AUDIOSR: VERSATILE AUDIO SUPER-RESOLUTION AT SCALE

Haohe Liu[1], *Ke Chen*[2], *Qiao Tian*[3], *Wenwu Wang*[1], *Mark D. Plumbley*[1]

[1]Centre for Vision Speech and Signal Processing, University of Surrey,
[2]University of California San Diego,
[3]Speech, Audio & Music Intelligence (SAMI), ByteDance

ABSTRACT

Audio super-resolution is a fundamental task that predicts high-frequency components for low-resolution audio, enhancing audio quality in digital applications. Previous methods have limitations such as the limited scope of audio types (e.g., music, speech) and specific bandwidth settings they can handle (e.g., 4 kHz to 8 kHz). In this paper, we introduce a diffusion-based generative model, *AudioSR*, that is capable of performing robust audio super-resolution on versatile audio types, including sound effects, music, and speech. Specifically, *AudioSR* can upsample any input audio signal within the bandwidth range of 2 kHz to 16 kHz to a high-resolution audio signal at 24 kHz bandwidth with a sampling rate of 48 kHz. Extensive objective evaluation on various audio super-resolution benchmarks demonstrates the strong result achieved by the proposed model. In addition, our subjective evaluation shows that *AudioSR* can act as a plug-and-play module to enhance the generation quality of a wide range of audio generative models, including AudioLDM, Fastspeech2, and MusicGen. Our code and demo are available at https://audioldm.github.io/audiosr.

***Index Terms*—** audio super-resolution, diffusion model

1. INTRODUCTION

Audio super-resolution (SR) aims to estimate the higher-frequency information of a low-resolution audio signal, which yields a high-resolution audio signal with an expanded frequency range. High-resolution audio signals usually offer a better listening experience, which is often referred to as high fidelity. Due to the ability to enhance audio signal quality, audio super-resolution plays a significant role in various applications, such as historical recording restoration [1].

Previous studies on audio super resolution have primarily focused on specific domains, with a particular emphasis on speech super resolution. Early research decomposes the speech super resolution task into spectral envelope estimation and excitation generation [2]. Recent works employing deep learning techniques, such as AECNN [3], NuWave [4], and NVSR [5], have shown superior performance compared to traditional methods. In addition to speech, there have been efforts to address music super resolution, including studies on general music [6] and specific instruments [7].

Apart from the limited scope of audio, existing research on audio super resolution also has primarily been conducted in controlled experimental settings, limiting its applicability in real-world scenarios. An important challenge in audio super-resolution, as highlighted in [5], is the issue of bandwidth mismatch. This occurs when the bandwidth of the test data differs from that of the training data, leading to model failure. However, this issue has not received significant attention in the literature, as previous works typically assume consistent bandwidth settings for both training and testing data. In practice, the input bandwidth of test audio can vary due to factors such as limitations in recording devices, sound characteristics, or applied compression processes. Only a few studies have explored flexible input bandwidth, including NVSR [5] and NuWave2 [8]. However, these methods still primarily focus on speech super resolution without generalizing to a broader domain.

In this paper, we propose a novel method that addresses the limitations of previous work on limited audio types and controlled sampling rate settings. We introduce a method called *AudioSR*, which extends audio super resolution to a general domain, including all audible sounds such as music, speech, and sound effects. Moreover, *AudioSR* is capable of handling a flexible input sampling rate between 4kHz and 32kHz, covering most of the use cases in real-world scenarios. It has been found that the prior knowledge learned by the neural vocoder is helpful for reconstructing higher frequency components in audio super resolution tasks [5]. Therefore, *AudioSR* follows [5] to perform audio super resolution on the mel-spectrogram and utilizes a neural vocoder to synthesize the audio signal. To estimate the high-resolution mel-spectrogram, we follow AudioLDM [9] to train a latent diffusion model on learning the conditional generation of high-resolution mel-spectrogram from low-resolution mel-spectrogram. Our experiment demonstrates that *AudioSR* has achieved promising super resolution results on speech, music, and sound effects with different input sampling rate settings. Our subjective evaluation on enhancing the output of text-to-audio model AudioLDM [9], text-to-music model MusicGen [10], and text-to-speech model Fastspeech2 [11]

show that *AudioSR* can be a plug-and-play module for most audio generation models to enhance listening quality. Our contributions are summarized as follows:

