Generative adversarial network-based approach to signal reconstruction from magnitude spectrogram

Keisuke Oyamada* , Hirokazu Kameoka† , Takuhiro Kaneko † ,
Kou Tanaka † , Nobukatsu Hojo † , Hiroyasu Ando *
* *University of Tsukuba, Japan*,
Email: s1720584@s.tsukuba.ac.jp
† *NTT Communication Science Laboratories, NTT Corporation, Japan*,

***Abstract*—In this paper, we address the problem of reconstructing a time-domain signal (or a phase spectrogram) solely from a magnitude spectrogram. Since magnitude spectrograms do not contain phase information, we must restore or infer phase information to reconstruct a time-domain signal. One widely used approach for dealing with the signal reconstruction problem was proposed by Griffin and Lim. This method usually requires many iterations for the signal reconstruction process and depending on the inputs, it does not always produce high-quality audio signals. To overcome these shortcomings, we apply a learning-based approach to the signal reconstruction problem by modeling the signal reconstruction process using a deep neural network and training it using the idea of a generative adversarial network. Experimental evaluations revealed that our method was able to reconstruct signals faster with higher quality than the Griffin-Lim method.**

***Index Terms*—phase reconstruction, deep neural networks, generative adversarial networks**

I. Introduction

This paper addresses the problem of reconstructing a time-domain signal solely from a magnitude spectrogram.

The magnitude spectrograms of real-world audio signals tend to be highly structured in terms of both spectral and temporal regularities. For example, pitch contours and formant trajectories are clearly visible from a magnitude spectrogram representation of speech compared with a time-domain signal. Therefore, there are many cases where processing magnitude spectrograms can deal with problems more easily than directly processing time-domain signals. In fact, many methods for monaural audio source separation are applied to magnitude spectrograms [1]–[3]. Furthermore, a magnitude spectrogram representation was recently found to be reasonable and effective for use with speech synthesis systems [4], [5].

Since a magnitude spectrogram does not contain phase information, we must restore or infer phase information to reconstruct a time-domain signal. This problem is called the signal (or phase) reconstruction problem. One widely used method for solving the signal reconstruction problem was proposed by Griffin and Lim [6] (hereafter referred to as the Griffin-Lim method). One of the drawbacks of the Griffin-Lim method is that it usually requires many iterations to obtain high-quality audio signals. This makes it particularly difficult to apply it to real-time systems. Furthermore, there are some cases where high-quality audio signals can never be obtained even though the algorithm is run for many iterations. To overcome these shortcomings of the Griffin-Lim method, we apply a learning-based approach to the signal reconstruction problem. Specifically, we propose modeling the reconstruction process of a time-domain signal from a magnitude spectrogram using a deep neural network (DNN) and propose introducing the idea of the generative adversarial network (GAN) [7] for training the signal generator network.

The remainder of the paper is organized as follows. We provide an overview of the phase reconstruction problem in Section II, introduce the Griffin-Lim method in Section III, and present our GAN-based approach in Section IV. Experimental evaluations, and supplements for training our model are provided in Section V. Finally, we offer our conclusions in Section VI.

This work was conducted as a part of research undertaken by the Center for Artificial Intelligence Science, University of Tsukuba and supported by JSPS KAKENHI Grant Number 17H01763.

II. Signal Reconstruction Problem

In this section, we provide an overview of the signal reconstruction problem.

