

Guitar Tone Morphing by Diffusion-based Model

Kuan-Yu Chen^{1*}, Kuan-Lin Chen^{1*}, Yu-Chieh Yu^{1*}, Jian-Jiun Ding¹

¹Graduate Institute of Communication Engineering, National Taiwan University, Taiwan

*These three authors contributed equally.

E-mail: {r13942135, r13942067, r12942142, jjding}@ntu.edu.tw

Abstract—In Music Information Retrieval (MIR), modeling and transforming the tone of musical instruments, particularly electric guitars, has gained increasing attention due to the richness of the instrument tone and the flexibility of expression. Tone morphing enables smooth transitions between different guitar sounds, giving musicians greater freedom to explore new textures and personalize their performances. This study explores learning-based approaches for guitar tone morphing, beginning with LoRA fine-tuning to improve the model performance on limited data. Moreover, we introduce a simpler method, named spherical interpolation using Music2Latent. It yields significantly better results than the more complex fine-tuning approach. Experiments show that the proposed architecture generates smoother and more natural tone transitions, making it a practical and efficient tool for music production and real-time audio effects.

I. INTRODUCTION

Guitar tone morphing is to perform transformation from one guitar sound to another smoothly, enabling more expressive performance and creative exploration. While traditional tone shaping relies on discrete effects or preset changes, morphing offers continuous control over timbre, bridging tonal spaces that were previously separated.

Early approaches to tone morphing were based on signal processing. Dudley’s Vocoder [1] decomposed audio into frequency bands for real-time transformation. The phase vocoder [2] improved on this by using short-time Fourier transforms to enable time-stretching and pitch-shifting. Additive resynthesis [3] constructed tone flexibly by controlling sine wave parameters.

With the rise of deep learning, guitar tone modeling has become increasingly data-driven. NSynth [4] used a WaveNet autoencoder to interpolate between instruments in latent space. GANSynth [5] improved audio fidelity by jointly modeling magnitude and phase.

Recently, diffusion models have shown strong performance in audio generation. DiffWave [6] produced high-quality waveforms through iterative denoising, while AudioLDM [7] and MusicLDM [8] extended these techniques to music datasets and text-to-audio tasks. Style interpolation has been explored through LoRA fine-tuning in SoundMorpher [9] and QKV manipulation for controllable morphing [10]. In parallel, Music2Latent [11] offered a lightweight, content-preserving approach to audio interpolation without diffusion.

Visual-domain methods such as IMPUS [12] and DiffMorpher [13] have also influenced our approach. These models achieve perceptual uniformity by interpolating both latent vectors and network weights.

In this work, we explore guitar tone morphing through four methods: three are based on latent diffusion models with various LoRA configurations, and one is to apply direct latent interpolation in Music2Latent. By combining ideas from the neural style transfer, latent blending, and diffusion generation, smooth and perceptually natural transitions between guitar tones with shared musical content can be achieved.

II. RELATED WORK

Research on direct audio tone morphing remains limited, particularly for guitar tones. Existing studies mostly explored switching between distinct effects rather than smooth transitions. MorphFader [10] used a TTAs model to blend two sounds by adjusting the attention mechanisms and was able to control extreme tone changes. SoundMorpher [9] applied a diffusion-based approach to generate perceptually smooth audio transitions, emphasizing uniform and natural-sounding morphs.

While audio tone morphing is still emerging, computer vision (CV) has advanced significantly in morphing and interpolation, providing valuable insights. For example, IMPUS [12] used diffusion to generate perceptually uniform image transitions. DiffMorpher [13] further improved this by interpolating both LoRA weights and noise input, allowing label-free and fine-grained visual morphing. The Music2Latent framework [11] provided useful guidance on style-content disentanglement in audio. These techniques inspired us to develop an advanced guitar tone interpolation approach.

In this work, guitar tone morphing is formulated as interpolation between mel spectrograms. Using latent diffusion models (LDM) [14], we adapt image interpolation methods to smoothly transition between audio tones, even when the inputs are significantly different.

We also apply the technique of audio style transfer, which transforms the sound style (e.g., through different amps) while preserving content. Since our goal is to interpolate rather than transfer, maintaining content stability while blending tonal characteristics is an important issue.