- Our proposed *AudioSR* is the first system to achieve audio super resolution in the general audio domain, covering various types including music, speech, and sound effects.
- *AudioSR* can handle a flexible audio bandwidth ranging from 2kHz to 16kHz, and extend it to 24kHz bandwith with 48kHz sampling rate.
- Besides the promising results on audio super resolution benchmarks, *AudioSR* can also enhance audio quality as a plug-and-play module for models like AudioLDM, MusicGen, and FastSpeech2.

2. PROBLEM FORMULATION

Given an analog signal that has been discretely sampled at a rate of l samples per second, resulting in a low-resolution sequence of values $x_l = [x_i]_{i=1,2,...T\cdot l}$, the goal of audio super-resolution (SR) is to estimate a higher resolution signal $y_h = [y_i]_{i=1,2,...T\cdot h}$ sampled at a rate of h samples per second, where $h > l$ and T is the total duration in seconds. According to Nyquist's theory, x_l and y_h have maximum frequency bandwidths of $l/2$ Hz and $h/2$ Hz respectively. Therefore, the information contained between frequencies of $h/2 - l/2$ Hz is missing from x_l, and estimating this "missing" frequency data is the core objective of the super resolution task.

In this paper, we follow the method proposed in NVSR [5] to decompose the original audio super resolution task into two steps, including *(i) High-resolution Mel spectrogram Estimation*, and *(ii) Mel Spectrogram to Waveform Reconstruction with a Neural Vocoder.* Specifically, we first resample x_l to x_h using cubic interpolation, where x_h has a higher sampling rate h but with limited maximum bandwidth of $l/2$ Hz. We follow the steps in [5] to calculate the mel spectrogram of both x_h and y_h, resulting $X_{m\times n}$ and $Y_{m\times n}$, respectively, where m is the number of time frames and n is the number of mel frequency bins. Then we utilize a generative model to learning the process of estimating Y based on X, which is denoted as $\mathcal{G}_\theta : X \mapsto \hat{Y}$, where θ are the parameters of model $\mathcal{G}$. Finally, a neural vocoder is employed to reconstruct the high sampling rate audio signal based on the estimation of Y, which can be formulated as $\mathcal{V}_\phi : \hat{Y} \mapsto \hat{y}_h$, where $\mathcal{V}$ is the neural vocoder and ϕ are the learnable parameters.

3. METHOD

The architecture of the proposed *AudioSR* is demonstrated in Figure 1. After resampling the low-resolution audio x_l to x_h, the system first calculates both the STFT spectrogram and the mel spectrogram of x_h. Note that the higher frequency bins in X_h are empty because x_h does not have high-frequency information. X_h is then used as a conditioning signal to guide

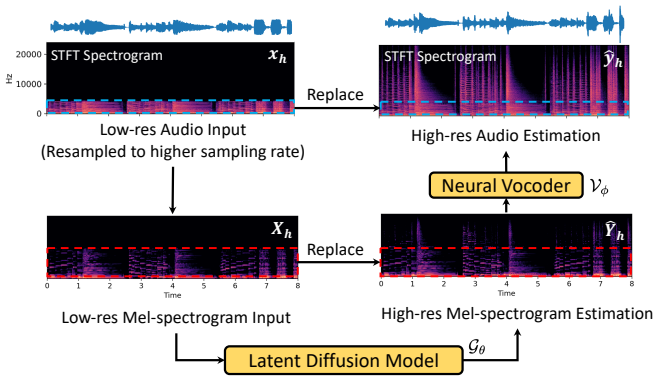

Fig. 1. The *AudioSR* architecture. The replacement-based post-processing aims to preserve the original lower-frequency information in the model output.

