Multi-SpectroGAN: High-Diversity and High-Fidelity Spectrogram Generation with Adversarial Style Combination for Speech Synthesis

Sang-Hoon Lee[1], **Hyun-Wook Yoon**[2], **Hyeong-Rae Noh**[1], **Ji-Hoon Kim**[3], **Seong-Whan Lee**[1,3]
[1]Department of Brain and Cognitive Engineering, Korea University, Seoul, Korea
[2]Department of Computer and Radio Communications Engineering, Korea University, Seoul, Korea
[3]Department of Artificial Intelligence, Korea University, Seoul, Korea
{sh_lee, hw_yoon, hr_noh, jihoon_kim, sw.lee}@korea.ac.kr

Abstract

While generative adversarial networks (GANs) based neural text-to-speech (TTS) systems have shown significant improvement in neural speech synthesis, there is no TTS system to learn to synthesize speech from text sequences with only adversarial feedback. Because adversarial feedback alone is not sufficient to train the generator, current models still require the reconstruction loss compared with the ground-truth and the generated mel-spectrogram directly. In this paper, we present Multi-SpectroGAN (MSG), which can train the multi-speaker model with only the adversarial feedback by conditioning a self-supervised hidden representation of the generator to a conditional discriminator. This leads to better guidance for generator training. Moreover, we also propose adversarial style combination (ASC) for better generalization in the unseen speaking style and transcript, which can learn latent representations of the combined style embedding from multiple mel-spectrograms. Trained with ASC and feature matching, the MSG synthesizes a high-diversity mel-spectrogram by controlling and mixing the individual speaking styles (e.g., duration, pitch, and energy). The result shows that the MSG synthesizes a high-fidelity mel-spectrogram, which has almost the same naturalness MOS score as the ground-truth mel-spectrogram.

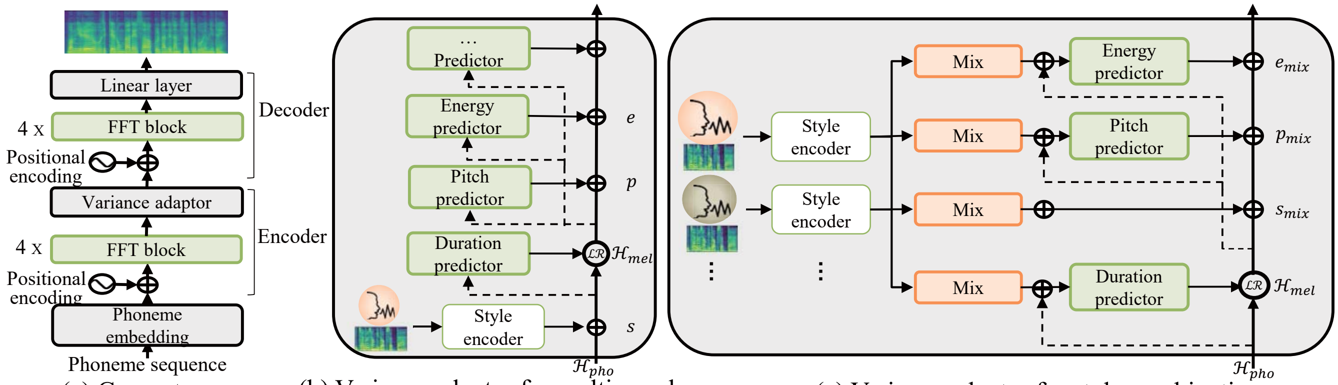
Introduction

Recently, there has been a significant progress in the end-to-end text-to-speech (TTS) model, which can convert a normal text into speech. When synthesizing speech, the recently proposed methods use additional speech audio as an input to reflect the style features from the input audio to the synthesized audio (Wang et al. 2018; Skerry-Ryan et al. 2018). However, there are limitations to transferring and controlling the style without a large amount of high-quality text-audio data (e.g., audiobook dataset). Moreover, because it is difficult to acquire high-quality data, some studies use the knowledge distillation method to improve the performance (Ren et al. 2019). However, knowledge distillation makes the training complicated, and the generated mel-spectrogram is not complete unlike the ground-truth mel-spectrogram (Ren et al. 2020).

For better generalization, the current models are trained with adversarial feedback. These generative adversarial networks (GANs) (Goodfellow et al. 2014) based TTS models demonstrate that adversarial feedback is important for learning to synthesize high-quality audio. MelGAN (Kumar et al. 2019) successfully converts mel-spectrograms to waveforms using a window-based discriminator. The Parallel WaveGAN (PWG) (Yamamoto, Song, and Kim 2020) also converts mel-spectrograms to raw waveforms using the adversarial feedback of audio with multi-resolution spectrogram losses. The GAN-TTS (Bińkowski et al. 2019) also generates raw speech audio with GANs conditioning features that are predicted by separate models. The EATS (Donahue et al. 2020) generates the raw waveform from raw phoneme inputs, which is learned end-to-end with various adversarial feedbacks and prediction losses. However, these methods have not yet learned the model without the prediction loss.

In this paper, we present the Multi-SpectroGAN (MSG), which can generate high-diversity and high-fidelity mel-spectrograms with adversarial feedback. We introduce an end-to-end learned frame-level condition and conditional discriminator to train the model without prediction loss between ground-truth and generated mel-spectrogram. By making the discriminator learn to distinguish which features are converted to mel-spectrogram with a frame-level condition, the generator is trained with frame-level adversarial feedback to synthesize high-fidelity mel-spectrograms. We also propose the adversarial style combination, which can learn the latent representations of mel-spectrograms synthesized with the mixed speaker embeddings. By training with adversarial feedback from the mixed-style mel-spectrogram, we demonstrate that the MSG synthesizes a more diverse mel-spectrogram by interpolation of multiple styles and synthesizes more natural audio of the unseen speaker. The main contributions of this study are as follows:

- Through an end-to-end learned frame-level condition and conditional discriminator, our model can learn to synthesize mel-spectrogram without prediction loss.
- We propose adversarial style combination, which learns the mixed style of mel-spectrogram with adversarial feedback.
- The MSG achieves a mean opinion score (MOS) of 3.90 with a small amount of multi-speaker data and almost the same MOS with ground-truth mel-spectrogram in single speaker model.