In summary, our approach adapts diffusion-based morphing, originally from image processing, to guitar tone interpolation. This enables smoother and more natural transitions between tones. We also integrate audio style transfer to enhance flexibility and expressiveness, making the system both natural to the ear and more controllable for users shaping timbre to their artistic goals.

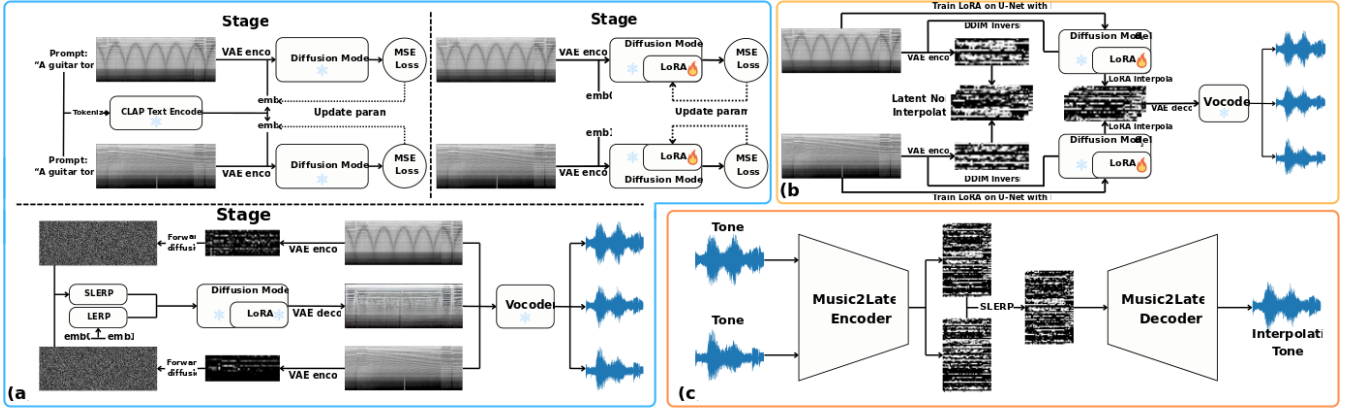


Fig. 1. Overall architecture of the proposed methods: (a) Single-Sided LoRA fine-tuning applied to the decoder; (b) Dual-Sided LoRA applied to both encoder and decoder; (c) Music2Latent interpolation that directly performs spherical interpolation in the latent space without parameter updating.

III. METHODOLOGY

In this study, we propose four different methods for guitar tone interpolation. The first three methods are based on the Latent Diffusion Model (LDM)[7], [8], [15] and differ mainly in how they utilize LoRA fine-tuning and whether they rely on textual prompts or purely latent representations. The fourth method uses the Music2Latent framework [11], which does not use diffusion models and instead directly interpolates within a learned latent space. We detail each of the four proposed approaches individually.

A. Interpolation and Style Transfer Techniques

Several important interpolation and style transfer techniques are utilized in the proposed approaches, including Spherical Linear Interpolation (SLERP), Linear Interpolation (LERP), and Adaptive Instance Normalization (AdaIN).

a) Spherical Linear Interpolation (SLERP): SLERP provides smooth interpolation between two vectors on the unit sphere, ensuring geometrically consistent transitions. Given two vectors \mathbf{v}_0 and \mathbf{v}_1 , SLERP is performed as follows.

First, both \mathbf{v}_0 and \mathbf{v}_1 are normalized:

$$\hat{\mathbf{v}} = \frac{\mathbf{v}}{\|\mathbf{v}\|}. \quad (1)$$

Next, we compute the angle θ_0 between them:

$$\theta_0 = \arccos(\hat{\mathbf{v}}_0 \cdot \hat{\mathbf{v}}_1). \quad (2)$$

When the vectors are nearly identical (i.e., θ_0 is very small), linear interpolation (LERP) is used for interpolation instead to avoid numerical problems.

$$\text{LERP}(\alpha, \hat{\mathbf{v}}_0, \hat{\mathbf{v}}_1) = (1 - \alpha)\hat{\mathbf{v}}_0 + \alpha\hat{\mathbf{v}}_1. \quad (3)$$

