f}}-1\right)^2\right] + \mathbb{E}_{\mathbf{x_r}\sim\mathcal{P}}\left[\left(\bar{C}_{\mathbf{x_r}}+1\right)^2\right],$$

with, $\bar{D}_{\mathbf{x_r}} = \sigma\left(\bar{C}_{\mathbf{x_r}}\right)$, $\bar{D}_{\mathbf{x_f}} = \sigma\left(\bar{C}_{\mathbf{x_f}}\right)$ and

$$\bar{C}_{\mathbf{x_r}} = C(\mathbf{x_r}) - \mathbb{E}_{\mathbf{x_f}\sim\mathcal{Q}}[C(\mathbf{x_f})]$$
$$\bar{C}_{\mathbf{x_f}} = C(\mathbf{x_f}) - \mathbb{E}_{\mathbf{x_r}\sim\mathcal{P}}[C(\mathbf{x_r})].$$

3.2. Additional Penalty terms

1. Gradient penalty in D: Gradient penalty regularization in D is used to avoid exploding or vanishing gradients. This cost term penalizes the model if the gradient L2 norm of the discriminator output with respect to the input moves away from its target norm value 1 [17].

$$\mathcal{L}_{GP}(D) = \mathbb{E}_{\tilde{\mathbf{x}},\mathbf{y}\sim\tilde{\mathcal{P}}}\left[\left(\|\nabla_{\tilde{\mathbf{x}},\mathbf{y}} C(\tilde{\mathbf{x}},\mathbf{y})\|_2 - 1\right)^2\right]$$

where, ϵ is sampled from a uniform distribution in $[0,\ 1]$, and $\tilde{\mathcal{P}}$ is the joint distribution of $\tilde{\mathbf{x}} = \epsilon\mathbf{x} + (1-\epsilon)\hat{\mathbf{x}}$ and $\mathbf{y}$. It is observed that gradient-penalty (GP) is required to stabilize the relativistic GAN models [16]. The discriminator loss is thus $\mathcal{L}_D + \lambda_{GP}\mathcal{L}_{GP}(D)$, where λ_{GP} is the hyper-parameter that controls the GP loss.

2. L1-loss penalty in G: Several prior works [10,11,21] showed that it is beneficial to use an additional loss term in G that minimizes the L1 distance between the generated samples $\hat{\mathbf{x}}$ and the clean examples $\mathbf{x}$. This L1 term is controlled by a new hyper-parameter λ_{L1}. The generator loss is thus $\mathcal{L}_G + \lambda_{L1}\|\hat{\mathbf{x}} - \mathbf{x}\|_1$.

4. EVALUATION SETUP

We used the dataset presented in [22] for comparing the various systems. The database is derived from the voice bank corpus [23] from which recordings from 28 speakers were chosen for the training set (11 572 utterances) and 2 for the test set (824 utterances). The training set simulates 40 different noisy scenarios with 10 different noise conditions (2 artificial and 8 from the DEMAND database [24]) at signal-to-noise ratios (SNRs) of 0, 5, 10 and 15 dB. The test set was created using 5 noise conditions (all from the DEMAND database, but different from training noise conditions) added at SNRs 2.5, 7.5, 12.5 and 17.5 dB. The utterances were downsampled from 48 kHz to 16 kHz for our experiments.

We used the same cGAN architecture with "U"-shaped skip connections as used in [10,12] with 11 convolutional layers each for G_{enc} and G_{dec} with filter-length 31 and stride = 2. The model used approximately 1 second of speech (16 384 samples) as input to the network. Thus, after 11 strided convolutional layers in G_{enc}, the temporal dimension shrunk to $16\,384/2^{11} = 8$.

The number of feature-maps used in the convolutional layers were: 16, 32, 32, 64, 64, 128, 128, 256, 256, 512 and 1024, resulting in an encoder output of size 8×2014. This output was concatenated with a latent vector of the same size which serves as input to the decoder part. G_{dec} followed the reverse procedure that doubled the temporal dimension after every layer resulting in a final output size that was identical to that of the input noisy signal.

Similar to [10], the D-network uses of the same structure as G_{enc}, but with a few differences: (i) it has two input channels (one for $\mathbf{x}$ or $\hat{\mathbf{x}}$; and one for $\mathbf{y}$), (ii) it uses a normalization layer before the non-linearity, (iii) it uses LeakyReLU non-linearity instead of PReLU, and (iv) there is an additional convolutional layer with one filter of width 1 (1×1 convolution) and its output is fed to a fully-connected layer to perform the binary classification. Instead of virtual batch normalization (VBN) as used in [10], we used instance normalization in every layer of D as it gave a better performance. To substantiate this, a comparison between VBN and instance normalization is also provided in the results section.