the pre-trained latent diffusion model to estimate the high-resolution mel spectrogram $\hat{Y}_h$. To ensure consistency in the low-frequency information between X_h and $\hat{Y}_h$, we replace the lower frequency part of $\hat{Y}_h$ with that of X_h. The mel-spectrogram after low-frequency replacement serves as the input to the neural vocoder, whose output is applied with a similar technique to replace the low-frequency information with that of the input low-resolution audio. We introduce the training of the latent diffusion model and neural vocoder in Section 3.1. The post-processing algorithm is elaborated in Section 3.2.

3.1. High-resolution Waveform Estimation

Latent diffusion model (LDM) has demonstrated promising results in various domains, including image synthesis [12] and audio generation [9]. In this study, we employ the LDM to estimate high-resolution mel-spectrograms. The training of our LDM is conducted within a latent space learned by a pre-trained variational autoencoder (VAE) $\mathcal{F}(\cdot)$. The VAE is trained to perform autoencoding with a small compressed latent space in the middle, denoted as $\mathcal{F} : X \mapsto z_0 \mapsto \hat{X}$. By leveraging the lower-dimensional representation z_0, the LDM can learn the generation of z_0 instead of X, resulting in a substantial reduction in computational cost. We adopt the methodology proposed in AudioLDM to optimize the VAE model, including the use of reconstruction loss, Kullback–Leibler divergence loss, and discriminative loss.

We follow the formulation introduced in AudioLDM [9] to implement the LDM, with improvements on the training objective, noise schedule, and conditioning mechanism. It has been found that the common noise schedule used in the diffusion model is flawed [13], particularly because the noise schedule in the final diffusion step of LDM does not correspond to a Gaussian distribution. To address this issue, we follow [13] to update the noise schedule to a cosine schedule. This adjustment ensures that a standard Gaussian distribution can be achieved at the final diffusion step during train-

ing. Additionally, we incorporate the velocity prediction objective [14] on reflection of using the new noise schedule. The final training objective of our LDM is

$$\text{argmin}_{\mathcal{G}_\theta} ||v_k - \mathcal{G}(z_k, k, \mathcal{F}_{\text{enc}}(X_l); \theta)||_2^2, \quad (1)$$

where z_k represents the data of z_0 at diffusion step $k \in [1, ..., K]$, $||\cdot||_2$ denotes the Euclidean distance, $\mathcal{F}_{\text{enc}}$ denotes the VAE encoder, and as described in [13], v_k is calculated based on z_0, representing the prediction target of $\mathcal{G}$ at time step k. We adopt the Transformer-UNet architecture proposed in [15] as $\mathcal{G}$. The input to $\mathcal{G}$ is obtained by concatenating z_k with the $\mathcal{F}_{\text{enc}}(X_l)$, which is the VAE latent extracted from the low-resolution mel-spectrogram X_l. To incorporate classifier-free guidance, following the formulation in [9], we replace $\mathcal{F}_{\text{enc}}(X_l)$ with an empty tensor at a random rate (e.g., 10%) during training. After training the latent diffusion model, we perform sampling using the DDIM sampler [16].

Neural Vocoder. The LDM is capable of estimating high-resolution mel spectrograms. However, since mel-spectrograms are not directly audible, we employ a neural vocoder based on HiFiGAN [17] to convert the mel-spectrograms into waveforms. To address the issue of spectral leakage when implementing the original HiFiGAN, we adopt the multi-resolution discriminator [18] into the HiFiGAN vocoder. We optimize the vocoder using diverse audio data, as discussed in Section 4, resulting in a vocoder that operates at a sampling rate of 48kHz and can work on diverse types of audio.

