FREGRAD: LIGHT AND FAST FREQUENCY-AWARE DIFFUSION VOCODER

Tan Dat Nguyen, Ji-Hoon Kim, Youngjoon Jang, Jaehun Kim, Joon Son Chung

Korea Advanced Institute of Science and Technology
Daejeon, Republic of Korea

ABSTRACT

The goal of this paper is to generate realistic audio with a lightweight diffusion-based vocoder designed to tackle the issue of slow inference time. Despite the remarkable progress in recent research toward audio quality, slow inference time persists as a consequence of the iterative denoising process. To address this issue, we propose FreGrad, a lightweight diffusion-based vocoder that leverages a frequency-aware structure to denoise in wavelet domain. Moreover, we find that the noise schedule transformation that helps satisfy the diffusion theory can further improve the audio quality. Thereby, our model extremely reduces size and inference time compared to prior works but still outperforms them in terms of quality. FreGrad achieves almost 4x times faster training time and 2.2x faster Real Time Factor (RTF) compared to baseline while reducing the model size by 1.5 (only 1.78M params) without sacrificing output quality. Through comprehensive experiments, we demonstrate that our framework is either superior or competitive in generating natural audio. We provide demo at:

Index Terms— Speech Synthesis, Diffusion, Vocoder

1. INTRODUCTION

Neural vocoder is a class of neural networks that generate waveforms from acoustic features and becomes a essential building block of numerous speech-related tasks, including speech enhancement [1, 2], voice conversion [3, 4], and text-to-speech [5, 6]. The pioneer neural vocoders [7, 8], established upon Autoregressive (AR) architecture, demonstrate the ability to produce highly authentic speech. However, the fundamental architecture of these vocoders requires a substantial number of sequential operations due to its inherent structure. To expedite the inference process, various non-autoregressive models have been proposed, including flow-based models [9, 10] and Generative Adversarial Networks (GANs) [11, 12, 13]. Nonetheless, these models still present a considerable memory usage.

Diffusion models employed in speech synthesis have demonstrated encouraging results in terms of synthesis quality [14, 15, 16, 17, 18]. These models consist of two stages: 1) the *forward* phase, converting data distribution into prior noise distribution, such as Gaussian noise; 2) the *reverse* phase, gradually recovering data from random noise using learned score function. However, they are hampered by the intrinsic sequential nature of their sampling steps, leading to a slowdown in the inference process.

In this paper, we propose FreGrad which solves the slow inference by constructing a lighter and faster model while maintaining synthesis quality by leveraging frequency awareness. The key to our idea is that we enable parallel processing by bisecting the waveform using discrete wavelet transform (DWT), a lossless downsampling method. DWT breaks down a signal into different frequency parts at different scales, helping us understand its changing frequencies over time.

From this, our approach involves decomposing each waveform into two sub-band sequences. These sequences are characterized by having half the length of the input waveform and distinctly contain low and high-frequency information. To utilize this feature, FreGrad parallelly process *forward* and *reverse* phase with these sequences instead of origin waveform. Following the *reverse* phase, sub-band sequences are transformed into a waveform by inverse DWT. Furthermore, inspired by [16, 17, 19], we increase the synthesized waveform quality by proposing new frequency-dependent prior distributions. To summarize, by harnessing DWT, our denoising task process on reduced length input and output compared to traditional waveform denoising. Additionally, the sub-band sequences encompass simplified frequency information, enabling FreGrad to achieve a lighter model while preserving output quality.

Moreover, we empower the parameter efficiency of FreGrad by model architecture and training setup. We propose a building block named Frequency-aware Dilated Convolution (Freq-DConv) which replaces the dilated convolution operator inside every residual block of baseline. We draw inspiration from [20] where they apply DWT at the feature level, however, our concept is distinct in nature. In Freq-DConv, we integrate the DWT and inverse DWT, respectively, before and after the dilated convolution. Thereby, we enable frequency awareness at the latent space operations. To train FreGrad, we propose a new objective which is a weighted sum of conventional diffusion loss and a multi-resolution STFT loss. Which gives frequency awareness to FreGrad from the input-output level at the learning phase. Finally, we found