Otherwise, SLERP is computed by:

$$\text{SLERP}(\alpha, \hat{\mathbf{v}}_0, \hat{\mathbf{v}}_1) = \frac{\sin((1 - \alpha)\theta_0)}{\sin(\theta_0)}\hat{\mathbf{v}}_0 + \frac{\sin(\alpha\theta_0)}{\sin(\theta_0)}\hat{\mathbf{v}}_1. \quad (4)$$

b) Adaptive Instance Normalization (AdaIN): AdaIN [16] is applicable to style transfer tasks. It adjusts the mean and variance of one feature map (x) to match another (y), thus aligning the style distributions:

$$\text{AdaIN}(x, y) = \sigma(y) \left(\frac{x - \mu(x)}{\sigma(x)} \right) + \mu(y). \quad (5)$$

B. Without LoRA Fine-tuning

This baseline method does not incorporate any LoRA fine-tuning. We begin by applying **Textual Inversion** to enhance the expressiveness of prompts in the latent space. Then, we encode two input audio clips using the pre-trained LDM encoder and directly interpolate their latent vectors using SLERP. The interpolated vector is decoded back into the Mel-spectrogram by the original decoder and is finally transformed into a waveform using a vocoder. This approach allows us to evaluate the tone interpolation capabilities of a purely pre-trained diffusion model without any task-specific adaptation.

C. Single-Sided LoRA Fine-tuning

Building on the AudioLDM framework [7], this method introduces LoRA fine-tuning to enhance prompt control and decoding quality. After performing textual inversion, we fine-tune only the conditional U-Net using LoRA to align text embeddings with latent features. Meanwhile, the unconditional U-Net is fine-tuned independently with a LoRA rank of 2, without textual input. During inference, we apply SLERP to interpolate latent vectors from the two input tones and perform linear interpolation (LERP) on their text embeddings. The interpolated representation is then decoded through the unconditional U-Net, the VAE, and the vocoder. This method enables more expressive tone blending by partially adapting the model to the task.

As illustrated in Fig. 1(a), the LoRA fine-tuning is applied only to the decoder side (U-Net), while the encoder remains frozen.

D. Dual-Sided LoRA Fine-Tuning

This method extends the single-sided approach by fine-tuning two separate U-Net models. Different models are applied for different input tones. Following textual inversion, each U-Net is individually fine-tuned using LoRA with rank 2 to specialize in its respective tone domain. During inference, we perform SLERP on the latent vectors and LERP on the prompt embeddings, as in the previous method. Additionally, we interpolate the parameters of the two U-Nets via LERP to construct a merged model tailored to the interpolated tone. The final latent data are decoded using the merged U-Net, followed by the VAE and the vocoder.

To further refine tone characteristics, we integrate AdaIN into the decoding pipeline. AdaIN makes the mean and variance of the output match those of the interpolated tone style, allowing for more precise control over the stylistic attributes of the output tone.

As shown in Figure 1(b), both conditional and unconditional U-Nets are fine-tuned independently and their parameters are interpolated at the inference process.

E. Music2Latent Interpolation

This method employs the Music2Latent framework [11], which does not rely on diffusion models or text conditioning. Instead, it uses a lightweight encoder–decoder architecture for latent audio reconstruction. We encode both input audio signals, interpolate their latent vectors using SLERP, and decode the result back into audio. While lacking fine-tuning, this method offers a clean benchmark for evaluating the interpolation quality within a learned latent space. We also apply AdaIN to better align the style characteristics during latent morphing, enhancing the naturalness of the resulting tone.

As illustrated in Figure 1(c), the system directly encodes and decodes the audio signal without text prompts or parameter tuning, relying purely on latent-space interpolation.

IV. EXPERIMENTS AND RESULTS

A. Dataset

Our dataset consists of real guitar recordings that were processed using a variety of tone-shaping effects to simulate a wide range of guitar tones. This approach ensures diversity and realism in the tonal characteristics captured. The dataset contains a wide range of professionally recorded tones that well reflect real-world playing conditions.

We define *five core tone morphing tasks*, each corresponding to a commonly used guitar tone transition inspired by standard guitar effects [17]. Each task contains 20 pairs of audio clips, where each pair represents a source-target tone mapping for evaluation or fine-tuning. The tasks are:

- **Clean to High Gain:** transforming a clean tone into one with heavy distortion.
- **Clean to Low Gain:** adding light distortion to a clean tone.