(a) Generator (b) Variance adaptor for multi-speaker (c) Variance adaptor for style combination

Figure 1: Generator and the variance adaptor architecture for style combination

Related Works

Text-To-Speech Autoregressive models such as Tacotron (Wang et al. 2017; Shen et al. 2018) were proposed to generate mel-spectrograms through an attention-based recurrent neural network (RNN) (Bulthoff et al. 2003). In this model, each frame is autoregressively generated through a sequential generative model conditioned on previously generated frames. However, this method is slow in inference, and it is difficult to model long-term dependencies, leading to word skipping or repetition problems.

To solve these problems, several non-autoregressive models have been proposed for faster generation. FastSpeech (Ren et al. 2019) adapted a feed-forward block from Transformer (Vaswani et al. 2017) with a self-attention mechanism to perform parallel generation. In addition, the model implemented a length regulator to properly match the character-level sequence with the frame-level sequence. FastSpeech2 (Ren et al. 2020) strengthens their model with additional variance information to predict acoustic features more accurately. In FastPitch (Łańcucki 2020), the author cascades fundamental frequency on the phoneme hidden representation (Lee and Kim 1999; Yang and Lee 2007).

With the improved performance of the speech synthesis model, several models have been proposed to control the speaking style of generated speech. One well-known method is the global style token (GST) (Wang et al. 2018), which makes the model learn a prosodic aspect of the variable-length audio signal through several style tokens without any style label. A variational autoencoder (VAE)-based style control model (Zhang et al. 2019) was also proposed while maintaining unsupervised learning in style features.

In the Transformer-based TTS model (Li et al. 2019), training a model with various speakers is challenging because of the difficulty in learning the text-to-speech alignment. (Li et al. 2020; Chen et al. 2020) identified that the limitation of using location-sensitive attention in the parallel computational model pose a difficulty for the Transformer-based model to learn the alignment between the linguistic and acoustic features. To solve this issue, (Chen et al. 2020) used diagonal constraints in encoder-decoder attention to make the model forcefully learn the diagonal area.

Waveform Generation Most speech synthesis models generate intermediate features such as mel-spectrograms to reduce computational time. Therefore, an additional module, named 'vocoder', is needed to generate a fully audible signal. In an autoregressive model such as Wavenet (Oord et al. 2016), each audio sample is generated sequentially, usually conditioned on previous samples. In general, an RNN-based vocoder, such as bidirectional-RNN or gated recurrent unit (GRU) is used; therefore, the model can predict each sample precisely without long-range constraint dependency. However, owing to the sequential generation process, the overall inference time is slow. Therefore, generating audio samples simultaneously is necessary.

For parallel generation models, non-autoregressive generation methods such as knowledge distillation (Oord et al. 2018) and flow-based generative models (Prenger, Valle, and Catanzaro 2019; Kim et al. 2018) have been proposed. These models can generate audio samples in parallel, but they suffer from relatively degraded generation quality. Therefore, the issue of improving audio quality has arisen in the parallel generation model. (Yoon et al. 2020). Recently, the use of GANs (Yamamoto, Song, and Kim 2020) to generate high-quality audio in real-time has shown remarkable performance in the field. However, the problem remains when the model is extended to the multi-speaker domain. Therefore, reducing inference time while maintaining audio quality is still a challenging task. Several attempts have been made to fully generate audio waveforms from text input. (Bińkowski et al. 2019) used various linguistic features including duration and pitch information, to produce high-fidelity audio. (Donahue et al. 2020) proposed a novel aligner, which can align between text and mel-frames in parallel.

Mixup Mixup was proposed to regularize the neural networks by training the model on convex combination of example-label pairs (Zhang et al. 2017). (Verma et al. 2019) proposed training the model on interpolations of hidden representation. The method for learning combined latent representation of autoencoder was proposed (Beckham et al. 2019). These methods improve the model to generalize for new latent representation which are not seen during training.

Multi-SpectroGAN

Our goal is to learn a generator which can synthesize high-diversity and high-fidelity mel-spectrograms by controlling and mixing the speaking style. For high-diversity mel-spectrograms, we introduce an adversarial style combination which can learn latent representations of the combined speaker embedding from multiple mel-spectrograms. To learn the generated mel-spectrogram with randomly mixed styles which doesn't have a ground truth mel-spectrogram, we propose an end-to-end learned frame-level conditional discriminator. It is also important for better guidance to make the model learn to synthesize speech with only adversarial feedback. We describe the details of the Multi-SpectroGAN architecture and adversarial style combination in the following subsections.

Generator

We use FastSpeech2 (Ren et al. 2020) as a generator consisting of a phoneme encoder with the variance adaptor denoted as $f(\cdot,\cdot)$, and decoder $g(\cdot)$. We use the phoneme encoder and decoder which consists of 4 feed-forward Transformer (FFT) blocks. Extending to the multi-speaker model, we introduce a style encoder that can produce a fixed-dimensional style vector from a mel-spectrogram like Figure 1.

Style Encoder The style encoder has a similar architecture to the prosody encoder of (Skerry-Ryan et al. 2018). Instead of 2D convolutional network with 3×3 filters and 2×2 stride, our style encoder uses a 6-layer 1D convolutional network with 3×1 filters and 2×2 stride, dropout, ReLU activation, and Layer normalization (Ba, Kiros, and Hinton 2016). We also use a gated recurrent unit (Cho et al. 2014) layer and take the final output to compress the length down to a single style vector. Before conditioning the length regulator and variance adaptor, the output is projected as the same dimension of the phoneme encoder output to add style information, followed by a tanh activation function. We denote the style encoder as $E_s(\cdot)$, which produces the style embedding

$$\boldsymbol{s} = E_s(\boldsymbol{y}), \tag{1}$$

where $\boldsymbol{s}$ refers to the style embedding extracted from the mel-spectrogram $\boldsymbol{y}$ through the style encoder E_s.

