VOICE IMPERSONATION USING GENERATIVE ADVERSARIAL NETWORKS

Yang Gao[*], *Rita Singh*[†], *Bhiksha Raj*

Electrical and Computer Engineering Department, Carnegie Mellon University
yanggao,rsingh,bhiksha@cs.cmu.edu

ABSTRACT

Voice impersonation is not the same as voice transformation, although the latter is an essential element of it. In voice impersonation, the resultant voice must convincingly convey the impression of having been naturally produced by the target speaker, mimicking not only the pitch and other perceivable signal qualities, but also the *style* of the target speaker. In this paper, we propose a novel neural-network based speech quality- and style-mimicry framework for the synthesis of impersonated voices. The framework is built upon a fast and accurate generative adversarial network model. Given spectrographic representations of source and target speakers' voices, the model learns to mimic the target speaker's voice quality and style, regardless of the linguistic content of either's voice, generating a synthetic spectrogram from which the time-domain signal is reconstructed using the Griffin-Lim method. In effect, this model reframes the well-known problem of style-transfer for images as the problem of style-transfer for speech signals, while intrinsically addressing the problem of durational variability of speech sounds. Experiments demonstrate that the model can generate extremely convincing samples of impersonated speech. It is even able to impersonate voices across different genders effectively. Results are qualitatively evaluated using standard procedures for evaluating synthesized voices.

Index Terms— Voice impersonation, generative adversarial network, style transformation, style transfer

1. INTRODUCTION

Voice impersonation, where one person attempts to mimic the voice of another to sound like the other person, is a complex phenomenon. It is often performed by the impersonator by intuitively trying to match the most prominent aspects of the voice of the person mimicked. The most common sets of characteristics in voice that are varied by the impersonator in this process are elements of *voice quality* and elements of *style*. Voice quality, however, is a hard-to-describe entity. While some aspects of it, such as nasality, roughness, breathiness etc. have been identified in the literature, most of these are not quantifiable in a manner that permits meaningful comparison across speakers. Elements of *style*, which include temporal pitch and energy patterns that are characteristic of the target speaker's cadence, speaking rate, prosody etc. are similarly ambiguous in terms of their exact definition, quantifiability and comparability. In addition to quality and style, idiosyncrasies of language such as the variants of grammar, choice of vocabulary, usage of words such as "um", "uh" etc. are also mimicked by the impersonator to create an impression of the target speaker's voice; these are however currently not the focus of this paper.

In this paper, we attempt to address the problem of automatic generation of impersonations by transferring the stylistic content of speech from the target speaker to that of the impersonator. The objective is to convert only specific aspects of the voice, without modifying other content. Ideally, we would be able to identify, isolate and explicitly measure the aspects of voice that characterize stylistic features, and then use them to modify the target voice. However, this is generally difficult because such features are subjectively or even objectively identifiable, but not quantifiable. To misquote the Hon. Potter Stewart, we may not know how to describe them, but we know when they occur.

Our objective is to develop a mechanism to transfer such identifiable (albeit unquantifiable) aspects of style. In the absence of mechanisms to isolate, quantify and measure these, generative approaches such as source-filter models within which explicit components of the source of filter can be modified [1], spectral-transformation models such as PSOLA [2], etc. become inappropriate. Instead, we must use an alternate approach, where the ability to transform the voice must be learned solely from the identifiability of the target characteristics.

Prior research that is of greatest relevance in this context relates to *voice transformation*, which deals with the specific problem of converting a source voice into a target one. Voice transformation has had a long history, and at the surface addresses some of the issues we mention. Conventionally, voice transformation modifies the instantaneous characteristics of a source signal, such as pitch [3] and spectral envelope. The strategies used range from simple codebook-based conversion [4] and minimum-mean-squared error linear estimators [5] to sophisticated neural network models [6]. While these methods are all frequently quite effective at transforming instantaneous characteristics of the signal, and can even map some prosodic cues, they are generally insufficient to capture unmeasurable, unquantifiable *style* in the more general sense of the word. When trained, they are heavily reliant on the availability of *parallel* recordings of the source and target speaker saying the same utterances, providing exact examples of what is considered ideal conversion. In most cases, in order to learn the voice conversion effectively, these recordings must also be perfectly time aligned, a requirement that is generally satisfied by time-warping the recordings to align them to one another. Realization of the *hard* targets required to learn the conversion is not only unrealistic, the alignment required may also be fundamentally inappropriate when the objective is not to learn to perform wholesale conversion of voice, but only to transform style.