- **Low Gain to High Gain:** increasing distortion on an already mildly distorted tone.
- **Clean to Modulation:** applying effects like chorus or flanger to a clean tone.
- **Modulation to High Gain:** Combining modulation effects with heavy distortion.

Each pair consists of two short clips (about 5 seconds each). While the original recordings were sampled at the rate of 44.1 kHz, most diffusion-based models for comparison, such as AudioLDM and SoundMorpher, require the audio to be downsampled to 16 kHz due to model constraints. As a result, we performed most experiments using the 16 kHz version of the dataset. However, the proposed model of spherical interpolation using Music2Latent supports native 44.1 kHz input, allowing it to preserve more detailed high-frequency content, which gives it a potential advantage in **timbral accuracy** and **clarity** during tone morphing.

Each experiment was conducted at a local pairwise level, where fine-tuning or interpolation is directly applied to each input-output tone pair. This setup reflects real-world tone morphing situations. As shown in Fig. 1, fine-tuning is only performed on the two input sides in these cases.

B. Evaluation Metrics

1) *Interpolation Tone Quality Assessment Metrics:* To comprehensively evaluate the quality of tone morphing, we employ both objective and subjective metrics:

- **CDPAM mean \pm std** is a perceptual audio similarity metric derived from deep models. It reflects how consistently the generated audio is perceived as similar to the target, based on features aligned with human hearing. The mean indicates average similarity, while the standard deviation captures variability across different comparisons.
- **Mean Opinion Score (MOS)** provides a subjective evaluation of audio quality. In this study, 20 human raters participated in a blind test to rate the naturalness and smoothness of the morphed outputs on a scale from 1 (poor) to 5 (excellent).

The objective metric CDPAM offers a quantitative and reproducible way to assess audio similarity, while MOS complements them by capturing perceptual nuances that automated metrics might overlook. Notably, MOS remains a widely trusted indicator in audio generation research, making it a critical component of our evaluation protocol. By combining these metrics, we gain a well-rounded understanding of how effective each model is in producing high-quality, perceptually convincing tone morphs.

2) *Spectral Convergence (SC) for Reconstruction Quality Evaluation:* To assess the reconstruction quality of the generated audio signals—particularly how faithfully the models decode internal representations back into waveform—we adopt the **Spectral Convergence (SC)** loss, a key component of the multi-resolution Short-Time Fourier Transform (STFT) loss.

This metric directly evaluates differences in the frequency domain by comparing the spectrograms of the predicted and

ground truth waveforms. Let M_r and M_t denote the magnitude spectrograms of the reconstructed and target audio, respectively. The SC loss is formulated as:

$$SC(M_r, M_t) = \frac{\sqrt{\sum_{m,k} (M_t(m, k) - M_r(m, k))^2}}{\sqrt{\sum_{m,k} M_t(m, k)^2}}. \quad (6)$$

Here, m and k represent the time and frequency indices in the spectrogram. Intuitively, the SC loss measures the normalized deviation between the predicted and target spectrograms, where a lower value indicates better spectral reconstruction.

To capture both short- and long-term frequency patterns, we compute SC values across multiple STFT configurations. Specifically, we use three window sizes: [1024, 2048, 512], corresponding hop sizes: [160, 240, 50], and window lengths: [600, 1200, 240]. These multi-resolution settings enable a robust evaluation of the generated audio’s structure across different temporal and spectral granularities.

This SC-based analysis is particularly insightful for latent diffusion models (LDMs), which typically employ a VAE *encoder-decoder pipeline* and/or *neural vocoders* to reconstruct waveform outputs. The SC score thus helps us identify how much information is lost or distorted during the decoding process, making it a critical metric for analyzing the fidelity of waveform reconstruction.

C. results

TABLE I
TONE MORPHING QUALITY EVALUATION ACROSS DIVERSE METHODS AND LDM VARIANTS WITH DIFFERENT LoRA CONFIGURATIONS. ↓ INDICATES LOWER IS BETTER; ↑ INDICATES HIGHER IS BETTER.