Style-conditional Variance Adaptor With the exception of using style conditional information for learning the multi-speaker model, we use the same variance adaptor of FastSpeech2 (Ren et al. 2020) to add variance information. By adding the style embedding predicted from the mel-spectrogram to the phoneme hidden sequence $\mathcal{H}_{pho}$, the variance adaptor predicts each variance information with the unique style of each speaker. For details, we denote the phoneme-side FFT networks as phoneme encoder $E_p(\cdot)$, which produces the phoneme hidden representation

$$\mathcal{H}_{pho} = E_p(\boldsymbol{x} + PE(\cdot)), \tag{2}$$

where $\boldsymbol{x}$ is the phoneme embedding sequence, and $PE(\cdot)$ is a triangle positional embedding (Li et al. 2019) for giving positional information to the Transformer networks. We extract the target duration sequences $\boldsymbol{\mathcal{D}}$ from Tacotron2 to map the length of the phoneme hidden sequence to the length of the mel-spectrogram

$$\mathcal{H}_{mel} = \mathcal{LR}(\mathcal{H}_{pho}, \boldsymbol{\mathcal{D}}). \tag{3}$$

The duration predictor predicts the log-scale of the length with the mean-square error (MSE)

$$\mathcal{L}_{Duration} = \mathbb{E}[\|log(\boldsymbol{\mathcal{D}} + 1) - \hat{\boldsymbol{\mathcal{D}}}\|_2], \tag{4}$$

where

$$\hat{\boldsymbol{\mathcal{D}}} = DurationPredictor(\mathcal{H}_{pho}, \boldsymbol{s}). \tag{5}$$

We also use the target pitch sequences $\boldsymbol{\mathcal{P}}$ and target energy sequences $\boldsymbol{\mathcal{E}}$ for each mel-spectrogram frame. We remove the outliers of each information and use the normalized value. Then we add the embedding of quantized $F0$ and energy sequences, $\boldsymbol{p}$ and $\boldsymbol{e}$, which are divided by 256 values.

$$\boldsymbol{p} = PitchEmbedding(\boldsymbol{\mathcal{P}}),\ \boldsymbol{e} = EnergyEmbedding(\boldsymbol{\mathcal{E}}). \tag{6}$$

The pitch/energy predictor predicts the normalized $F0$/energy value with the MSE between the ground-truth $\boldsymbol{\mathcal{P}}$, $\boldsymbol{\mathcal{E}}$ and the predicted $\hat{\boldsymbol{\mathcal{P}}}$, $\hat{\boldsymbol{\mathcal{E}}}$

$$\begin{aligned} \mathcal{L}_{Pitch} &= \mathbb{E}[\|\boldsymbol{\mathcal{P}} - \hat{\boldsymbol{\mathcal{P}}}\|_2], \\ \mathcal{L}_{Energy} &= \mathbb{E}[\|\boldsymbol{\mathcal{E}} - \hat{\boldsymbol{\mathcal{E}}}\|_2], \end{aligned} \tag{7}$$

where

$$\begin{aligned} \hat{\boldsymbol{\mathcal{P}}} &= PitchPredictor(\mathcal{H}_{mel}, \boldsymbol{s}), \\ \hat{\boldsymbol{\mathcal{E}}} &= EnergyPredictor(\mathcal{H}_{mel}, \boldsymbol{s}). \end{aligned} \tag{8}$$

The encoder $f(\cdot,\cdot)$ consisting of a phoneme encoder and style-conditional variance adaptor is trained with the variance prediction loss

$$\min_f \mathcal{L}_{var} = \mathcal{L}_{Duration} + \mathcal{L}_{Pitch} + \mathcal{L}_{Energy}. \tag{9}$$

During training, we use not only the ground-truth value of each information, such as (Ren et al. 2020), but also the predicted value of each information with adversarial style combination to learn the variety of generated mel-spectrograms without the ground-truth. The sum of each informational hidden sequence $\mathcal{H}_{total}$ is passed to the decoder as a generator $g(\cdot)$ to generate a mel-spectrogram as

$$\mathcal{H}_{total} = \mathcal{H}_{mel} + \boldsymbol{s} + \boldsymbol{p} + \boldsymbol{e} + PE(\cdot), \tag{10}$$

$$\hat{\boldsymbol{y}} = g(\mathcal{H}_{total}), \tag{11}$$

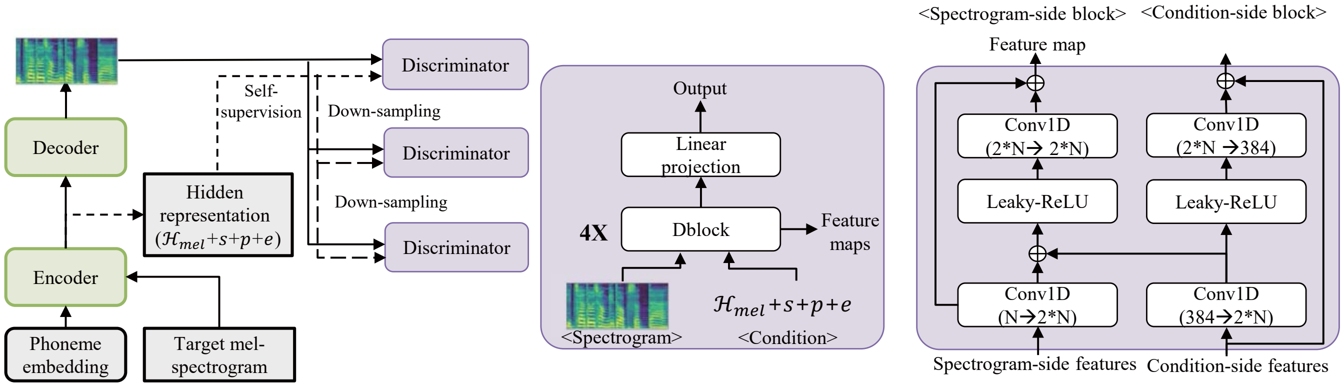
where $\hat{\boldsymbol{y}}$ is the predicted mel-spectrogram. Our baseline models use the reconstruction loss with mean-absolute error (MAE) as

$$\mathcal{L}_{rec} = \mathbb{E}[\|\boldsymbol{y} - \hat{\boldsymbol{y}}\|_1], \tag{12}$$

where $\boldsymbol{y}$ is the ground-truth mel-spectrogram.