3.2. Post-processing and Pre-processing

Post-processing. The input low-resolution audio features X_h and x_h are identical to the lower frequency bands in the estimation target, Y_h and y_h. As a result, we can reuse the available information from X_h and x_h to enhance both the LDM output $\hat{Y}_h$ and neural vocoder output $\hat{y}_h$. To accomplish this, we first determine the 0.99 roll-off frequency c of the entire input audio based on an open-source method[1] applied to both X_h and the STFT spectrogram of y_h. Subsequently, we replace the spectrogram components below the cutoff frequency in the LDM output $\hat{Y}_h$ and vocoder output $\hat{y}_h$, with the corresponding information in the X_h and x_h, respectively. This post-processing method can ensure the final output does not significantly alter the lower-frequency information.

Pre-processing. To minimize the mismatch between model training and evaluation, we perform preprocessing to the input audio during evaluation with a lowpass-filtering operation. We use the same method in post-processing to calculate the 0.99 roll-off frequency and perform lowpass filtering with an order 8 *Chebyshev* filter.

4. EXPERIMENT

Training Datasets. The datasets used in this paper include MUSDB18-HQ [19], MoisesDB [20], MedleyDB [21], FreeSound[2] [22], and the speech dataset from OpenSLR[3], which are downloaded by following the link provided by VoiceFixer [1]. All the audio data used are resampled at 48kHz sampling rate. The total duration of the training data is approximately 7000 hours. We utilize all these datasets to optimize VAE, LDM, and HiFi-GAN.

Training Data Simulation. We follow the method introduced in NVSR [5] to simulate low-high resolution audio data pairs. Given a high-resolution audio data y_h, we first perform lowpass filtering to the audio with a cutoff frequency uniformly sampled between 2kHz and 16kHz. To address the filter generalization problem [3], the type of the lowpass filter is randomly sampled within Chebyshev, Elliptic, Butterworth, and Boxcar, and the order of the lowpass filter is randomly selected between 2 and 10.

Evaluation Datasets. We performed both subjective and objective evaluations. For subjective evaluations, we adopt the output of MusicGen (caption from MusicCaps [23]), AudioLDM (caption from AudioCaps [24]), and Fastspeech2 (transcription from LJSpeech [25]) to study if the *AudioSR* can enhance the quality of the generation. For MusicGen we use audio tagging[4] to filter out the non-musical generation output. Finally, we collected 50 samples from MusicGen, 50 samples from AudioLDM, and 20 samples from FastSpeech2, and processed them with *AudioSR* for subjective evaluations on listener preference. Besides, we curate three benchmarks for objective evaluation, including ESC50 (sound effect) [26], AudioStock (music)[5], and VCTK (speech) [5]. The AudioStock dataset is built by hand-picking 100 high-quality music with 10 different genres. We only use the fold-5 in the ESC50 dataset as the evaluation set.

Evaluation Metrics For objective evaluation, we adopt the Log-Spectral Distance (LSD) metric, as used in prior studies [3, 5]. Following the setup of [15], we conduct two types of subjective evaluation on Amazon Mturk[6]: Overall quality rating and preference comparison. In the overall quality rating, raters assign a score between 1 and 5 to reflect the audio quality. In the preference comparison, raters compare two audio files and select the one that sounds better.

5. RESULT

We trained two versions of *AudioSR* for evaluation: the basic *AudioSR* that works on arbitrary audio types and input sampling rates, and a speech data fine-tuned variant called *AudioSR-Speech*. Our primary baseline for comparison is NVSR [5], which employs a similar mel-spectrogram and vocoder-based pipeline for audio super resolution tasks. The main distinction between *AudioSR* and NVSR lies in the