We use $\mathbf{x} = [x(0), \ldots, x(T-1)]^{\mathsf{T}} \in \mathbb{R}^T$ to denote a time domain signal and $c_{f,n} \in \mathbb{C}$ to denote the time frequency component of $\mathbf{x}$ where f and n indicate frequency and time indices, respectively. By defining $\mathbf{w}_{f,n} = [w_{f,n}(0), \ldots, w_{f,n}(T-1)]^{\mathsf{T}} \in \mathbb{C}^T$ as a complex sinusoid of frequency ω_f modulated by a window function centered at time t_n, $c_{f,n}$ is defined by the inner product between $\mathbf{x}$ and $\mathbf{w}_{f,n}$, namely $c_{f,n} = \mathbf{w}_{f,n}^{\mathsf{H}}\mathbf{x}$. With a short-time Fourier transform (STFT), t_n corresponds to the center time of frame n and $\mathbf{w}_{f,n}$ is the modulated complex sinusoid padded with zeros over the range outside the frame. By using $\mathbf{c} \in \mathbb{C}^{FN}$ to denote a vector obtained by stacking all the time-frequency components $c_{f,n}$, the relationship between $\mathbf{c}$ and $\mathbf{x}$ can be written as

$$\mathbf{c} = \mathbf{W}\mathbf{x}, \tag{1}$$

where $\mathbf{W}$ is a $FN \times T$ matrix where each row is $\mathbf{w}_{f,n}^{\mathsf{H}}$. Hereafter, we call $\mathbf{c}$ a complex spectrogram. Since the total number FN of time frequency points is usually set at more than the number T of sample points of the time domain signal, $\mathbf{c}$ is a redundant representation of $\mathbf{x}$. Namely, $\mathbf{c}$ belongs to

ISBN 978-90-827970-1-5

a T-dimensional linear subspace $\mathcal{C}$ spanned by each column vector of $\mathbf{W}$. With an STFT, all the elements of a complex spectrogram must satisfy certain conditions to ensure that the waveforms within the overlapping segment of consecutive frames are consistent. By using $\mathbf{a}$ to denote the magnitude spectrogram of $\mathbf{c}$ where each element of $\mathbf{a}$ is given by the absolute value of the element of $\mathbf{c}$, the signal reconstruction problem can be cast as an optimization problem of estimating $\mathbf{x}$ solely from $\mathbf{a}$ using the redundancy constraint as a clue.

III. Griffin-Lim Method

One widely used way of solving the phase reconstruction problem involves the Griffin-Lim method [6]. In this section, we derive the iterative algorithm of the Griffin-Lim method following the derivation given in [8].

Whether or not a given $\mathbf{c}$ satisfies the redundancy constraint so that $\mathbf{c}$ is a complex spectrogram associated with a time domain signal can be evaluated by examining whether or not the orthogonal projection $\mathbf{W}\mathbf{W}^{+}\mathbf{c}$ of $\mathbf{c}$ to the subspace $\mathcal{C}$ matches $\mathbf{c}$. Here, $\mathbf{W}^{+}$ is a pseudo inverse matrix of $\mathbf{W}$ satisfying

$$\begin{aligned}\mathbf{W}^{+}\mathbf{c} &= \operatorname*{argmin}_{\mathbf{x}} \|\mathbf{c}-\mathbf{W}\mathbf{x}\|_2^2 \\ &= (\mathbf{W}^{\mathsf{H}}\mathbf{W})^{-1}\mathbf{W}^{\mathsf{H}}\mathbf{c}. \end{aligned} \tag{2}$$

With an STFT, (2) corresponds to an inverse STFT. Thus, $\mathbf{W}\mathbf{W}^{+}\mathbf{c}$ is the STFT of the inverse STFT of $\mathbf{c}$. Now, by using $\boldsymbol{\phi}$ to denote a vector where each element is the phase $\phi_{f,n} \equiv e^{\jmath\theta_{f,n}}$, the phase reconstruction problem for a given $\mathbf{a}$ is formulated as an optimization problem of estimating $\boldsymbol{\phi}$ that minimizes

$$\mathcal{J}(\boldsymbol{\phi}) = \|\mathbf{a}\odot\boldsymbol{\phi} - \mathbf{W}\mathbf{W}^{+}(\mathbf{a}\odot\boldsymbol{\phi})\|_2^2, \tag{3}$$

where $\odot$ denotes an element-wise product. From (2), $\mathbf{W}\mathbf{W}^{+}(\mathbf{a}\odot\boldsymbol{\phi})$ is the point closest to $\mathbf{a}\odot\boldsymbol{\phi}$ in the subspace $\mathcal{C}$. Thus, we can rewrite (3) as

$$\mathcal{J}(\boldsymbol{\phi}) = \min_{\tilde{\mathbf{c}}\in\mathcal{C}} \|\mathbf{a}\odot\boldsymbol{\phi} - \tilde{\mathbf{c}}\|_2^2. \tag{4}$$