The model was trained using the Adam optimizer [25] for 80 epochs with a learning rate of 0.0002 using a batch-size of 100. The speech signals were windowed using sliding windows of length 16 384 with 50% overlap. During testing, the enhanced signals were reconstructed by adding the generated signals with the same overlap and dividing the overlapping sections by 2 to compensate for the 50% overlap. We also applied a pre-emphasis filter of impulse response $[-0.95,\ 1]$ to all input samples and the enhanced signals were de-emphasized during testing.

The hyper-parameters that control the additional penalty terms were set as $\lambda_{L1} = 200$ and $\lambda_{GP} = 10$ such that they have the same order of magnitude with respect to the discriminator loss. Similar to the prior works [12,13], we also omitted the latent noise input $\mathbf{z}$ as it did not affect the performance. The whole project was developed in Keras [26] with Tensorflow [27] back-end and is available on github[1].

The speech enhancement performance was evaluated using the following measures: the short-term objective intelligibility (STOI) metric [28], perceptual evaluation of speech quality (PESQ) [29], segmental SNR (segSNR), cepstral distance (CD) and log-likelihood ratio (LLR). Higher values of PESQ, STOI and segSNR, and lower values of CD and LLR indicate better performance.

[1]The proposed cGAN-based framework is available at `https://github.com/deepakbaby/se_relativisticgan`

Table 1. *Comparison of the different GAN-based speech enhancement systems. The normalization technique used in the discriminator is given in brackets, where VBN: virtual batch normalization and IN: instance normalization. Higher values of PESQ, STOI and segSNR, and lower values of CD and LLR indicate better performance. The best results obtained are highlighted in bold font.*

Setting	STOI	PESQ	CD	LLR	segSNR
Unprocessed	0.921	1.97	4.41	0.46	8.77
LSTM-IRM [6]	0.931	2.48	2.76	0.33	15.73
AECNN	0.937	2.59	2.99	0.45	16.93
LSGAN (VBN) [10]	0.925	2.18	3.39	0.44	15.43
LSGAN (IN)	0.937	2.50	3.11	0.44	16.45
WGAN-GP (IN)	0.937	2.54	2.87	0.38	17.56
Proposed SERGAN Models					
RSGAN-GP (IN)	0.940	2.60	2.90	0.42	17.58
RaSGAN-GP (IN)	0.938	2.61	2.90	0.37	17.46
RaLSGAN-GP (IN)	0.938	2.59	2.98	0.43	17.24
RSGAN-GP	**0.942**	2.59	2.58	**0.31**	17.57
RaSGAN-GP	**0.942**	2.59	**2.56**	0.33	**17.68**
RaLSGAN-GP	0.940	**2.62**	2.90	0.33	17.17

5. RESULTS

The speech enhancement performance obtained for the various speech enhancement systems are provided in Table 1. An LSTM-based ideal ratio mask (IRM) estimation speech enhancement system is used as a baseline system [6]. This model had 3 LSTM layers that were trained to minimize the mean-square error between the predicted and the target IRMs for enhancing the gammatone spectrogram of noisy speech. The generator network without the discriminator (denoted as AECNN) is also included as a second baseline system to investigate the impact of adversarial training. This AECNN uses the same encoder-decoder structure in G which is trained to minimize the $L1$ loss between the AECNN output and the underlying clean speech.

The LSGAN model using instance normalization (IN) resulted in a better performance than using virtual batch normalization (VBN). LSGAN training was unstable when no normalization technique was used in the discriminator. The LSGAN model with instance normalization performed still worse than the AECNN model, implying that the adversarial training in fact resulted in a poorer generator.

The relativistic GAN models yielded a better speech enhancement performance, suggesting that a better discriminator leads to a better generator that outperforms the AECNN model. Relativistic GAN models without any normalization yielded the best performance in terms of STOI, CD and LLR, whilst providing comparable PESQ and segSNR scores over the other systems.

Simulation results show that relativistic GAN models with binary cross-entropy loss outperform the least-square and Wasserstein loss functions. While most prior speech enhancement GANs required some normalization in D to help stabilize the training, the proposed SERGAN models with gradient penalty does not require any normalization. In fact, not using any normalization in D resulted in the best performing speech enhancement models in this work. Note that all the experiments in this work included only modifications to the discriminator part only and our results show that better discriminators yielded stable training and overall better performing speech enhancement system.

6. CONCLUSIONS AND FUTURE WORK

This paper introduced SERGANs with different relativistic GAN cost functions for speech enhancement in the time-domain. The simulations showed that SEGANs with gradient penalty improve the speech enhancement capability of the generator, in addition to providing a stable training behavior. This work also introduced several new cost functions such as standard binary cross-entropy loss and its relativistic variants (RSGAN and RaSGAN), relativistic least-square loss (RaLSGAN) and Wasserstein loss (WGAN) for GAN-based speech enhancement. Almost all prior works used LSGANs since SGANs were observed to be unstable. This work reintroduces the standard binary cross-entropy loss with RSGAN and RaSGAN for speech enhancement and showed that these models can outperform their LSGAN counterparts. The proposed SERGAN models were shown to perform better than the AECNN model, implying that the speech enhancement performance in GAN-based models critically depends on the discriminator quality. It is also observed that gradient penalty is crucial for a stable training of SERGAN models, which in turn leads to a faster training as the normalization in D is no longer required.