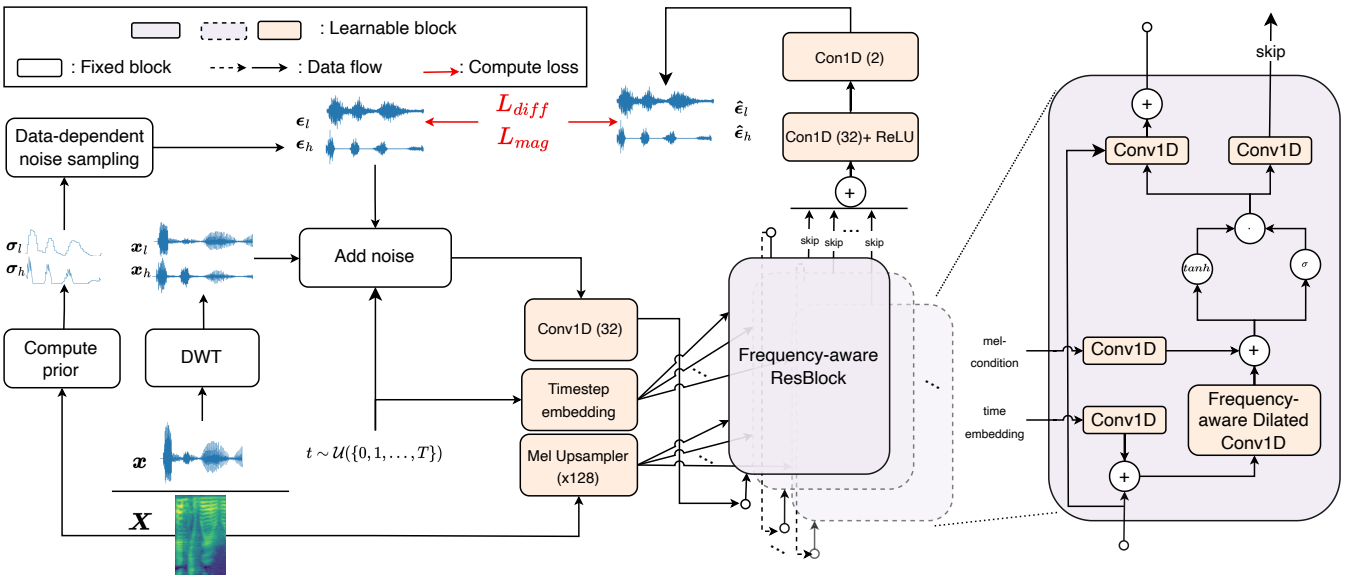


Fig. 1. Training procedure and model architecture of FreGrad. First of all (*bottom left*), we compute wavelet components and prior distributions from waveform $\boldsymbol{x}$ and mel-spectrogram $\boldsymbol{X}$. Noise sampled from prior distribution is added to wavelet features to produce noisy sample at timestep t. Given mel-spectrogram and timestep embedding, FreGrad approximates the noise that is added to input. Objective of training phase is weighted sum of L_{diff} and L_{mag} between ground-truth and predicted noise.

that noise schedules in previous works are sub-optimal. Given the important role of noise scheduling in diffusion [21, 22, 23], we apply a transformation to elevate its effectiveness. Considering all the aforementioned contributions, FreGrad demonstrates a notably accelerated inference Real-Time Factor (RTF), achieving an improvement of 2.2× times, and furthermore, it achieves a nearly fourfold reduction in training time, all while maintaining a substantially diminished model size compared to the baseline, by a factor of 1.5. Furthermore, the proposed model has relatively unchanged quality compared to baselines in various subjective and objective metrics.