In this context, recent advances in the science of learning generative models provide us new directions. Rather than attempting to learn a mapping between parallel signals, the new models attempt to *discriminate* instead between data that do have the desired (identifiable but possibly unquantifiable) stylistic feature(s), and those that do not. Generators that attempt to produce data with any specific characteristic(s) must now learn to do so such that they "fool" the discriminator. Since the features are unquantifiable, the discrimi-

[*] Thanks to Raymond (Yangyang) Xia for his suggestions and advice

[†] This work was funded by Schmidt Sciences, Palo Alto, CA.

nator itself must, in fact also be learned. Both the generators and the discriminators are modeled by deep neural networks, which are known to be able to model any transformation with appropriate design and sufficient training data. Since the primary driver of the learning process is discrimination, parallel data such as those needed for conventional voice-conversion methods are not required.

These *Generative Adversarial Networks*, or GANs have been very successfully applied to a variety of problems in image generation [7], learning feature representations [8] and *style transfer* [9, 10, 11, 12, 13], wherein the algorithms involved result in fast and vivid generation of images of different artistic styles ranging from simple photographs to painting styles of selected artists. In our work, we harness the power of these models for the problem of style transfer in speech.

At the outset, we note that speech signals have several problems that are not inherent to images. Unlike images, speech sounds are not of fixed size (i.e., not fixed in duration), and lose much of their stylistic characteristics when they are scaled down to be so. Generation of time-series data such as speech is also a more challenging problem compared to images. Naive implementations of the process may result in generation of data that have lost linguistic, stylistic or even intelligible content. In this work, we propose multiple GAN models for the problem of voice transformation. Our models, and their corresponding learning algorithms, are designed to consider the specific challenges inherent in speech. Specifically, we show how, by appropriate choice of model structure and learning algorithm, and by introducing the appropriate discriminators in the GAN framework, *specific* characteristics of the voice might be retained without modifying others or losing linguistic content, in order to emulate different aspects of impersonation or voice mimicry.

In Section 2 we briefly outline the concept of GANs. In Section 3 we describe our designs of GANs for voice modifications. In Section 4 we present experimental evaluations of the proposed models and conclude with discussions in Section 5.

2. GENERATIVE ADVERSARIAL NETWORKS

In spite of their rather short history, GANs [7] are already quite well known. We briefly summarize their key features here, in order to set the background for the rest of the paper.

2.1. The basic GAN model

The Generative Adversarial Network is a generative model which, at its foundation, is a generative model for a data variable. The model is intended to generate samples that closely match draws from the actual distribution of the data. These models differ from conventional generative models in a fundamental way in the manner in which they are learned. Conventional generative models are trained through likelihood maximization criteria, such that some (empirical estimate of the) divergence measure between the synthetic distribution encoded by the generative models, and the true distribution of the data, is minimized. In contrast, GANs are trained discriminatively, such that samples generated from the model cannot be distinguished from actual draws from the true distribution of the data.

Consider any random variable $\mathbf{x}$ with a probability distribution $P_\mathbf{x}$ that is unknown, but from which samples may be drawn. For instance, $\mathbf{x}$ may represent images of a particular class, samples of which may be readily available, but their actual distribution may be unknown. The GAN attempts to generate samples of $\mathbf{x}$ that are indistinguishable from actual samples drawn from the true distribution. The original GAN model [7] comprises a generator $G(\mathbf{z})$ and discriminator $D(\mathbf{x})$. The generator G takes as input a random variable $\mathbf{z}$ drawn from some standard probability distribution function $P_\mathbf{z}$, *e.g.* a standard Normal distribution, and produces an output vector $\mathbf{x}_\mathbf{z}$.

The discriminator $D()$ attempts to discriminate between samples $\mathbf{x} \sim P_\mathbf{x}$ that are drawn from $P_\mathbf{x}$, the true (but unknown) distribution we aim to model, and samples produced by the Generator G. Let T represent the event that a vector $\mathbf{x}$ was drawn from $P_\mathbf{x}$. The discriminator attempts to compute the *a posteriori* probability of T, *i.e.* $D(\mathbf{x}) = P(T|\mathbf{x})$.