[1] https://librosa.org/doc/main/generated/librosa.feature.spectral_rolloff.html

[2] https://labs.freesound.org/

[3] https://openslr.org/

[4] https://github.com/kkoutini/PaSST

[5] https://audiostock.net/

[6] https://www.mturk.com/

Objective Evaluation											Subjective Evaluation	
VCTK (Speech)				AudioStock (Music)				ESC-50 (Sound Effect)			ESC-50 (4kHz Cutoff Freq)	
Cutoff-frequency	4kHz	8kHz	12kHz	Cutoff-frequency	4kHz	8kHz	16kHz	4kHz	8kHz	16kHz	System	Overall Quality
GT-Mel	0.64	0.64	0.64	GT-Mel	0.61	0.61	0.61	0.84	0.84	0.84	GT-Mel	4.35
Unprocessed	5.15	4.85	3.84	Unprocessed	4.25	3.48	1.99	3.90	3.07	2.25	Unprocessed	3.01
NuWave [4]	1.42	1.36	1.22	NVSR-DNN	1.67	1.49	1.13	**1.64**	1.59	1.76	NVSR-DNN	2.84
NVSR [5]	**0.91**	**0.81**	0.70	NVSR-ResUNet	1.70	1.34	0.95	1.80	1.69	1.67	NVSR-ResUNet	3.16
AudioSR	1.30	1.11	0.94	*AudioSR*	**0.99**	**0.74**	**0.73**	1.74	**1.57**	**1.35**	*AudioSR*	**4.01**
AudioSR-Speech	1.03	0.82	**0.69**									

Table 1. Objective and subjective evaluation results for 48kHz audio super resolution of speech, music, and sound effect data with varying cutoff frequencies in the input audio. The objective metric used for evaluation is the LSD, where lower values indicate superior performance. The subjective metric measures the overall listening quality, with higher values indicating better performance.

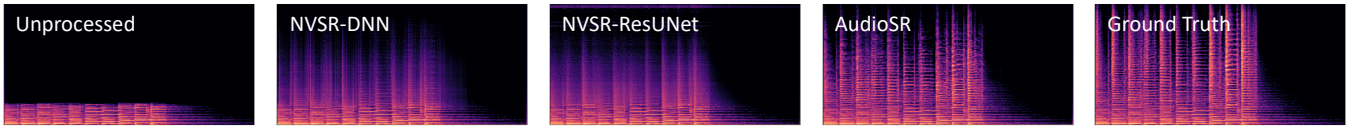

Fig. 2. Comparison of different systems. *AudioSR* performs significantly better than the baseline NVSR models.

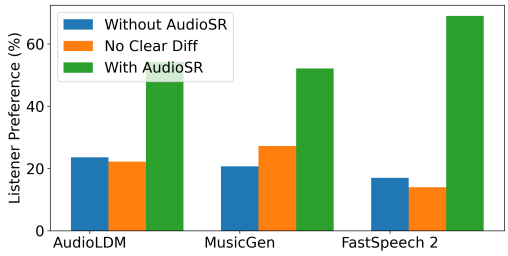

Fig. 3. Subjective evaluation shows that applying *AudioSR* for audio super-resolution on the output of audio generation models can significantly enhance the perceptual quality.

mel-spectrogram estimation approach: *AudioSR* utilizes a latent diffusion model, while NVSR employs either a multi-layer perceptron (NVSR-DNN) or a residual UNet (NVSR-ResUNet). For speech super resolution, we also compare with NuWave [4] as a baseline model, which also employs a diffusion model for audio super resolution.

Table 1 shows *AudioSR* has achieved promising results on both objective and subjective evaluation. For music super resolution, *AudioSR* achieves state-of-the-art performance across all cutoff frequency settings, outperforming the baseline NVSR model by a large margin. For speech super resolution, *AudioSR-Speech* achieves the best performance on the 24kHz to 48kHz upsampling task. Also, the comparison between *AudioSR* and *AudioSR-Speech* indicates that finetuning on a small domain of data can significantly improve the LSD.

The LSD metric does not always align with perceptual quality. In the 8kHz (i.e., 4kHz cutoff frequency) to 48kHz upsampling task on the ESC-50 dataset, we observed that NVSR-DNN achieved the best performance with an LSD score of 1.64. However, subjective evaluations indicated that the perceptual quality of NVSR-DNN was the worst with a score of 2.84, significantly lower than *AudioSR*'s score of 4.01. These findings suggest that LSD may not be a suitable evaluation metric for audio super resolution tasks on sound effect data, warranting further investigation in future research.