According to the principle of the majorization-minimization algorithm [9], it can be shown that $\mathcal{J}^{+}(\boldsymbol{\phi},\tilde{\mathbf{c}}) \equiv \|\mathbf{a}\odot\boldsymbol{\phi}-\tilde{\mathbf{c}}\|_2^2$ is a majorizer of $\mathcal{J}(\boldsymbol{\phi})$ where $\tilde{\mathbf{c}}\in\mathcal{C}$ is an auxiliary variable and a stationary point of $\mathcal{J}(\boldsymbol{\phi})$ can be found by iteratively performing the following updates:

$$\tilde{\mathbf{c}} \leftarrow \operatorname*{argmin}_{\tilde{\mathbf{c}}\in\mathcal{C}} \|\mathbf{a}\odot\boldsymbol{\phi}-\tilde{\mathbf{c}}\|_2^2 = \mathbf{W}\mathbf{W}^{+}(\mathbf{a}\odot\boldsymbol{\phi}), \tag{5}$$

$$\boldsymbol{\phi} \leftarrow \operatorname*{argmin}_{\boldsymbol{\phi}} \|\mathbf{a}\odot\boldsymbol{\phi}-\tilde{\mathbf{c}}\|_2^2 = \angle\tilde{\mathbf{c}}. \tag{6}$$

Here $\angle\cdot$ denotes an operation that divides each element of a vector by its absolute value. With an STFT, Eq. (5) can be interpreted as the inverse STFT of $\mathbf{a}\odot\boldsymbol{\phi}$ followed by the STFT whereas Eq. (6) is a procedure for replacing the phase $\boldsymbol{\phi}$ with the phase of $\tilde{\mathbf{c}}$ updated via (5). This algorithm is procedurally equivalent to the Griffin-Lim method [6].

The Griffin-Lim method usually requires many iterations to obtain a high-quality audio signal. This makes it particularly difficult to apply to real-time systems. Furthermore, there are some cases where high-quality audio signals can never be obtained even though the algorithm is run for many iterations, for example when $\mathbf{a}$ is an artificially created magnitude spectrogram. In the next section, we propose a learning-based approach to the phase reconstruction problem to overcome these shortcomings of the Griffin-Lim method.

IV. GAN-based signal reconstruction

A. Modeling phase Reconstruction Process

By using $\boldsymbol{\phi}^{(0)}$ to denote the initial value of $\boldsymbol{\phi}$, and defining $h(\mathbf{a},\boldsymbol{\phi}) \equiv \mathbf{W}\mathbf{W}^{+}\mathbf{a}\odot\boldsymbol{\phi}$ and $g(\mathbf{c}) \equiv \angle\mathbf{c}$, the iterative algorithm of the Griffin-Lim method can be expressed as a multilayer composite function

$$\hat{\mathbf{c}} = h(\mathbf{a}, g(\cdots g(h(\mathbf{a}, g(h(\mathbf{a},\boldsymbol{\phi}^{(0)})))) \cdots)). \tag{7}$$

Here, h is a linear projection whereas g is a nonlinear operation applied to the output of h. Hence, (7) can be viewed as a deep neural network (DNN) where the weight parameters and the activation functions are fixed. From this point of view, finding an algorithm that converges more quickly to a better solution than the Griffin-Lim algorithm can be regarded as learning the weight parameters (and the activation functions) of the DNN. This idea is inspired by the deep unfolding framework [10], which uses a learning strategy to obtain an improved version of a deterministic iterative inference algorithm by unfolding the iterations and treating them as layers in a DNN. Fortunately, an unlimited number of pair data of $\mathbf{c}$ and $\{\mathbf{a},\boldsymbol{\phi}\}$ can be collected very easily by computing the complex, magnitude and phase spectrograms of time domain signals. This is very advantageous for efficiently training our DNN.