Discriminator

Unlike the previous GAN-based TTS model, our model can be learned to synthesize the mel-spectrogram from a text sequence without calculating the loss compared with the ground-truth spectrogram directly. To train the model without $\mathcal{L}_{rec}$, we design a frame-level conditional discriminator using the end-to-end learned frame-level condition.



(a) Multi-SpectroGAN (b) Discriminator (c) Dblock

Figure 2: Frame-level conditional discriminator. Each discriminator has 4 Dblocks consisting of spectrogram-side block and condition-side block. Each side has two non-strided 1D convolutional networks with kernel size of 3. Conditional hidden states are added to spectrogram-side hidden states by the same filter size after first convolutional layer.

End-To-End Learned Frame Level Condition To learn to distinguish between the frame-level real and generated mel-spectrogram, the discriminator uses the encoder outputs as a frame-level condition that is learned in a generator during training. Note that $\boldsymbol{c}$ is the sum of linguistic, style, pitch, and energy information, which is end-to-end learned in a generator during training and is expressed as:

$$\boldsymbol{c} = \underbrace{\mathcal{H}_{mel}}_{\text{linguistic}} + \underbrace{\boldsymbol{s}}_{\text{style}} + \underbrace{\boldsymbol{p}}_{\text{pitch}} + \underbrace{\boldsymbol{e}}_{\text{energy}}. \tag{13}$$

Frame-level Conditional Discriminator As shown in Figure 2, we adopt a multi-scale discriminator that has identical network structure like MelGAN (Kumar et al. 2019). While MelGAN motivates the multiple discriminators at different scales to learn features for the different frequency ranges of the audio, we choose multiple discriminators to learn features for different ranges of linguistic, pitch, and energy information. Each discriminator consists of 4 Dblocks that have a mel-spectrogram side block and a condition side block. Each block uses a 2-layer non-strided 1D convolutional network with the Leaky-ReLU activation function to extract the adjacent frame information. We add the hidden representation of the condition side block to the mel-spectrogram side hidden representation. Similar to (Vaswani et al. 2017), residual connections and layer normalization is used at each block output for optimization.

We use the least-squares GAN (LSGAN) (Mao et al. 2017) formulation to train the Multi-SpectroGAN. The discriminators D_k learn to distinguish between real spectrogram $\boldsymbol{y}$ and reconstructed one from $\boldsymbol{x}$, $\boldsymbol{y}$. We minimize the GAN loss from the mel-spectrogram. The encoder $f(\cdot,\cdot)$ and decoder $g(\cdot)$ as a generator, and discriminator D are trained by the following losses:

$$\min_{D_k} \mathbb{E}[\|D_k(\boldsymbol{y},\boldsymbol{c})-1\|_2+\|D_k(\hat{\boldsymbol{y}},\boldsymbol{c})\|_2], \forall k=1,2,3 \tag{14}$$

$$\mathcal{L}_{adv} = \mathbb{E}\left[\sum_{k=1}^{3}\|D_k(\hat{\boldsymbol{y}},\boldsymbol{c})-1\|_2\right]. \tag{15}$$

Feature Matching To improve the representations learned by the discriminator, we use a feature matching objective like (Kumar et al. 2019). Unlike the MelGAN, which minimizes the MAE between the discriminator feature maps of real and generated audio, we minimize the MAE between the feature maps of each spectrogram-side block:

$$\mathcal{L}_{fm} = \mathbb{E}\left[\sum_{i=1}^{4}\frac{1}{N_i}\|D_k^{(i)}(\boldsymbol{y},\boldsymbol{c})-D_k^{(i)}(\hat{\boldsymbol{y}},\boldsymbol{c})\|_1\right], \tag{16}$$

where $D_k^{(i)}$ refers to the i^{th} spectrogram-side block output of the k^{th} discriminator, and N_i is the number of units in each block output. The generator trains with the following objective:

$$\min_{f,g} \mathcal{L}_{msg} = \mathcal{L}_{adv} + \lambda\mathcal{L}_{fm} + \mu\mathcal{L}_{var}. \tag{17}$$

Adversarial Style Combination

By introducing the adversarial loss, we would like to synthesize a more realistic audio signal with high-fidelity generated mel-spectrogram. In addition, our goal is to generate a more diverse audio signal with an even unseen style. To do this, we propose the adversarial style combination (ASC), which can make the mel-spectrogram more realistic with the mixed style of multiple source speakers. Similar to (Beckham et al. 2019) interpolating the hidden state of the autoencoder for adversarial mixup resynthesis, we use two types of mixing, binary selection between style embeddings, and manifold mixup (Verma et al. 2019) by the linear combination of style embeddings from the different speakers:

$$s_{mix} = \alpha\boldsymbol{s}_i + (1-\alpha)\boldsymbol{s}_j, \tag{18}$$

where $\alpha \in \{0,1\}$ is sampled from a Bernoulli distribution in binary selection and $\alpha \in [0,1]$ is sampled from the Uniform(0,1) distribution in manifold mixup. The variance adaptor predicts each information with a mixed style embedding. Unlike pitch and energy, we use the ground-truth $\mathcal{D}$ randomly selected from multiple source speakers because

Model	MOS	95% CI
GT	4.20	± 0.03
GT (Mel + PWG)	3.94	± 0.03
Transformer TTS (Mel + PWG)	3.83	± 0.03
FastSpeech (Mel + PWG)	3.52	± 0.04
FastSpeech2 (Mel + PWG)	3.85	± 0.03
MSG (Mel + PWG)	3.91	± 0.03