Model	CDPAM _{mean} ± std ↓	MOS ↑
AudioLDM		
w/o LoRA	0.32 ± 0.100	3.17
w/ 1 LoRA	0.45 ± 0.140	1.07
w/ 2 LoRA	0.22 ± 0.132	3.03
AudioLDM2		
w/o LoRA	0.25 ± 0.120	3.30
w/ 1 LoRA	0.34 ± 0.110	2.70
w/ 2 LoRA	0.33 ± 0.122	3.23
MusicLDM		
w/o LoRA	0.85 ± 0.116	1.97
w/ 1 LoRA	0.19 ± 0.120	3.70
w/ 2 LoRA	0.08 ± 0.114	1.20
Spherical Music2Latent		
Interpolation	0.13 ± 0.060	4.3

a) *Perceptual Quality Evaluation*: We evaluated the performance of various models on the guitar tone morphing task, as summarized in **Table I**. The assessment focused on two key metrics: **CDPAM mean ± std** and **Mean Opinion Score (MOS)**. Among all systems, spherical interpolation using Music2Latent achieved the highest MOS, indicating that listeners consistently rated its outputs as the most natural and musically coherent.

What stands out is the performance of spherical interpolation using Music2Latent, which achieved the best subjective rating

TABLE II
SPECTRAL CONVERGENCE (SC) LOSS ACROSS DIFFERENT VOCODER FRAMEWORKS, WITH AND WITHOUT VAE. A LOWER SC MEANS BETTER PERFORMANCE.

Model	Mean	Median
Without VAE		
BigVGAN	0.03978	0.04005
HifiGAN (AudioLDM)	0.72955	0.15795
HifiGAN (AudioLDM2)	0.72955	0.15795
HifiGAN (MusicLDM)	0.30206	0.15017
With VAE		
HifiGAN (AudioLDM)	1.80133	0.79835
HifiGAN (AudioLDM2)	0.49473	0.13271
HifiGAN (MusicLDM)	0.46880	0.11239
BigVGAN (AudioLDM)	1.10628	0.35980
BigVGAN (AudioLDM2)	0.50800	0.12355
BigVGAN (MusicLDM)	0.65704	0.22113

despite being structurally different from other LDMs. Unlike other models that rely on VAE and vocoder components, it performs interpolation directly in the latent space and uses a different decoding strategy. This may be helpful for preserving structural and stylistic continuity between source and target tones, contributing to its superior perceptual quality.

In conclusion, our results highlight that **structure-preserving latent interpolation** can play a crucial role in improving the subjective quality of tone morphing. The following section explores how VAE and vocoder components can affect reconstruction quality in LDM.

b) *Reconstruction Capabilities of Different Frameworks*:

We evaluated the reconstruction performance of different structures using the training dataset. For models without a VAE, the audio is first converted to a Mel spectrogram and then directly reconstructed using a vocoder. For models that include a VAE, the spectrogram is first encoded and decoded through the VAE before being passed to the vocoder for waveform generation.

BigVGAN[18] uses a consistent pre-trained model, `nvidia/bigvgan_v2_44khz_128band_512x`. For HiFiGAN-based models, we also apply the same set of pre-trained checkpoints to ensure fair comparison. The SC loss results across different setups are shown in **Table II**.

From the table, we can see that BigVGAN performs best in terms of SC loss when used without a VAE. However, when paired with a VAE trained within the HiFiGAN framework, reconstruction quality improves further. This suggests a strong compatibility between the VAE and vocoder in that setup. However, combining BigVGAN with VAEs from other LDM architectures results in higher SC loss, possibly due to a mismatched design.

These findings highlight that many existing models rely on several interdependent components, such as VAEs, vocoders, and auxiliary modules, to achieve good audio quality. This reliance adds complexity, risks error propagation, and destabilizes reconstruction. In contrast, our proposed model removes these dependencies by design, offering a simpler pipeline that is easier to train and more robust. This streamlined structure ensures stable performance while maintaining competitive reconstruction fidelity.

V. CONCLUSION

Our study highlights the critical importance of an effective encoder in achieving high-quality audio generation and reconstruction. Extensive experiments demonstrated that the proposed architecture of spherical interpolation using Music2Latent outperforms traditional VAE-based models, delivering superior audio clarity and fidelity. The enhanced encoder design enables more precise latent space representations, directly contributing to improve the output quality.