In the following, we consider a DNN that uses $\mathbf{a}$ and $\boldsymbol{\phi}$ as inputs and generates $\mathbf{c}$ (or $\mathbf{x}$) as an output. We call this DNN a generator G and express the relationship between the input and output as $\hat{\mathbf{c}} = G(\mathbf{a},\boldsymbol{\phi})$.

B. Learning Criterion

For the generator training, one natural choice for the learning criterion would be a similarity metric (e.g., the ℓ_1 norm) between the generator output and a target complex spectrogram (or signal). Manually defining a similarity metric amounts to assuming a specific form of the probability distribution of the target data (e.g., a Laplacian distribution for the ℓ_1 norm). However, the data distribution is unknown. If we use a similarity metric defined in the data space as the learning criterion, the generator will be trained in such a way that the outputs that averagely fit the target data are considered optimal. As a result, the generator will learn to generate only oversmoothed signals. This is undesirable as the oversmoothing of reconstructed signals causes audio quality degradation. To avoid this, we propose using a similarity metric implicitly learned using a generative adversarial network (GAN) [7]. In addition to the generator network, we introduce a discriminator network D that learns to correctly discriminate the complex spectrograms $\hat{\mathbf{c}}$ generated by the generator and the complex spectrograms of real audio signals. Given a target

ISBN 978-90-827970-1-5 © EURASIP 2018

complex spectrogram $\mathbf{c}$, the discriminator D is expected to find a feature space where $\hat{\mathbf{c}}$ and $\mathbf{c}$ are as separate as possible. Thus, we expect that minimizing the distance between $\hat{\mathbf{c}}$ and $\mathbf{c}$ measured in a hidden layer of the discriminator would make $\hat{\mathbf{c}}$ indistinguishable from $\mathbf{c}$ in the data space. By using $D(\cdot, \mathbf{a}) \in \mathbb{R}$ to denote the discriminator network D, we first consider the following criteria for the discriminator

$$V(D) = \frac{1}{2}\mathbb{E}_{(\mathbf{c},\mathbf{a})\sim p_{\mathbf{c},\mathbf{a}}(\mathbf{c},\mathbf{a})}\left[(D(\mathbf{c},\mathbf{a})-1)^2\right] + \frac{1}{2}\mathbb{E}_{\mathbf{a}\sim p_{\mathbf{a}}(\mathbf{a}),\boldsymbol{\phi}\sim p_{\boldsymbol{\phi}}(\boldsymbol{\phi})}\left[D(G(\mathbf{a},\boldsymbol{\phi}),\mathbf{a})^2\right]. \quad (8)$$

Here, the target label corresponding to real data is assumed to be 1 and that corresponding to the data generated by the generator G is 0. Thus, (8) means that $V(D)$ becomes 0 only if the discriminator D correctly distinguishes the "fake" complex spectrograms generated by the generator G and the "real" complex spectrograms of real audio signals. Therefore, the goal of D is to minimize $V(D)$. As for the generator G, one of the goals is to deceive the discriminator D so as to make the "fake" complex spectrograms as indistinguishable as possible from the "real" complex spectrograms. This can be accomplished by minimizing the following criterion

$$U(G) = \frac{1}{2}\mathbb{E}_{\mathbf{a}\sim p_{\mathbf{a}}(\mathbf{a}),\boldsymbol{\phi}\sim p_{\boldsymbol{\phi}}(\boldsymbol{\phi})}\left[(D(G(\mathbf{a},\boldsymbol{\phi}),\mathbf{a})-1)^2\right]. \quad (9)$$