Table 1: MOS with 95% CI for a single speaker model

the duration predictor may predict the wrong duration at the early training step. Each variance information is predicted by different ratios of mixed style embedding. We call it "style combination", in which the final mixed hidden representation is the combination of each variance information from different mixed styles:

$$\mathcal{H}_{mix} = \underbrace{\mathcal{H}_{mel} + \boldsymbol{s}_{mix} + \boldsymbol{p}_{mix} + \boldsymbol{e}_{mix}}_{\boldsymbol{c}_{mix}} + PE(\cdot), \tag{19}$$

$$\hat{\boldsymbol{y}}_{mix} = g(\mathcal{H}_{mix}), \tag{20}$$

where $\boldsymbol{p}_{mix}$ and $\boldsymbol{e}_{mix}$ are the pitch and energy embedding of the predicted value from mixed styles, respectively, and $\boldsymbol{c}_{mix}$ is fed to discriminator as the frame-level condition for mel-spectrogram $\hat{\boldsymbol{y}}_{mix}$ generated by style combination. The discriminator is trained using the following objective:

$$\begin{aligned}\min_{D_k} \mathbb{E}[\|D_k(\boldsymbol{y}, \boldsymbol{c}) - 1\|_2 + &\|D_k(\hat{\boldsymbol{y}}, \boldsymbol{c})\|_2 \\ &+ \|D_k(\hat{\boldsymbol{y}}_{mix}, \boldsymbol{c}_{mix})\|_2], \forall k = 1, 2, 3.\end{aligned} \tag{21}$$

The generator is trained by the following loss:

$$\min_{f,g} \mathcal{L}_{asc} = \mathcal{L}_{adv} + \lambda \mathcal{L}_{fm} + \mu \mathcal{L}_{var} + \nu \mathcal{L}_{mix}, \tag{22}$$

where

$$\mathcal{L}_{mix} = \mathbb{E}\left[\sum_{k=1}^{3} \|D_k(\hat{\boldsymbol{y}}_{mix}, \boldsymbol{c}_{mix}) - 1\|_2\right]. \tag{23}$$

Experiments and Results

We evaluated in the single-speaker and multi-speaker dataset. Ablation studies are performed for downsampling size, loss function, and conditional information. We also evaluated the style-combined speech by control and interpolation of multiple styles. We used a Nvidia Titan V to train the single-speaker model with the LJ-speech dataset and the multi-speaker model with the VCTK dataset. Each dataset is split into train, validation, and test. Mel-spectrogram is transformed following the work of (Shen et al. 2018) with a window size of 1024, hop size of 256, 1024 points of Fourier transform, and 22,050 Hz sampling rate. We use the ADAM (Kingma and Ba 2015) optimizer with β_1 = 0.9, β_2 = 0.98, and ϵ = 10^{-9}, and apply the same learning rate schedule as that of (Vaswani et al. 2017) with an initial learning rate of 10^{-4} for f, g, and D. The λ, μ, and ν are set to 10, 1 and 1. To convert the mel-spectrogram to audio, we use the pretrained PWG vocoder (Yamamoto, Song, and Kim 2020) consisting of 30-layers of dilated residual convolution blocks.

Model	τ	CMOS	Convergence
MSG	2	0	350k
MSG	3	+0.07	650k
MSG	4	+0.06	1,000k

Table 2: CMOS comparison for the down-sampling size

Model	Loss function	MOS
FastSpeech2	$\mathcal{L}_{var}$+$\mathcal{L}_{rec}$	3.85 ± 0.03
MSG (w/o $\boldsymbol{c}$)	$\mathcal{L}_{var}$+$\mathcal{L}_{adv}$	-
MSG (w/ $\boldsymbol{c}$)	$\mathcal{L}_{var}$+$\mathcal{L}_{adv}$	3.14 ± 0.06
MSG (w/ $\boldsymbol{c}$)	$\mathcal{L}_{var}$+$\mathcal{L}_{adv}$+$\mathcal{L}_{rec}$	3.85 ± 0.03
MSG (w/ $\boldsymbol{c}$)	$\mathcal{L}_{var}$+$\mathcal{L}_{adv}$+$\mathcal{L}_{rec}$+$\mathcal{L}_{fm}$	3.89 ± 0.03
MSG (w/ $\boldsymbol{c}$)	$\mathcal{L}_{var}$+$\mathcal{L}_{adv}$+$\mathcal{L}_{fm}$	3.91 ± 0.03

Table 3: Ablation study for the loss function

Single-speaker Speech Synthesis

Naturalness MOS To evaluate the quality of the synthesized mel-spectrogram, we conducted a subjective MOS test. We randomly selected 100 sentences from the test dataset. The audio generated from each model was sent to Amazon's Mechanical Turk (MTurk). Samples were evaluated by 20 raters on a scale from 1 to 5 with 0.5 point increments. We compared the MSG model with the ground-truth audio (GT), the converted audio from the mel-spectrogram of the GT, and other TTS models using PWG. As shown in Figure 1, the MOS results show that the MSG has an almost similar score to the ground-truth mel-spectrogram, which demonstrates our discriminator and the frame-level conditional information improves voice quality even though the same generator architecture (Ren et al. 2020) is used.

Down-sampling Size We use average pooling with different kernel sizes to compare downsampling size τ. The model with a downsampling size of 3 has the highest score. The smaller size of downsampling makes the model converge early step with a -0.07 CMOS score. The larger size of the downsampling causes the model to converge slowly but shows a similar MOS. Therefore, we adopted a downsampling size of 3 for our MSG model.

Loss Function We conducted the ablation study for the loss functions and the conditional discriminator. When the conditional information of the discriminator is replaced with z noise and trained with the loss function of $\mathcal{L}_{var}$ and $\mathcal{L}_{adv}$, this model does not train at all. On the other hand, the model using conditional information in the discriminator can learn to synthesize the mel-spectrogram without $\mathcal{L}_{rec}$ or $\mathcal{L}_{fm}$ which must be calculated between the ground-truth and generated mel-spectrogram. This demonstrates that the frame-level conditional discriminators using the end-to-end learned frame-level condition make it possible to train the model even if the generated mel-spectrogram does not have ground-truth audio. However, we also use the additional loss function $\mathcal{L}_{rec}$ or $\mathcal{L}_{fm}$ to improve the audio quality. Although most TTS models train with $\mathcal{L}_{rec}$, it is too strong supervision