2. BACKGROUNDS

Forward process of a diffusion model involves iteratively perturbing a signal by adding a controlled amount of noise at each timestep, progressively moving it towards the terminal distribution, e.g. isotropic Gaussian. Noisy signal $\boldsymbol{x}_t$ achieved from $\boldsymbol{x}_{t-1}$ is defined by transition probability:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}(\boldsymbol{x}_t; \sqrt{1-\beta_t}\boldsymbol{x}_{t-1}, \beta_t\mathbf{I}) \tag{1}$$

with predefined noise schedule $\beta_t \in \{\beta_0, \ldots, \beta_T\}$ and $\boldsymbol{x}_0$ is groundtruth sample. Futhermore, $\boldsymbol{x}_t$ can be directly achieved from $\boldsymbol{x}_0$ using reparameterization trick:

$$\boldsymbol{x}_t = \sqrt{\gamma_t}\boldsymbol{x}_0 + \sqrt{1-\gamma_t}\boldsymbol{\epsilon} \tag{2}$$

where $\alpha_t = 1 - \beta_t$, $\gamma_t = \prod_{i=1}^{t} \alpha_t$, and $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. With a sufficiently large T, the distribution of $\boldsymbol{x}_T$ is an Isotropic Gaussian distribution. So, we can generate a sample in distribution $q(\boldsymbol{x}_0)$ by tracing the exact *reverse* process $p(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ from an initial point $\boldsymbol{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

Since $p(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ depends on the entire data distribution, we approximate it by neural networks $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ which approximate the distribution $\mathcal{N}(\boldsymbol{x}_{t-1}; \mu_\theta(\boldsymbol{x}_t, t), \sigma_\theta^2(\boldsymbol{x}_t, t))$. The variance σ_θ^2 is often predefined as $\frac{1-\gamma_{t-1}}{1-\gamma_t}\beta_t$ or β_t. One widely used definition of mean is [24]:

$$\mu_\theta(\boldsymbol{x}_t, t) = \frac{1}{\alpha_t}(\boldsymbol{x}_t - \frac{\beta_t}{\sqrt{1-\gamma_t}}\epsilon_\theta(\boldsymbol{x}_t, t)) \tag{3}$$

with ϵ_θ is a neural network that learns to predict the noise given x_t and t. In practice, the objective of training ϵ_θ is often simplified to minimize $\mathbf{E}_{t,\boldsymbol{x}_t,\epsilon}\left[\|\boldsymbol{\epsilon} - \epsilon_\theta(\boldsymbol{x}_t, t)\|_2^2\right]$.

PriorGrad [16] starts sampling procedure from an initial point $x_T \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{data})$. $\boldsymbol{\Sigma}_{data}$ is the diagonal matrix $diag\left[(\sigma_0^2, \sigma_1^2, \ldots, \sigma_L^2)\right]$ with σ_i^2 is normalized frame-level energy of mel-spectrogram length L at i-th frame. The training objective is modified to:

$$L_{diff} = \mathbf{E}_{t,\boldsymbol{x}_t,\boldsymbol{\epsilon},\boldsymbol{c}}\left[\|\boldsymbol{\epsilon} - \epsilon_\theta(\boldsymbol{x}_t, t, \boldsymbol{c})\|^2_{\boldsymbol{\Sigma}^{-1}_{data}}\right] \tag{4}$$

where $\|\mathbf{x}\|^2_{\boldsymbol{\Sigma}} = \mathbf{x}^\top\boldsymbol{\Sigma}^{-1}\mathbf{x}$ and $\boldsymbol{c}$ is melspectrogram.

3. FREGRAD

In FreGrad, we employ DWT function $\Phi(\cdot)$ with Haar mother wavelet to decompose groundtruth waveform $\boldsymbol{x} \in \mathbb{R}^D$ into

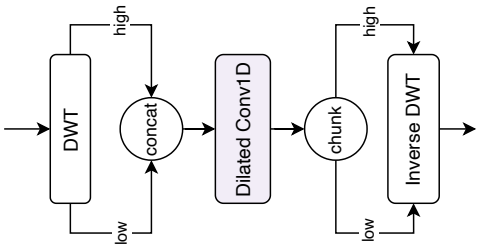


Fig. 2. Frequency-aware Dilated Convolution

the wavelet features:

$$\left[\boldsymbol{x}^l, \boldsymbol{x}^h\right]_0 = \Phi(\boldsymbol{x}_0) \tag{5}$$

with $\{\boldsymbol{x}^l, \boldsymbol{x}^h\} \subset \mathbb{R}^{D/2}$ stand for low and high-frequency subband achieved from DWT, respectively. We obtain the noisy wavelet features $\boldsymbol{x}_t^l$ and $\boldsymbol{x}_t^h$ at timestep t by Eqn. (2) with noise $\boldsymbol{\epsilon}_l$ and $\boldsymbol{\epsilon}_h$ sample from our designed prior distribution. The added noise are are concurrently estimated by neural network:

$$\hat{\boldsymbol{\epsilon}}^l, \hat{\boldsymbol{\epsilon}}^h = \epsilon_\theta(\boldsymbol{x}_t^l, \boldsymbol{x}_t^h, t, \boldsymbol{X}) \tag{6}$$