Another goal for G is to make $\hat{\mathbf{c}} = G(\mathbf{a},\boldsymbol{\phi})$ as close as possible to the target complex spectrogram $\mathbf{c}$. By using $D_l(\cdot)$ to denote the output of the l-th layer of the discriminator D, we would also like G to minimize

$$I(G) = \sum_{l=0}^{L} w_l \|D_l(\mathbf{c}) - D_l(G(\mathbf{a},\boldsymbol{\phi}))\|_2^2, \quad (10)$$

where w_l is a fixed weight, which weighs the importance of the l-th layer feature space. Here, the 0-th layer corresponds to the input layer, namely $D_0(\mathbf{c}) = \mathbf{c}$.

The learning objectives for D and G can thus be summarized as follows:

$$D : V(D) \rightarrow \text{minimize}, \quad (11)$$

$$G : U(G) + \lambda I(G) \rightarrow \text{minimize}, \quad (12)$$

where λ is a fixed weight.

A general framework for training a generator network in such a way that it can deceive a real/fake discriminator network is called a generative adversarial network (GAN) [7]. The novelty of our proposed approach is that we have successfully adapted the GAN framework to the signal reconstruction problem by incorporating an additional term (10). The GAN framework using (8), (9) as the learning criteria is called the least squares GAN (LSGAN) [11]. Note that GAN frameworks using other learning criteria such as [12] have also been proposed. Thus, we can also use the learning criteria employed in [7], [12] or others instead of (8), (9).

V. Experimental Evaluation

We tested our method and the Griffin-Lim method using real speech samples.

A. Experimental Settings

1) Dataset: We used clean speech signals excerpted from [13] as the experimental data. The speech data consisted of utterances of 30 speakers. The utterances of 28 speakers were used as the training set and the remaining utterances were used as the evaluation set. For the mini-batch training, we divided each training utterance into 1-second-long segments with an overlap of 0.5 seconds. All the speech data were downsampled to 16 kHz. Magnitude spectrograms were obtained with an STFT using a Blackman window that was 64 ms long with a 32 ms overlap.

2) Network Architecture: Fig. 1 shows the network architectures we constructed for this experiment. The left half shows the architecture of the generator G and the right half shows that of the discriminator D. The light blue blocks indicate convolutional layers, and k, s, and c on each convolutional layer represent hyper-parameters. The yellow blocks indicate activation functions. PReLU [14] was used for the generator G and Leaky ReLU [15] was used for the discriminator D. The violet blocks indicate element-wise sums, and the green block indicates the concatenation of features along the channel axis. The red blocks indicate fully-connected layers. Blocks without symbols have the same hyper-parameters as the previous blocks. Note that we referred to [16] when constructing these architectures. The generator G is fully convolutional [17], thus allowing an input to have an arbitrary length. The weight constant w_l was set to 0 for $l = 0$ and 1 for $l \neq 0$. λ was set to 1. RMSprop [18] was used as the optimization algorithm and the learning rate was 5×10^{-5}C$\alpha = 0.5$. The mini-batch size was 10 and the number of epochs was 73.

Instead of directly feeding an input magnitude spectrogram and a randomly-generated phase spectrogram into the generator G, we used a complex spectrogram reconstructed using the Griffin-Lim method after 5 iterations as the G input. Both the input and output of the generator G have 2 channels, one corresponding to the real part and the other corresponding to the imaginary part of the complex spectrogram. For preprocessing, we normalized the complex spectrograms of the training data to obtain zero-mean and unit-variance at each frequency. At test time, the scale of the generator output at each frequency was restored.

We added a block that applies an inverse STFT to the generator output before feeding it into the discriminator D. We found this particularly important as the training did not work well without this block.