To train the GAN, we attempt to learn G such that $D(\mathbf{x}_\mathbf{z})$, the score output by the discriminator in response to productions by G is maximized (i.e. G "fools" the discriminator). At the same time we attempt to learn D such that $D(\mathbf{x}_\mathbf{z})$ is minimized, while also maximizing $D(\mathbf{x})$ for any $\mathbf{x} \sim P_\mathbf{x}$. All of these objectives can be concurrently achieved through the following optimization:

$$\min_G \max_D E_{\mathbf{x}\sim P_\mathbf{x}}[\log D(\mathbf{x})] + E_{z\sim P_\mathbf{z}}[\log(1 - D(x_\mathbf{z}))]$$

The GAN training framework is illustrated in Figure 1.

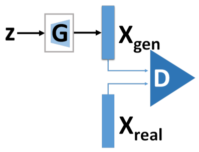


Fig. 1. The original GAN model

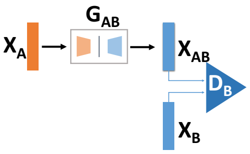


Fig. 2. Style transfer by GAN

2.2. GANs for style transfer

The basic GAN has been extended in a number of ways in the literature [9, 10, 11, 12, 13], particularly in the context of style transfer among images, *e.g.* as in Figure 2. The common underlying denominator in all of these models is that an input data instance (usually an image) x_A drawn from a distribution P_A is *transformed* to an instance x_{AB} by a generator (more aptly called a "transformer"), G_{AB}. The aim of the transformer is to convert x_A into the style of the variable x_B which natively occurs with the distribution P_B.

The discriminator D_B attempts to distinguish between genuine draws of x_B from P_B and instances x_{AB} obtained by transforming draws of x_A from P_A. The actual optimization is achieved as follows. We define

$$\begin{aligned} L_G &= E_{x_A\sim P_A}[\log(1 - D_B(x_{AB}))] \\ L_D &= -E_{x_B\sim P_B}[\log D_B(x_B)] - E_{x_A\sim P_A}[\log(1 - D_B(x_{AB})] \end{aligned} \tag{1}$$

To train the GAN, its two components are alternately updated by minimizing the two losses in Equation 1. The generator G is updated by minimizing the "generator loss" L_G, while the discriminator is updated to minimize the "discriminator loss" L_D.

Our work is however more directly based on the "DiscoGAN" model [9], shown in Figure 3. The DiscoGAN is a symmetric model which attempts to transform two categories of data, A and B, into each other. The DiscoGAN includes two generators (more aptly called "transformers") G_{AB} and G_{BA}. G_{AB} attempts to transform any draw x_A from the distribution P_A of A into $x_{AB} = G_{AB}(x_A)$, such that x_{AB} is indistinguishable from draws x_B from the distribution P_B of B. G_{BA} does the reverse.

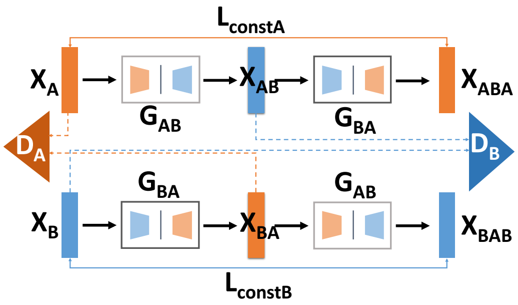


Fig. 3. The DiscoGAN model

It also includes two discriminators, D_A and D_B. D_A attempts to discriminate between actual draws from P_A and draws from P_B that have been transformed by G_{BA}, and D_B performs the analogous operations for draws from P_B. The generators and discriminators must all be jointly trained.