Model	Mix	ratio	MOS	MCD_{13}	F_0 RMSE	Top-1 acc.
GT	-	-	4.11±0.03	-	-	93%
GT (Mel + PWG)	-	-	4.00±0.03	4.46	43.59	84%
Tacotron2 (Mel + PWG)	-	-	3.81±0.04	5.88	44.51	75%
GST (Mel + PWG)	-	-	3.89±0.04	5.59	45.10	**80%**
FastSpeech2 (Mel + PWG)	-	-	3.81±0.04	5.78	46.90	67%
MSG (Mel + PWG)	-	-	3.89±0.04	5.59	45.71	72%
MSG+ASC (Mel + PWG)	Bern	$\{r, r, r, ...\}$	3.85±0.04	**5.54**	45.36	70%
MSG+ASC (Mel + PWG)	Mixup	$\{r, r, r, ...\}$	3.89±0.04	5.60	45.31	69%
MSG+ASC (Mel + PWG)	Bern	$\{r_s, r_p, r_e, ...\}$	3.87±0.04	5.57	47.06	**79%**
MSG+ASC (Mel + PWG)	Mixup	$\{r_s, r_p, r_e, ...\}$	**3.90±0.04**	5.57	**43.97**	73%

Table 4: Results of subjective and objective tests for seen speaker. Bern refers that the ratio is sampled from a Bernoulli distribution. Mixup refers that the ratio is sampled from the uniform (0,1) distribution. We compare the models with same ratios $\{r, r, r,...\}$ and different ratios for mixing the style and each variance $\{r_s, r_p, r_e,...\}$ where r_s, r_p, and r_e are the ratios for mixing the style, pitch, and energy embeddings respectively.

Model	Mix	ratio	MOS	MCD_{13}	F_0 RMSE	Top-1 acc.
GT	-	-	4.00±0.03	-	-	95%
GT (Mel + PWG)	-	-	3.96±0.03	4.26	49.56	88%
Tacotron2 (Mel + PWG)	-	-	3.76±0.04	6.33	46.26	17%
GST (Mel + PWG)	-	-	**3.83±0.04**	6.15	41.71	5%
FastSpeech2 (Mel + PWG)	-	-	3.67±0.04	6.18	48.31	20%
MSG (Mel + PWG)	-	-	3.80±0.04	6.10	48.02	23%
MSG+ASC (Mel + PWG)	Bern	$\{r, r, r, ...\}$	3.80±0.04	6.11	**47.04**	**30%**
MSG+ASC (Mel + PWG)	Mixup	$\{r, r, r, ...\}$	**3.82±0.04**	**6.07**	47.69	27%
MSG+ASC (Mel + PWG)	Bern	$\{r_s, r_p, r_e, ...\}$	3.75±0.04	6.14	48.10	28%
MSG+ASC (Mel + PWG)	Mixup	$\{r_s, r_p, r_e, ...\}$	3.81±0.04	6.08	47.22	**30%**

Table 5: Results of subjective and objective tests for unseen speaker.

to train with adversarial loss; therefore, adversarial loss has a slight influence on the model. Unlike $\mathcal{L}_{rec}$, the $\mathcal{L}_{fm}$ is affected by the discriminator, and it shows the highest MOS score when the model was trained with $\mathcal{L}_{fm}$.

Multi-speaker Speech Synthesis

We trained each model using 30 speakers in the VCTK dataset. We evaluated each model with "seen speaker" and "unseen speaker" of reference audio for style. The "seen speaker" of reference audio indicates the audio of the speaker seen during training. The "unseen speaker" of reference audio indicates the audio of the speaker unseen during training, which is evaluated for the zero-shot style transfer. Audio samples of the generated speech are provided.[1]

Naturalness MOS For the subjective MOS test of each multi-speaker model, we randomly selected 40 speakers (20 seen and 20 unseen speakers) and 5 sentences from a test dataset of each speaker. The samples were evaluated by 20 raters on a scale of 1-5 with 0.5 point increments through Amazon MTurk. We compared our models with GT, the converted audio from the mel-spectrogram of the GT, and other TTS models (Tacotron2, GST, Tansformer-based TTS, and FastSpeech2). For multi-speaker Tacotron2, we add the style encoder and concatenate with the transcript embedding. In a Transformer-based TTS model, it is not possible to synthesize any audio because of the wrong alignment. For multi-speaker FastSpeech2, we train the model with the same style encoder and add the style embedding to transcript embedding. Even though using the same generator structure with FastSpeech2, the results show our method improves the audio quality of 0.08 for seen speaker and 0.13 for unseen speaker. When trained with ASC, the models have better performance on both the seen and unseen speakers.

Objective Evaluation We conducted an objective evaluation using mel-cepstral distortion (MCD) (Kubichek 1993), F_0 root mean squared error (RMSE), and speaker classification (Wan et al. 2018). To evaluate each metric, each model synthesized 100 utterances for both the seen and unseen speaker. For comparison of F_0 RMSE, we used target duration for FastSpeech2 and our models, and teacher-forcing synthesis with target mel-spectrogram for Tacotron2 and GST. Even though the GST shows the highest MOS score in the unseen speaker, the top-1 speaker classification accuracy is 5%, where the GST only synthesizes the learned voice during training. When the model is trained with ASC, the results verify that learning the combined latent representation in training makes the model synthesize a more diversed mel-spectrogram even for unseen speakers.