B. Data Augmentation

It is a well-known fact that the difference between signals is hardly perceptible to human ears when the magnitude spectrograms and the inter-frame phase differences are the same. This implies that there is an arbitrariness in the initial phases of spectrograms that are perceived similarly. By utilizing this property, we can augment the training data for G and D by preparing many different waveforms that are the same except for the initial phases. We expect that this data augmentation would allow the generator to concentrate on learning a way

ISBN 978-90-827970-1-5

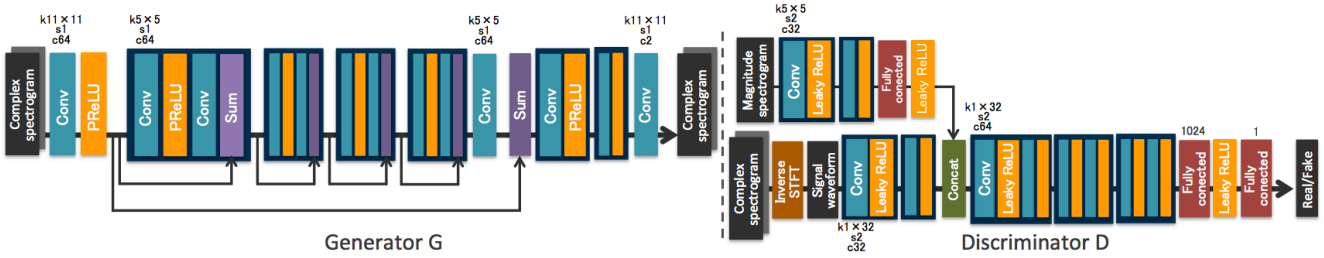

Fig. 1. Network architectures of generator and discriminator. Light blue blocks indicate convolutional layers. In each convolutional layer, k, s, and c represent kernel size, stride size, and number of channels, respectively. Here, k1 × ∗ indicates a one-dimensional convolutional layer whose kernel size is ∗. Red blocks indicate fully connected layer. In each fully connected layer, the numbers represents size of output unit.

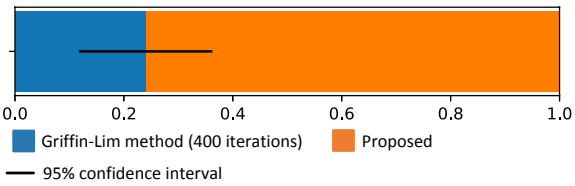

Fig. 2. Result of the AB test. The orange area indicates the rate of the A and B pairs for which the listeners preferred A (proposed). The black bar indicates the 95% confidence interval.

of inferring appropriate inter-frame phase differences given a magnitude spectrogram, thus facilitating efficient learning.

C. Dimensionality Reduction

Note that the real and imaginary parts of the Fourier transform of a real-valued signal become even and odd functions, respectively. Owing to this symmetric structure, it is sufficient to restore/infer spectral components within the frequency range from 0 up to the Nyquist frequency. We can therefore restrict the sizes of the input and output of the generator to this frequency range.

D. Subjective Evaluation

We compared our proposed method with the Griffin-Lim method in terms of the perceptual quality of reconstructed signals by conducting an AB test, where "A" and "B" were reconstructed signals obtained respectively with the proposed and baseline methods. With this listening test, "A" and "B" were presented in random orders to eliminate bias as regards the order of stimuli. Five listeners participated in our listening test. Each listener was presented with {"A","B"} ×10 signals and asked to select "A"or "B" for each pair. The Griffin-Lim method was run for 400 iterations. The signals were 2 to 5 seconds long.

The preference scores are shown in Fig. 2. As the result shows, the reconstructed signals obtained with the proposed method were preferred by the listeners for 76% of the 50 pairs.