The training process for the DiscoGAN is similar to that for the model in Figure 2, with one significant modification: in addition to the losses that emphasize the competition between the generators and the discriminators, we now include the requirement that G_{AB} and G_{BA} must be inverses of each other to the extent possible, *i.e.* for any x_A from A, $x_{ABA} = G_{BA}(G_{AB}(x_A))$ must be close to the original x_A, and similarly for any x_B from B, $x_{BAB} = G_{AB}(G_{BA}(x_B))$ must be close to the original x_B. This requirement is encoded through two reconstruction losses L_{CONST_A} and L_{CONST_B} where

$$L_{CONST_A} = d(G_{BA}(G_{AB}(x_A)), x_A) \tag{2}$$

and L_{CONST_B} is symmetrically defined. The generator loss for G_{AB} is defined as:

$$L_{GAN_{AB}} = L_{CONST_A} + L_{G_B} \tag{3}$$

where L_{G_B} is defined as in Equation 1. We define the generator loss for G_{AB} in a symmetric manner. The overall generator loss is $L_G = L_{GAN_{AB}} + L_{GAN_{BA}}$. The discriminator loss L_D is defined as $L_D = L_{D_A} + L_{D_B}$, L_{D_A} and L_{D_B} are defined as in Equation 1. Finally, in the implementation of DiscoGAN [9], a *feature loss* is also added to compare the feature similarity between the generated data and the real data. As before, the generators and discriminators are trained by alternate minimization of the generator and discriminator losses.

3. GANS FOR VOICE MIMICRY

The DiscoGAN was originally designed to transform style in images. In order to apply the model to speech, we first convert it to an invertible, picture-like representation, namely a spectrogram. We operate primarily on the *magnitude* spectrogram, retaining the phase of input signals to be transformed, to recreate the transformed signals from the transformed magnitude spectrogram.

But first, we must make several key modifications to the DiscoGAN model. The original DiscoGAN was designed to operate on images of fixed size. For it to work with inherently variable-sized speech signals, this constraint must be relaxed in its new design. Secondly, it is important to ensure that the *linguistic* information in the speech signal is not lost, even through the signal itself is modified. Sufficient constraints must be added to the model for this. Finally, since our objective is to modify specific aspects of the speech, *e.g.* style, we must add extra components to our model to achieve this. We call our model, which incorporates all these modifications, the *VoiceGAN*.

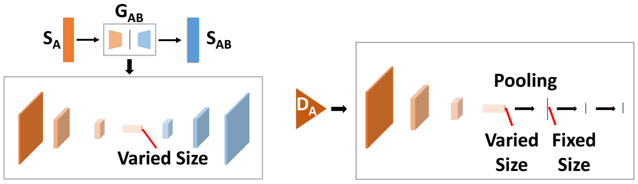
3.0.1. Retaining Linguistic Information

Linguistic information is encoded largely in the details of the spectral envelope. To ensure that this is retained, we modify our reconstruction loss as:

$$L_{CONST_A} = \alpha d(x_{ABA}, x_A) + \beta d(x_{AB}, x_A) \tag{4}$$

Here, the term $d(x_{AB}, x_A)$ attempts to retain the *structure* of x_A even after it has been converted to x_{AB}. Careful choice of α and β ensures both, accurate reconversion and retention of linguistic information, after conversion to x_{AB}.

3.0.2. Variable-length Input Generator and Discriminator



(a) Generator G_A

(b) Discriminator D_A

Fig. 4. Visualization of Generator G_A and Discriminator D_A. Architectures of other counterparts are similar in structure. The number of convolutional layers is larger in the actual implementation.

To account for the fact that unlike images, speech signals are of variable length that cannot be scaled up or down, we must make modifications to the generators and discriminators. The modified structures are shown in Figure 4. Figure 4 (a) shows the structure of the original generator in DiscoGAN. Based on its fully convolutional structure, it can handle variable length inputs. Figure 4 (b), we shows the architectural details for our proposed discriminator in VoiceGAN. In this, an adaptive pooling layer is added after the CNN layers, and before the fully connected layer. It includes channel-wise pooling in which each channel's feature map is pooled into a single element. This converts any variable-sized feature map into a vector of a fixed number of dimensions, with as many components as the number of channels.

3.0.3. Style Embedding Model (D_S)

In addition to the discriminator that distinguishes between the generated data and real data, we add a second type of discriminator to our model to further extract the target *style* information from input data and to make sure that the generated data still has this style information embedded in it. To achieve this, we include a discriminator D_S that is similar in architecture to that in Figure 5.