[1] https://anonymsg.github.io/MSG/Demo/index.html

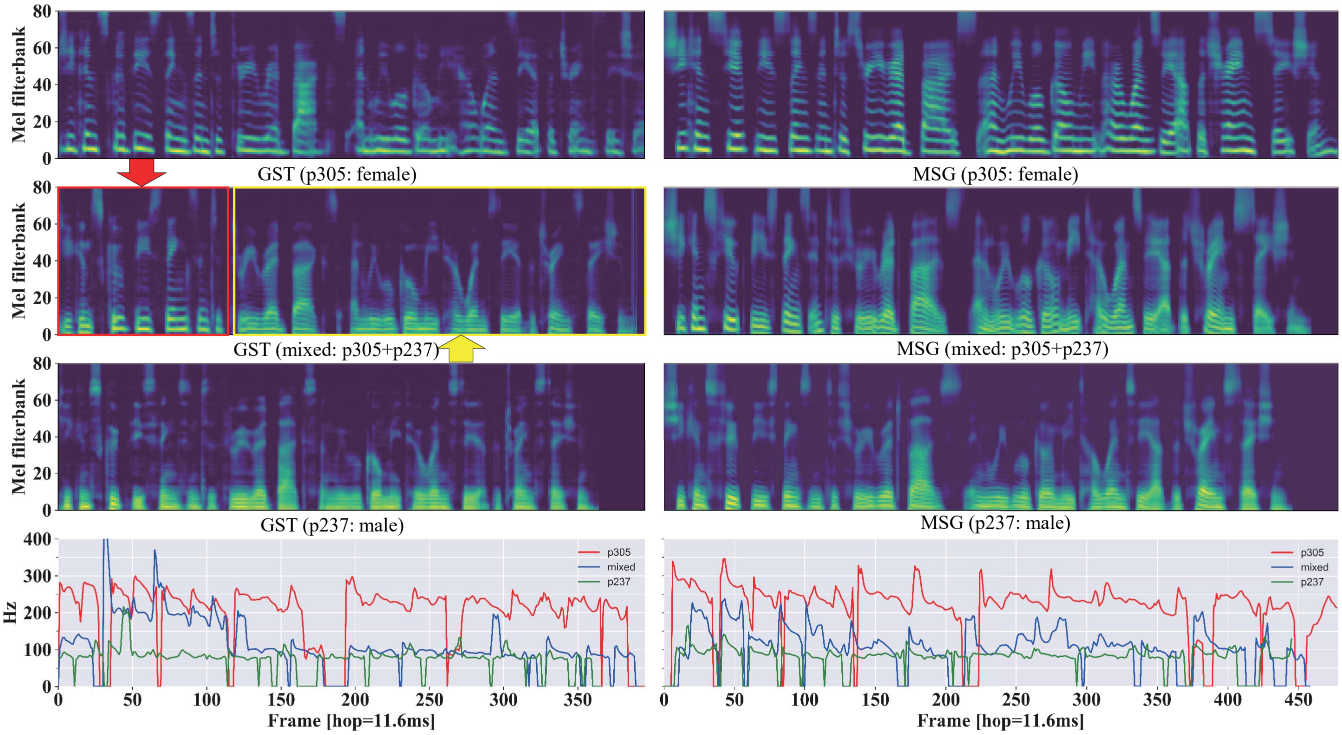


Figure 3: Mel-spectrogram and F_0 contour of the GST (Left) and MSG (Right).

Condition	Loss function	MOS
MSG ($\boldsymbol{c}$)	$\mathcal{L}_{var} + \mathcal{L}_{adv}$	3.57± 0.07
$-\mathcal{H}_{mel}$	$\mathcal{L}_{var} + \mathcal{L}_{adv}$	[does not train]
$-\boldsymbol{s} - \boldsymbol{p}$	$\mathcal{L}_{var} + \mathcal{L}_{adv}$	[does not train]
$-\boldsymbol{p}$	$\mathcal{L}_{var} + \mathcal{L}_{adv}$	3.52± 0.07
$-\boldsymbol{e}$	$\mathcal{L}_{var} + \mathcal{L}_{adv}$	3.54± 0.07

Table 6: Ablation study for condition of discriminator

Ablation Study We conducted an ablation study for the conditions in the discriminator. To evaluate the effectiveness of each conditional information, we trained the model without $\mathcal{L}_{fm}$. The model without $\mathcal{H}_{mel}$ does not train at all, which demonstrates that linguistic information is essential to learn to synthesize the frame-level mel-spectrogram. Unlike a single-speaker model that can learn to synthesize without style $\boldsymbol{s}$ or pitch $\boldsymbol{p}$ information, the multi-speaker model without $\boldsymbol{s}$ and $\boldsymbol{p}$ does not train at all. The model without $\boldsymbol{p}$ and $\boldsymbol{e}$ shows that each information has an effect on naturalness.

Style Combination

For the robustness of style transfer and control, we synthesize the mel-spectrogram with mixed style embedding which are interpolated style embedding of two speakers (1 male and 1 female). Figure 3 shows the mel-spectrograms and $F0$ contour (women, mixed and men style embedding) of GST (Left) and MSG (Right) model for the same sentence. The attention-based autoregressive models have some problems. Even when using an unseen and mixed style, the models synthesize a mel-spectrogram with a seen style during training. In addition, the change in the voice occurs at the same utterance as in Figure 3. Even in most cases, word skipping and repetition occur because the models fail to align.

Unlike attention-based autoregressive models, the MSG model trained with adversarial style combination synthesizes the mel-spectrogram robustly even with mixed-style embedding. The results demonstrate that the synthesis with the interpolated style embedding can generate a new style of mel-spectrogram by a combination of two styles. We also synthesized a particular style of a mel-spectrogram in combination with the desired proportions of each variance information (e.g., duration, pitch, and energy).

Conclusion and Future Work

We presented a Multi-SpectroGAN, which can generate high-diversity and high-fidelity mel-spectrograms with adversarial style combination. We demonstrated that it is possible to train the model with only adversarial feedback by conditioning a self-supervised latent representation of the generator to the discriminator. Our results also showed the effectiveness of mixing hidden states in the audio domain, which can learn the mel-spectrogram generated from a combination of mixed latent representations. By exploring various style combination for mixup, we show that learning the mel-spectrogram of mixed style made the model generalize better even in the case of unseen transcript and unseen speaker. For future work, we will train the Multi-SpectroGAN with few-shot learning and cross-lingual style transfer frameworks.

Acknowledgments

This work was supported by Institute for Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2019-0-01371, Development of brain-inspired AI with human-like intelligence & No. 2019-0-00079, Artificial Intelligence Graduate School Program, Korea University), Netmarble AI Center, and the Seoul R&BD Program (CY190019).

References

Ba, J. L.; Kiros, J. R.; and Hinton, G. E. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450* .

Beckham, C.; Honari, S.; Verma, V.; Lamb, A. M.; Ghadiri, F.; Hjelm, R. D.; Bengio, Y.; and Pal, C. 2019. On adversarial mixup resynthesis. In *Advances in Neural Information Processing Systems*, 4346–4357.

Bińkowski, M.; Donahue, J.; Dieleman, S.; Clark, A.; Elsen, E.; Casagrande, N.; Cobo, L. C.; and Simonyan, K. 2019. High fidelity speech synthesis with adversarial networks. *arXiv preprint arXiv:1909.11646* .