A key novelty of this work is to apply dual-sided LoRA fine-tuning on the latent diffusion model. By fine-tuning both the forward and backward processes within the U-Net structure, we achieve greater stability and consistency in the generated audio. Experiments show that this approach significantly improves robustness across diverse inputs.

Looking forward, the strong reconstruction capabilities of the proposed model suggest promising extensions. One possible extension is local tone control within music, enabling users to adjust specific segments or instruments in a track with fine granularity. Another one is to leverage its latent space for advanced source separation, including isolating vocals, drums, or other components with minimal artifacts, leading to new possibilities in remixing and music production.

REFERENCES

- [1] H. Dudley, "The vocoder," *Bell Laboratories Record*, vol. 17, no. 3, pp. 122–126, 1939.
- [2] J. L. Flanagan and R. M. Golden, "Phase vocoder," *Bell System Technical Journal*, vol. 45, no. 9, pp. 1493–1509, 1966.
- [3] X. Serra and J. O. Smith, "Spectral modeling synthesis: A sound analysis/synthesis system based on a deterministic plus stochastic decomposition," *Computer Music Journal*, vol. 14, no. 4, pp. 12–24, 1990.
- [4] J. Engel, C. Resnick, A. Roberts, S. Dieleman, and D. Eck, "Neural audio synthesis of musical notes with wavenet autoencoders," in *Int. Conf. Machine Learning (ICML)*, 2017, pp. 1068–1077.
- [5] C. Donahue, K. Y. Yeh, J. McAuley, and J. P. Bello, "Gansynth: Adversarial neural audio synthesis," in *Int. Conf. Learning Representations (ICLR)*, 2019, pp. 1–17.
- [6] Z. Kong, A. Kumar, S. Pang, and S. Daulton, "Diffwave: A versatile diffusion model for audio synthesis," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021, pp. 1–17.
- [7] H. Liu, Z. Chen, Y. Yuan, *et al.*, "Audioldm: Text-to-audio generation with latent diffusion models," *arXiv preprint arXiv:2301.12503*, 2023.
- [8] K. Chen, Y. Wu, H. Liu, M. Nezhurina, T. Berg-Kirkpatrick, and S. Dubnov, "Musicldm: Enhancing novelty in text-to-music generation using beat-synchronous mixup strategies," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2024, pp. 1206–1210.
- [9] X. Niu, J. Zhang, and C. P. Martin, "Soundmorpher: Perceptually-uniform sound morphing with diffusion model," *arXiv preprint arXiv:2410.02144*, 2024.
- [10] P. Kamath, C. Gupta, and S. Nanayakkara, "Morphfader: Enabling fine-grained controllable morphing with text-to-audio models," in *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2025, pp. 1–5.
- [11] M. Pasini, S. Lattner, and G. Fazekas, "Music2latent: Consistency autoencoders for latent audio compression," *arXiv preprint arXiv:2408.06500*, 2024.
- [12] Z. Yang, Z. Yu, Z. Xu, *et al.*, "IMPUS: Image morphing with perceptually-uniform sampling using diffusion models," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: <https://openreview.net/forum?id=gG38EBe2S8>.
- [13] K. Zhang, Y. Zhou, X. Xu, B. Dai, and X. Pan, "Diffmorpher: Unleashing the capability of diffusion models for image morphing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 7912–7921.
- [14] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.
- [15] H. Liu, Y. Yuan, X. Liu, *et al.*, "Audioldm 2: Learning holistic audio generation with self-supervised pretraining," *IEEE/ACM Trans. Audio, Speech, and Language Processing*, vol. 32, pp. 2871–2883, 2024.
- [16] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1501–1510.
- [17] D. M. Brewster, *Introduction to Guitar Tone & Effects: An Essential Manual for Getting the Best Sounds from Electric Guitars, Amplifiers, Effect Pedals, and Digital Processors*. Hal Leonard Corporation, 2003.
- [18] S. G. Lee, W. Ping, B. Ginsburg, B. Catanzaro, and S. Yoon, "Bigvgan: A universal neural vocoder with large-scale training," *arXiv preprint arXiv:2206.04658*, 2022.