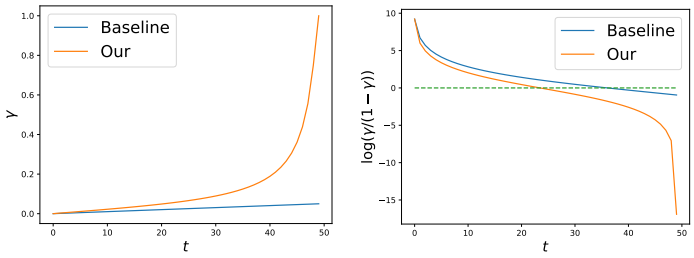
where $\boldsymbol{X} \in \mathbb{R}^{K \times L}$ stands for mel-spectrogram that has K frequency bins and L frames. In inference, clean wavelet features after denoised are converted into speech by inverse DWT: $\hat{\boldsymbol{x}}_0 = \Phi^{-1}([\hat{\boldsymbol{x}}_l, \hat{\boldsymbol{x}}_h]_0)$ where $\{\hat{\boldsymbol{x}}_l, \hat{\boldsymbol{x}}_h\}$ are denoised signal achieved from sampling.

3.1. Frequency-aware learning

Since each wavelet feature contains specific frequency-band information, we separately compute prior distribution $\{\boldsymbol{\sigma}_l, \boldsymbol{\sigma}_h\}$ from $\{\boldsymbol{X}_{0:\lfloor \frac{K}{2} \rfloor}, \boldsymbol{X}_{\lfloor \frac{K}{2} \rfloor:K}\}$ as in [16], respectively. We also apply low-pass and high-pass filters in order to control the cut-off frequency alias in computing $\boldsymbol{\sigma}$ and post-processing. Besides, we introduce Frequency-aware Dilated Convolution (Freq-DConv) with structure as depicted in Fig. 2. This block replaces the dilated convolution inside each block of origin ResBlock [14] to provide attention to frequency while training the model. Each signal will be decomposed into low and high-frequency subbands before applying dilated convolution. Replacement causes a small change in network forward speed but significantly improve learning performance.

Different from [12, 25], FreGrad only use Multi-resolution loss STFT magnitude loss $L_{mag} = \frac{1}{M}\Sigma_{m=1}^{M} L_{mag}^{(m)}$, where L_{mag} is single STFT loss and M is the total number of resolution. Besides, the diffusion loss Eqn. (4) separately computes the loss between noises that are added to low and high-frequency subbands. Our objective of the training process is:

$$L = \Sigma_{i \in \{l,h\}} \left[L_{diff}(\boldsymbol{\epsilon}_i, \hat{\boldsymbol{\epsilon}}_i) + \lambda L_{mag}(\boldsymbol{\epsilon_i}, \hat{\boldsymbol{\epsilon}}_i)\right] \tag{7}$$



(a) Noise level γ (b) log SNR

Fig. 3. Noise level and log SNR though time steps. We shift the baseline noise level as shown in (a) to get the satisfied SNR as shown in (b)

3.2. Noise schedule transformation

The diffusion-based model starts sampling from pure noise. This means, as discussed in [21, 22], signal-to-noise ratio (SNR) should be zero at the final step of *forward* process. Widely used noise schedules in previous works [16, 14, 17, 18] fails to reach SNR near 0 at the final step. We adopt [23] to simply transform an arbitrary noise schedule in order to satisfy mentioned requirement.

$$\sqrt{\bar{\gamma}}_{new} = \frac{\sqrt{\bar{\gamma}}_0}{\sqrt{\bar{\gamma}}_0 - \sqrt{\bar{\gamma}}_T + \tau}(\sqrt{\bar{\gamma}} - \sqrt{\bar{\gamma}}_T + \tau) \tag{8}$$

τ helps avoid dividing by zero in sampling. By this simple transformation, new noise schedule can highly satisfy SNR requirement as depicted in Fig. 3.