E. Generalization ability

To confirm the generalization ability of the proposed method, we tested it on musical audio signals excerpted from [19]. Examples of the reconstructed signals are shown in Fig. 3. With these examples, we can observe a discontinuous point in the reconstructed signal obtained with the Griffin-Lim method. On the other hand, the proposed method appears to

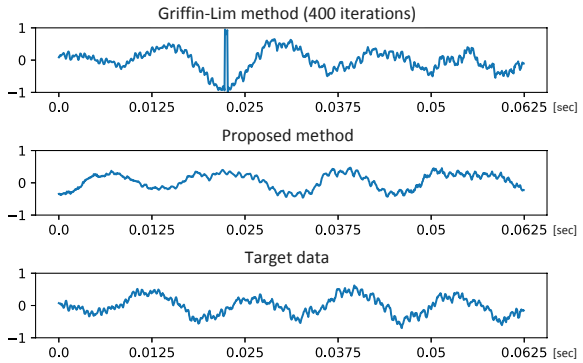

Fig. 3. Waveforms of reconstructed music data [19]. The first row shows the acoustic signal reconstructed with the Griffin-Lim method, the second shows the proposed method, and the third is the target acoustic signal (real-world acoustic signal).

have worked successfully, even though the model was trained using speech data.

F. Comparison of Processing Times

We further compared the proposed method with the Griffin-Lim method in terms of the processing times needed to reconstruct time domain signals. For comparison, we measured the processing times for various speech lengths. We used speech data shorter than 6 seconds for the evaluation. Here, the network architecture of our proposed method was the same as Fig. 1, and the Griffin-Lim method was run for 400 iterations. The CPU used in this experiment was "Intel Core i7-6850K CPU @ 3.60GHz". The GPU was "NVIDIA GeForce GTX 1080". We implemented the Griffin-Lim method using the fast Fourier transform function in NumPy [20]. We implemented our model with Chainer [21]. Fig. 4 shows the result. As the speech data become longer, the processing time increases linearly. When executing the proposed method using the GPU, the time needed to reconstruct a signal was only about one-tenth the length of that signal. On the other hand, the Griffin-Lim method executed using the CPU took about the same time as the length of the signal. Therefore, if we can use a GPU, the proposed method can be run in real time. However, when using the CPU, the proposed method took about three times longer than the length of the signal. If we want to execute the proposed method in real-time using a CPU, we would need to construct a more compact architecture than that shown in Fig. 1. One simple way would be to replace the convolutional layers with downsampling and upsampling layers.

ISBN 978-90-827970-1-5 © EURASIP 2018

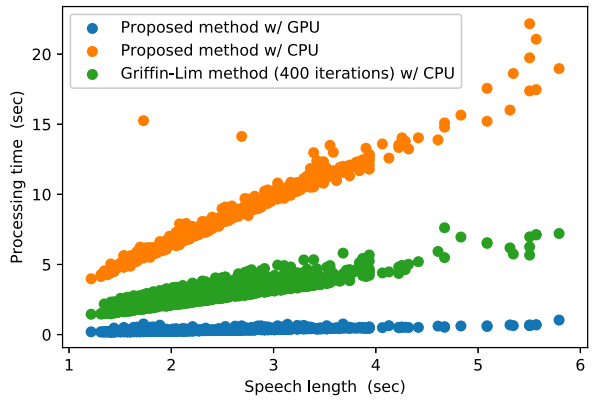

Fig. 4. The change in processing time with respect to the change in speech length. Blue points show the processing time with the proposed method a GPU. Green points show the time with the Griffin-Lim method with a CPU. Orange points show the time with the proposed method with a CPU.

VI. Conclusion

This paper proposed a GAN-based approach to signal reconstruction from magnitude spectrograms. The idea was to model the signal reconstruction process using a DNN and train it using a similarity metric implicitly learned using a GAN discriminator. Through subjective evaluations, we showed that the proposed method was able to reconstruct higher quality time domain signals than the Griffin-Lim method, which was run for 400 iterations. Furthermore, we showed that the proposed method can be executed in real-time when using a GPU. Future work will include the investigation of a network architecture appropriate for CPU implementations.