The discriminator D_S determines if the original and transformed signals match the desired style. To do so we introduce the following style loss:

$$\begin{aligned} L_{D_{STYLE-A}} =& d(D_S(x_A), label_A) + d(D_S(x_{AB}), label_B) \\ & + d(D_S(x_{ABA}), label_A) \end{aligned} \tag{5}$$

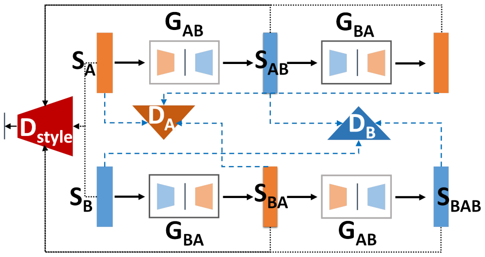


Fig. 5. The proposed VoiceGAN Model

$$L_{D_{STYLE}} = L_{D_{STYLE-A}} + L_{D_{STYLE-B}} \tag{6}$$

Note that the style loss could include multiple discriminators for multiple aspects of style.

3.0.4. *Total Loss*

Our final training objectives to be minimized for the generator and discriminator are represented by L_G and L_D respectively as follows:

$$\begin{aligned} L_G &= L_{GAN_{AB}} + L_{GAN_{BA}} \\ &= L_{G_B} + L_{CONST_A} + L_{G_A} + L_{CONST_B} \end{aligned} \tag{7}$$

$$L_D = L_{D_A} + L_{D_B} + L_{D_{STYLE}} \tag{8}$$

4. EXPERIMENTS AND RESULTS

We use the TIDIGITS [14] dataset. This dataset comprises a total of 326 speakers: 111 men, 114 women, 50 boys and 51 girls. Each speaker reads 77 digit sentences. The sampling rate of the audio is 16000 Hz. We chose to use this database due to its relatively simple linguistic content. For the purpose of demonstration, we choose an unquantifiable, but identifiable characteristic: gender. Our goal then is to show that these data can be used to *learn* to convert the gender of a speaker's voice. In the discussion below, therefore, "style" refers to gender. We note that any other characteristic may have been similarly chosen.

4.1. Model implementation

The model architecture is that of the VoiceGAN described above. The generator network in the model comprises a 6-layer CNN encoder and a 6-layer transposed CNN decoder. The discriminator network comprises a 7-layer CNN with adaptive pooling. We employ batch normalization [15] and leaky ReLU activations [16] in both the networks. The number of filters in each layer is an increasing power of 2 (32, 64, 128). When training the networks, a smoothness constraint, comprising the cumulative first order difference between adjacent columns in the spectrogram, is added to the loss to enhance the temporal continuity of the generated spectrogram. Results are available at [17].

4.2. Quality evaluation of generated results

4.2.1. *Style Classification Test*

We use an independently-trained CNN-based classifier to predict the style of our generated data. The classifier was trained on 800 utterances from speakers of both genders. The results show that 100% of the generated data are classified as the target speaker's style, which indicates that our VoiceGAN network achieves good style transfer performance.

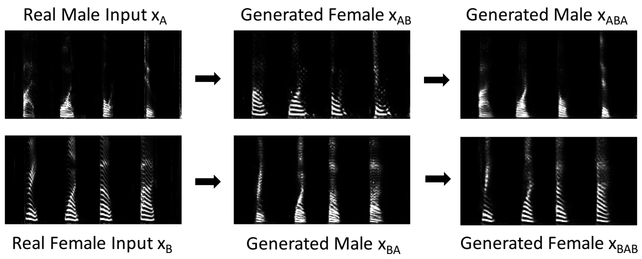


Fig. 6. Visualization of spectrograms generated from a speaker saying "3 1 oh 5" (first row) and "5 1 4 2" (second row). For each spectrogram, frequencies on the y-axis range from 0-4 kHz.

Table 1. NIST STNR TEST

Data (use GL-method)	A (dB)	B (dB)
Original signal	55.60±4.97	52.91±3.58
X_A and X_B	54.97±6.28	52.15±3.70
X_{AB} and X_{BA}	49.64±1.80	49.92±4.36
X_{ABA} and X_{BAB}	53.58±2.69	50.05±2.12

4.2.2. *Speech Signal to Noise Ratio (SNR) Test*

To evaluate the quality of our generated speech signal [18], we also conduct a signal-to-noise (SNR) ratio test using the standard NIST STNR method and the WADA SNR method [19]. The results are shown in Table 2. For each data class, we randomly select 40 samples from our test dataset (20 for each speaker) and compute the mean and variance of the generated results. The WADA test results are all around 100 dB since our generated noise is not well-modeled by Gaussian noise. The STNR test results show that our generated data is of good quality. For evaluation, the time-domain signal is reconstructed from the generated spectrogram using the Griffin-Lim method, which is based on an iterative procedure that minimizes the mean square error between the modified magnitude spectrogram and the actual signals spectrogram. Details of this method are explained in [20]. We find that the Griffin-Lim method does not reduce the voice quality to any significant degree.