Bulthoff, H. H.; Lee, S.-W.; Poggio, T.; and Wallraven, C. 2003. *Biologically motivated computer vision*. Springer-Verlag.

Chen, M.; Tan, X.; Ren, Y.; Xu, J.; Sun, H.; Zhao, S.; and Qin, T. 2020. MultiSpeech: Multi-Speaker Text to Speech with Transformer. *arXiv preprint arXiv:2006.04664* .

Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* .

Donahue, J.; Dieleman, S.; Bińkowski, M.; Elsen, E.; and Simonyan, K. 2020. End-to-End Adversarial Text-to-Speech. *arXiv preprint arXiv:2006.03575* .

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2672–2680.

Kim, S.; Lee, S.-g.; Song, J.; Kim, J.; and Yoon, S. 2018. FloWaveNet: A generative flow for raw audio. *arXiv preprint arXiv:1811.02155* .

Kingma, D. P.; and Ba, J. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*.

Kubichek, R. 1993. Mel-cepstral distance measure for objective speech quality assessment. In *Proceedings of IEEE Pacific Rim Conference on Communications Computers and Signal Processing*, volume 1, 125–128.

Kumar, K.; Kumar, R.; de Boissiere, T.; Gestin, L.; Teoh, W. Z.; Sotelo, J.; de Brébisson, A.; Bengio, Y.; and Courville, A. C. 2019. Melgan: Generative adversarial networks for conditional waveform synthesis. In *Advances in Neural Information Processing Systems*, 14881–14892.

Łańcucki, A. 2020. FastPitch: Parallel Text-to-speech with Pitch Prediction. *arXiv preprint arXiv:2006.06873* .

Lee, S.-W.; and Kim, S.-Y. 1999. Integrated segmentation and recognition of handwritten numerals with cascade neural network. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 29(2): 285–290.

Li, N.; Liu, S.; Liu, Y.; Zhao, S.; and Liu, M. 2019. Neural speech synthesis with transformer network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 6706–6713.

Li, N.; Liu, Y.; Wu, Y.; Liu, S.; Zhao, S.; and Liu, M. 2020. RobuTrans: A Robust Transformer-Based Text-to-Speech Model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 8228–8235.

Mao, X.; Li, Q.; Xie, H.; Lau, R. Y.; Wang, Z.; and Paul Smolley, S. 2017. Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, 2794–2802.

Oord, A.; Li, Y.; Babuschkin, I.; Simonyan, K.; Vinyals, O.; Kavukcuoglu, K.; Driessche, G.; Lockhart, E.; Cobo, L.; Stimberg, F.; et al. 2018. Parallel wavenet: Fast high-fidelity speech synthesis. In *International conference on machine learning*, 3918–3926. PMLR.

Oord, A. v. d.; Dieleman, S.; Zen, H.; Simonyan, K.; Vinyals, O.; Graves, A.; Kalchbrenner, N.; Senior, A.; and Kavukcuoglu, K. 2016. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499* .

Prenger, R.; Valle, R.; and Catanzaro, B. 2019. Waveglow: A flow-based generative network for speech synthesis. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3617–3621. IEEE.

Ren, Y.; Hu, C.; Qin, T.; Zhao, S.; Zhao, Z.; and Liu, T.-Y. 2020. FastSpeech 2: Fast and High-Quality End-to-End Text-to-Speech. *arXiv preprint arXiv:2006.04558* .

Ren, Y.; Ruan, Y.; Tan, X.; Qin, T.; Zhao, S.; Zhao, Z.; and Liu, T.-Y. 2019. Fastspeech: Fast, robust and controllable text to speech. In *Advances in Neural Information Processing Systems*, 3171–3180.

Shen, J.; Pang, R.; Weiss, R. J.; Schuster, M.; Jaitly, N.; Yang, Z.; Chen, Z.; Zhang, Y.; Wang, Y.; Skerrv-Ryan, R.; et al. 2018. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4779–4783. IEEE.

Skerry-Ryan, R.; Battenberg, E.; Xiao, Y.; Wang, Y.; Stanton, D.; Shor, J.; Weiss, R.; Clark, R.; and Saurous, R. A. 2018. Towards End-to-End Prosody Transfer for Expressive Speech Synthesis with Tacotron. In *International Conference on Machine Learning*, 4700–4709.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, 5998–6008.

Verma, V.; Lamb, A.; Beckham, C.; Najafi, A.; Mitliagkas, I.; Lopez-Paz, D.; and Bengio, Y. 2019. Manifold mixup:

Better representations by interpolating hidden states. In *International Conference on Machine Learning*, 6438–6447. PMLR.

Wan, L.; Wang, Q.; Papir, A.; and Moreno, I. L. 2018. Generalized end-to-end loss for speaker verification. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4879–4883. IEEE.

Wang, Y.; Skerry-Ryan, R.; Stanton, D.; Wu, Y.; Weiss, R. J.; Jaitly, N.; Yang, Z.; Xiao, Y.; Chen, Z.; Bengio, S.; et al. 2017. Tacotron: Towards end-to-end speech synthesis. *arXiv preprint arXiv:1703.10135* .

Wang, Y.; Stanton, D.; Zhang, Y.; Skerry-Ryan, R.; Battenberg, E.; Shor, J.; Xiao, Y.; Ren, F.; Jia, Y.; and Saurous, R. A. 2018. Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis. *arXiv preprint arXiv:1803.09017* .

Yamamoto, R.; Song, E.; and Kim, J.-M. 2020. Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6199–6203. IEEE.

Yang, H.-D.; and Lee, S.-W. 2007. Reconstruction of 3D human body pose from stereo image sequences based on top-down learning. *Pattern Recognition* 40(11): 3120–3131.

Yoon, H.-W.; Lee, S.-H.; Noh, H.-R.; and Lee, S.-W. 2020. Audio Dequantization for High Fidelity Audio Generation in Flow-based Neural Vocoder. *arXiv preprint arXiv:2008.06867* .

Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2017. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412* .

Zhang, Y.-J.; Pan, S.; He, L.; and Ling, Z.-H. 2019. Learning latent representations for style control and transfer in end-to-end speech synthesis. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6945–6949. IEEE.