Since the proposed SERGAN models outperform their AECNN counterpart and a state-of-the-art LSTM-based spectral enhancement system, applying SERGANs as a front-end for automatic speech recognition systems is a promising research direction.

7. REFERENCES

[1] Y. Wang, A. Narayanan, and D. Wang, "On training targets for supervised speech separation," *IEEE/ACM-TASLP*, vol. 22, no. 12, pp. 1849–1858, Dec 2014.

[2] Y. Xu, J. Du, L. R. Dai, and C. H. Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM-TASLP*, vol. 23, no. 1, pp. 7–19, 2015.

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

[4] L. Sun, J. Du, L. R. Dai, and C. H. Lee, "Multiple-target deep learning for LSTM-RNN based speech enhancement," in *HSCMA*, Mar 2017, pp. 136–140.

[5] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, "Deep recurrent networks for separation and recognition of single-channel speech in nonstationary background audio," in *New Era for Robust Speech Recognition, Exploiting Deep Learning.*, 2017, pp. 165–186.

[6] D. Baby and S. Verhulst, "Biophysically-inspired features improve the generalizability of neural network-based speech enhancement systems," in *INTERSPEECH*. ISCA, Sep 2018.

[7] A. Narayanan and D. Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition," in *IEEE-ICASSP*, 2013, pp. 7092–7096.

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

[9] I. J. Goodfellow, J. Pouget-Abadie *et al.*, "Generative adversarial nets," in *NIPS*, vol. 27, Dec 2014, pp. 2672–2680.

[10] S. Pascual, A. Bonafonte, and J. Serrà, "SEGAN: speech enhancement generative adversarial network," in *INTERSPEECH*. ISCA, Aug 2017, pp. 3642–3646.

[11] D. Michelsanti and Z. H. Tan, "Conditional generative adversarial networks for speech enhancement and noise-robust speaker verification," in *INTERSPEECH*. ISCA, Aug 2017, pp. 2008–2012.

[12] A. Pandey and D. Wang, "On adversarial training and loss functions for speech enhancement," in *IEEE-ICASSP*, Apr 2018, pp. 5414–5418.

[13] C. Donahue, B. Li, and R. Prabhavalkar, "Exploring speech enhancement with generative adversarial networks for robust speech recognition," *CoRR*, vol. abs/1711.05747, 2017.

[14] K. Wang, J. Zhang, S. Sun, Y. Wang, F. Xiang, and L. Xie, "Investigating generative adversarial networks based speech dereverberation for robust speech recognition," *CoRR*, vol. abs/1803.10132, 2018.

[15] T. Salimans, I. J. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *NIPS*, vol. 29, Dec 2016, pp. 2226–2234.

[16] A. Jolicoeur-Martineau, "The relativistic discriminator: A key element missing from standard GAN," *CoRR*, vol. abs/1807.00734, 2018.

[17] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *NIPS*, Dec 2017, pp. 5769–5779.

[18] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," *CoRR*, vol. abs/1701.07875, 2017.

[19] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *CoRR*, vol. abs/1411.1784, 2014.

[20] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," in *IEEE-ICCV*, Oct 2017, pp. 2813–2821.

[21] P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *IEEE-CVPR*, Jul 2017, pp. 5967–5976.

[22] C. Valentini-Botinhao, X. Wang, S. Takaki, and J. Yamagishi, "Investigating rnn-based speech enhancement methods for noise-robust text-to-speech," in *ISCA-SSW*, Sep 2016, pp. 146–152.

[23] C. Veaux, J. Yamagishi, and S. King, "The voice bank corpus: Design, collection and data analysis of a large regional accent speech database," in *IEEE-OCOCOSDA/CASLRE*, Nov 2013, pp. 1–4.

[24] J. Thiemann, N. Ito, and E. Vincent, "The diverse environments multi-channel acoustic noise database (DEMAND): A database of multichannel environmental noise recordings," *Proceedings of Meetings on Acoustics*, vol. 19, no. 1, p. 035081, 2013.

[25] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.

[26] F. Chollet *et al.*, "Keras," https://keras.io, 2015.

[27] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[28] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time-frequency weighted noisy speech," *IEEE-TASLP*, vol. 19, no. 7, pp. 2125–2136, Sep 2011.

[29] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *IEEE-ICASSP*, May 2001, pp. 749–752.