5. CONCLUSIONS

The VoiceGAN model is observably able to transfer style from one speaker to another. As proposed however, this model remains vanilla and many extensions are possible. The method is easily extended to other stylistic features that may be identified. In principle, while longer-term prosodic-level style features may also be transferred, simple binary discriminators may no longer be useful for such characteristics. More continuous-valued discrimination may be required. We have not verified if multiple style aspects may be *concurrently* modified. These remain areas of ongoing research. In preliminary experiments we have verified that even *linguistic* content may be modified if we so choose; however doing so in a measurable and controlled manner is a challenge that remains to be addressed. Future versions of the VoiceGAN model will continue to incorporate the most relevant innovations in the area of adversarial modeling.

6. REFERENCES

[1] Daniel Erro, Eva Navas, and Inma Hernaez, “Parametric voice conversion based on bilinear frequency warping plus amplitude scaling,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 3, pp. 556–566, 2013.

[2] Hélene Valbret, Eric Moulines, and Jean-Pierre Tubach, “Voice transformation using PSOLA technique,” *Speech communication*, vol. 11, no. 2-3, pp. 175–187, 1992.

[3] K. Sreenivasa Rao, “Voice conversion by mapping the speaker-specific features using pitch synchronous approach,” *Computer Speech and Language*, vol. 24, pp. 474–494, 2010.

[4] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, “Voice conversion through vector quantization,” *Journal of the Acoustic Society of Japan (E)*, vol. 11, no. 2, pp. 71–76, 1990.

[5] Tomoki Toda, Alan W Black, and Keiichi Tokuda, “Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2222–2235, 2007.

[6] Ling-Hui Chen, Zhen-Hua Ling, Li-Juan Liu, and Li-Rong Dai, “Voice conversion using deep neural networks with layer-wise generative training,” *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 22, no. 12, pp. 1859–1872, 2014.

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, “Generative adversarial nets,” in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[8] Alec Radford, Luke Metz, and Soumith Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” *arXiv preprint arXiv:1511.06434*, 2015.

[9] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jungkwon Lee, and Jiwon Kim, “Learning to discover cross-domain relations with generative adversarial networks,” *arXiv preprint arXiv:1703.05192*, 2017.

[10] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” *arXiv preprint arXiv:1703.10593*, 2017.

[11] Ahmed Elgammal, Bingchen Liu, Mohamed Elhoseiny, and Marian Mazzone, “Can: Creative adversarial networks, generating” art” by learning about styles and deviating from style norms,” *arXiv preprint arXiv:1706.07068*, 2017.

[12] Ming-Yu Liu, Thomas Breuel, and Jan Kautz, “Unsupervised image-to-image translation networks,” *arXiv preprint arXiv:1703.00848*, 2017.

[13] Leon A Gatys, Alexander S Ecker, and Matthias Bethge, “Image style transfer using convolutional neural networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2414–2423.

[14] Linguistic Data Consortium, “TIDIGITS,” https://catalog.ldc.upenn.edu/LDC93S10, 1993.

[15] Sergey Ioffe and Christian Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in *International Conference on Machine Learning*, 2015, pp. 448–456.

[16] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li, “Empirical evaluation of rectified activations in convolutional network,” *arXiv preprint arXiv:1505.00853*, 2015.

[17] Yolanda Gao, “Audio examples,” https://github.com/Yolanda-Gao/Spectrogram-GAN.

[18] Wei Ming Liu, Keith A Jellyman, John SD Mason, and Nicholas WD Evans, “Assessment of objective quality measures for speech intelligibility estimation,” in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*. IEEE, 2006, vol. 1, pp. I–I.

[19] Chanwoo Kim and Richard M Stern, “Robust signal-to-noise ratio estimation based on waveform amplitude distribution analysis,” in *Ninth Annual Conference of the International Speech Communication Association*, 2008.

[20] Daniel Griffin and Jae 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.