4. EXPERIMENTS

4.1. Training setup

We conduct experiments on single English speaker LJSpeech-1.1[1] which contains 13100 samples. We use 13000 samples for training and 100 remaining samples for testing. Mel-spectrograms used for all model are computed from groundtruth audio with 80-band mel features with 1024 FFT points, from 80Hz to 8000Hz, and hop length of 256 (but, use 300 for WaveGrad to keep baseline structure).

Implementation detail: Our model backbone is bases on [14] which replaced Dilated Convolution with Frequency-aware Dilated Convolution. The network consists of 30 Frequency-aware Resblock (FreqResBlock) with dilation cycle length of 7 and a hidden dimension of 32. We reuse both timestep embedding and Mel-Upsampler setting from baseline but reduce half of the upsampling rate of Mel-Upsampler. We use the open-source Pytorch Wavelets [2] and apply 1D transformation using Haar wavelet and other default parameters. For multi-resolution STFT loss, we choose $M = 3$,

[1] https://keithito.com/LJ-Speech-Dataset/
[2] https://pytorch-wavelets.readthedocs.io/

Table 1. *Objective and subjective test results, with their 95% confidence intervals, computed on same set of 100 samples. These samples are synthesized using the same noise schedule with training phase. We conduct MOS test on 50 samples from this set. For all metrics, lower is better. RTF is computed on AMD EPYC 7452 32-Core Processor with single GeForce RTX 3080.*

Model	LS-MAE (↓)	MR-STFT (↓)	MCD (↓)	RMSE_f_0 (↓)	Param (↓)	RTF (↓)	MOS (↓)
GT	–	–	–	–	–	–	0.47 ± 0.02
WaveGrad	0.59 ± 0.01	1.39 ± 0.01	3.06 ± 0.05	39.97 ± 2.10	15.81M	**0.29**	
DiffWave	0.56 ± 0.01	1.18 ± 0.01	3.20 ± 0.07	40.10 ± 1.97	2.62M	0.64	
PriorGrad	0.47 ± 0.02	1.14 ± 0.03	2.22 ± 0.04	40.42 ± 2.21	2.62M	0.65	
Our	$\mathbf{0.45 \pm 0.02}$	$\mathbf{1.12 \pm 0.01}$	$\mathbf{2.19 \pm 0.03}$	$\mathbf{38.73 \pm 2.16}$	**1.78M**	**0.29**	

Table 2. *Subjective metrics for inference with 6 iterations. We inherit the inference steps from PriorGrad for all models without optimization.*

	RMSE_f_0	MCD	RTF
WaveGrad	38.59 ± 2.10	3.20 ± 0.05	0.04
DiffWave	39.87 ± 2.13	3.17 ± 0.07	0.07
PriorGrad	39.21 ± 2.13	2.37 ± 0.04	0.09
Our	39.64 ± 2.13	2.48 ± 0.02	**0.04**

Table 3. *CMOS results of ablation study for FreGrad*

	CMOS	RTF	MCD
Our	0	0.29	$\mathbf{2.19 \pm 0.03}$
w/o Freq-DConv	-0.140	0.18	2.58 ± 0.04
w/o separate prior		0.29	2.32 ± 0.03
w/o zero SNR		0.29	2.25 ± 0.03

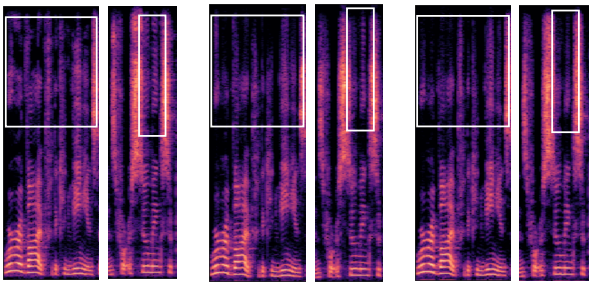
with FFT size of $[512, 1024, 2048]$ and the Hanning window lengths of $[240, 600, 1200]$. Finally, we choose $\lambda = 1e^{-1}$ and $\tau = 1e^{-4}$ for our experiments.