References

[1] P. Smaragdis, C. Févotte, G. J. Mysore, N. Mohammadiha, and M. Hoffman, "Static and dynamic source separation using nonnegative factorizations: A unified view," *IEEE Signal Processing Magazine*, vol. 31, no. 3, pp. 66–75, 2014.

[2] T. Virtanen, J. Florent Gemmeke, B. Raj, and P. Smaragdis, "Compositional models for audio processing: Uncovering the structure of sound mixtures," *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 125–144, 2015

[3] H. Kameoka, "Non-negative matrix factorization and its variants for audio signal processing," in *Applied Matrix and Tensor Variate Data Analysis*, T. Sakata (Ed.), Springer Japan, 2016.

[4] S. Takaki, H. Kameoka, and J. Yamagishi, "Direct modeling of frequency spectra and waveform generation based on phase recovery for DNN-based speech synthesis," in *Proc. Interspeech*, pp. 1128–1132, 2017.

[5] Y. Wang, RJ. Skerry-Ryan, D. Stanton, Y. Wu, R.J. Weiss, et al., "Tacotron: A fully end-to-end text-to-speech synthesis model," arXiv preprint arXiv:1703.10135, 2017.

[6] D. W. Griffin and J. S. Lim, "Signal estimation from modified short-time Fourier transform," *IEEE Trans. ASSP*, vol. 32, no. 2, pp. 236–243, 1984.

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, et al., "Generative adversarial nets," in *Adv. NIPS*, pp. 2672–2680, 2014.

[8] J. Le Roux, H. Kameoka, N. Ono, S. Sagayama, "Fast signal reconstruction from magnitude STFT spectrogram based on spectrogram consistency," in *Proc. DAFx*, pp. 397–403, 2010.

[9] D. R. Hunter and K. Lange, "Quantile regression via an MM algorithm," *Journal of Computational and Graphical Statistics*, vol. 9, pp. 60–77, 2000.

[10] J. R. Hershey, J. Le Roux, and F. Weninger, "Deep unfolding: Model-based inspiration of novel deep architectures," arXiv preprint arXiv:1409.2574.

[11] X. Mao, Q. Li, H. Xie, R.Y. Lau, Z. Wang, et al., "Least squares generative adversarial networks," in *Proc. ICCV*, pp. 2813–2821, 2017.

[12] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," arXiv preprint arXiv:1701.07875, 2017.

[13] C. Valentini-Botinhao, "Noisy speech database for training speech enhancement algorithms and TTS models, [dataset]," University of Edinburgh. School of Informatics. CSTR, 2016. http://dx.doi.org/10.7488/ds/1356.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proc. ICCV*, pp. 1026–1034, 2015.

[15] A.L. Maas, A.Y. Hannun, and A.Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. ICML*, vol. 30, no. 1, pp. 3, 2013.

[16] C. Ledig, L. Thesis, F. Huszár, J. Caballero, A. Cunningham, et al., "Photo-realistic single image super-resolution using a generative adversarial network," arXiv preprint arXiv:1609.04802, 2016.

[17] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in Proc. CVPR, pp. 3431–3440, 2015.

[18] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," COURSERA: Neural networks for machine learning, vol. 4, no. 2, pp. 26–31, 2012.

[19] CAFÉ DEL CHILLIA, "In The Story That We Say," https://www.jamendo.com/track/1455877/in-the-story-that-we-say, 2017.

[20] S. Walt, S.C. Colbert, and G. Varoquaux, "The NumPy array: a structure for efficient numerical computation," Computing in Science & Engineering, vol. 13, no. 2, pp. 22-30, 2011.

[21] S. Tokui, K. Oono, S. Hido, and J. Clayton, "Chainer: a next-generation open source framework for deep learning," in Proc. LearningSys in the twenty-ninth annual conference on NIPS, vol. 5, 2015.

ISBN 978-90-827970-1-5 © EURASIP 2018