Metrics: We consider PriorGrad [3] [16], DiffWave [4] [14], and WaveGrad [5] [18] as our baseline. All of them are trained following publicly available implementation with the same training data and batch size of 16. For subjective evaluation, we rate naturalness of samples using a five-point scale and report the mean opinion score (MOS). In CMOS test, we rate the quality of generated samples from the ablation model with our model from -3 to 3 in term of quality. Raters evaluate the samples using headphones. For objective evaluation, we use 5 metrics: (i) real time factor (RTF), (ii) log-mel spectrogram mean absolute error (LS-MAE), (iii) mel-cepstral distortion (MCD), (iv) root mean square error of f_0 estimated by [26] (RMSE_f_0), (v) number of model parameters (Params), (vi) Multi-Resolution Short-time Fourier Transform loss [12] (MR-STFT). Both MCD and RMSE_f_0 use dynamic time warping.

4.2. Audio quality and sampling speed

Objective and subjective results are shown in Table 1. FreGrad consistently outperforms the baseline in objective metrics all shown objective metrics. Based on our qualitative observation, this improvement comes from the capacity to reconstruct high-frequency components while keeping the low-frequency information. As depicted in Fig. 4, baseline tends to over-smooth high-frequency information because the learning process has no explicit knowledge of frequency, but FreGrad can highly preserve these features. Besides, with a lighter model size, which reduces 20% parameters of baseline, our model can still generate low-frequency information as comparative as baseline. Moreover, our model decreases 45% of RTF by using wavelet features which process in signal with 2 times shorter than the signal used in the baseline. Finally, the proposed FreGrad achieves .... MOS. Although the RTF of FreGrad and WaveGrad are similar, the MOS score is highly improved by FreGrad. Furthermore, the benefit of the proposed model could be found in its fast training speed. It only requires 1.5 days to complete 1M training iterations, while baseline requires more than 6 days. Besides, we conduct three ablation studies to verify the effectiveness of our proposed ideas. The results Table 3 show that

The more refinements, the smoother the f0 is table 2. However, the change is small and has no evidence to conclude



(a) GT (b) Our (c) PriorGrad

Fig. 4. The baseline (c) exhibits a tendency to over-smooth high-frequency information resulting in the evitable removal of ground truth high-frequency information. In contrast, FreGrad accurately reconstructs this information (b).

5. CONCLUSION

We proposed a lighter accelerated diffusion-based vocoder FreGrad. The proposed model concentrates on giving frequency awareness to the denoising process through using wavelet features and Frequency-aware Dilated Convolution.

[3] https://github.com/microsoft/NeuralSpeech/
[4] https://github.com/lmnt-com/diffwave
[5] https://github.com/lmnt-com/wavegrad

Besides, we also employ multi-resolution STFT loss and transform noise schedule to robust learning capacity of a smaller size model. Our experimental results show that the proposed model consistently outperforms baseline on both subjective and objective metrics with extremely reduced RTF. In the future, several aspects are worthy of exploring. First, applying multiple DWT functions to create simpler target distribution. Hence, model size and inference speed can be accelerated. Second, enhancing forward speed of Frequency-aware Dilated Convolution. Finally, optimizing the DWT calculating speed can effectively reduce the sampling and training speed of the model.

6. REFERENCES

[1] S. Maiti and M. I. Mandel, "Parametric resynthesis with neural vocoders," in WASPAA. 2019, pp. 303–307, IEEE.

[2] J. Su, Z. Jin, and A. Finkelstein, "Hifi-gan: High-fidelity denoising and dereverberation based on speech deep features in adversarial networks," in INTERSPEECH. 2020, pp. 4506–4510, ISCA.

[3] K. Qian, Y. Zhang, S. Chang, X. Yang, and M. Hasegawa-Johnson, "Autovc: Zero-shot voice style transfer with only autoencoder loss," in ICML. 2019, vol. 97 of Proceedings of Machine Learning Research, pp. 5210–5219, PMLR.

[4] Y. A. Li, A. Zare, and N. Mesgarani, "Starganv2-vc: A diverse, unsupervised, non-parallel framework for natural-sounding voice conversion," in Interspeech. 2021, pp. 1349–1353, ISCA.

[5] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Ryan, R. A. Saurous, Y. Agiomyrgiannakis, and Y. Wu, "Natural TTS synthesis by conditioning wavenet on MEL spectrogram predictions," in ICASSP. 2018, pp. 4779–4783, IEEE.

[6] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T. Liu, "Fastspeech: Fast, robust and controllable text to speech," in NeurIPS, 2019, pp. 3165–3174.

[7] S. Mehri, K. Kumar, I. Gulrajani, R. Kumar, S. Jain, J. Sotelo, A. C. Courville, and Y. Bengio, "Samplernn: An unconditional end-to-end neural audio generation model," in ICLR (Poster). 2017, OpenReview.net.

[8] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," in SSW. 2016, p. 125, ISCA.

[9] R. Prenger, R. Valle, and B. Catanzaro, "Waveglow: A flow-based generative network for speech synthesis," in ICASSP. 2019, pp. 3617–3621, IEEE.

[10] W. Ping, K. Peng, K. Zhao, and Z. Song, "Waveflow: A compact flow-based model for raw audio," in ICML. 2020, vol. 119 of Proceedings of Machine Learning Research, pp. 7706–7716, PMLR.

[11] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," in NeurIPS, 2020.

[12] R. Yamamoto, E. Song, and J. Kim, "Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in ICASSP. 2020, pp. 6199–6203, IEEE.

[13] J. Kim, S. Lee, J. Lee, and S. Lee, "Fre-gan: Adversarial frequency-consistent audio synthesis," in Interspeech. 2021, pp. 2197–2201, ISCA.

[14] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, "Diffwave: A versatile diffusion model for audio synthesis," in ICLR, 2021.

[15] R. Huang, M. W. Y. Lam, J. Wang, D. Su, D. Yu, Y. Ren, and Z. Zhao, "Fastdiff: A fast conditional diffusion model for high-quality speech synthesis," in IJCAI, 2022, pp. 4157–4163.

[16] S. Lee, H. Kim, C. Shin, X. Tan, C. Liu, Q. Meng, T. Qin, W. Chen, S. Yoon, and T. Liu, "Priorgrad: Improving conditional denoising diffusion models with data-dependent adaptive prior," in ICLR, 2022.

[17] Y. Koizumi, H. Zen, K. Yatabe, N. Chen, and M. Bacchiani, "Specgrad: Diffusion probabilistic model based neural vocoder with adaptive noise spectral shaping," in INTERSPEECH. 2022, pp. 803–807, ISCA.

[18] N. Chen, Y. Zhang, H. Zen, R. J. Weiss, M. Norouzi, and W. Chan, "Wavegrad: Estimating gradients for waveform generation," in ICLR, 2021.

[19] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, and M. A. Kudinov, "Grad-tts: A diffusion probabilistic model for text-to-speech," in ICML. 2021, vol. 139 of Proceedings of Machine Learning Research, pp. 8599–8608, PMLR.

[20] H.-D. Phung, Q. Dao, and A. Tran, "Wavelet diffusion models are fast and scalable image generators," CVPR, 2022.

[21] T. Chen, "On the importance of noise scheduling for diffusion models," CoRR, vol. abs/2301.10972, 2023.

[22] E. Hoogeboom, J. Heek, and T. Salimans, "simple diffusion: End-to-end diffusion for high resolution images," CoRR, vol. ICML, 2023.

[23] S. Lin, B. Liu, J. Li, and X. Yang, "Common diffusion noise schedules and sample steps are flawed," CoRR, vol. abs/2305.08891, 2023.

[24] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in NeurIPS, 2020.

[25] Z. Chen, X. Tan, K. Wang, S. Pan, D. P. Mandic, L. He, and S. Zhao, "Infergrad: Improving diffusion models for vocoder by considering inference in training," in ICASSP. 2022, pp. 8432–8436, IEEE.

[26] M. Morise, "Harvest: A high-performance fundamental frequency estimator from speech signals," in INTERSPEECH. 2017, pp. 2321–2325, ISCA.