As depicted in Figure 3, our subjective preference test demonstrates that the utilization of *AudioSR* significantly enhances the perceptual quality of the AudioLDM, MusicGen, and FastSpeech2 output. It is worth noting that the output of MusicGen is already in a high sampling rate of 32kHz, which may contribute to the relatively high rate of "No Clear Difference" responses. However, MusicGen still exhibits a significantly improved perceptual quality after applying *AudioSR*.

6. CONCLUSION

This paper presents *AudioSR*, a 48kHz audio super-resolution model that is capable of working with diverse audio types and arbitrary sampling rate settings. Through evaluation of multiple audio super-resolution benchmarks, *AudioSR* demonstrates superior and robust performance on various types of audio and sampling rates. Additionally, our subjective evaluation highlights the effectiveness of *AudioSR* in enabling plug-and-play quality improvement for the audio generation models, including AudioLDM, MusicGen, and Fastspeech2. Future work includes extending *AudioSR* for real-time applications and exploring appropriate evaluation protocols for audio super-resolution in the general audio domain.

7. ACKNOWLEDGMENTS

This research was partly supported by the British Broadcasting Corporation Research and Development, Engineering and Physical Sciences Research Council (EPSRC) Grant EP/T019751/1 "AI for Sound", and a PhD scholarship from the Centre for Vision, Speech and Signal Processing (CVSSP), Faculty of Engineering and Physical Science (FEPS), University of Surrey. For the purpose of open access, the authors have applied a Creative Commons Attribution (CC BY) license to any Author Accepted Manuscript version arising.

8. REFERENCES

[1] H. Liu, Q. Kong, Q. Tian, Y. Zhao, D. Wang, C. Huang, and Y. Wang, "VoiceFixer: Toward general speech restoration with neural vocoder," *arXiv preprint:2109.13731*, 2021.

[2] J. Kontio, L. Laaksonen, and P. Alku, "Neural network-based artificial bandwidth expansion of speech," *Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 873–881, 2007.

[3] H. Wang and D. Wang, "Towards robust speech super-resolution," *Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 2058–2066, 2021.

[4] J. Lee and S. Han, "NuWave: A diffusion probabilistic model for neural audio upsampling," *arXiv preprint:2104.02321*, 2021.

[5] H. Liu, W. Choi, X. Liu, Q. Kong, Q. Tian, and D. Wang, "Neural vocoder is all you need for speech super-resolution," *INTERSPEECH*, pp. 4227–4231, 2022.

[6] S. Hu, B. Zhang, B. Liang, E. Zhao, and S. Lui, "Phase-aware music super-resolution using generative adversarial networks," *INTERSPEECH*, pp. 4074–4078, 2020.

[7] N. C. Rakotonirina, "Self-attention for audio super-resolution," in *International Workshop on Machine Learning for Signal Processing*. IEEE, 2021.

[8] S. Han and J. Lee, "NUWave 2: A general neural audio upsampling model for various sampling rates," *arXiv preprint:2206.08545*, 2022.

[9] H. Liu, Z. Chen, Y. Yuan, X. Mei, X. Liu, D. Mandic, W. Wang, and M. D. Plumbley, "AudioLDM: Text-to-audio generation with latent diffusion models," *International Conference on Machine Learning*, 2023.

[10] J. Copet, F. Kreuk, I. Gat, T. Remez, D. Kant, G. Synnaeve, Y. Adi, and A. Défossez, "Simple and controllable music generation," *arXiv preprint:2306.05284*, 2023.

[11] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T. Liu, "Fastspeech 2: Fast and high-quality end-to-end text to speech," in *International Conference on Learning Representations*, 2021.

[12] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 684–10 695.

[13] S. Lin, B. Liu, J. Li, and X. Yang, "Common diffusion noise schedules and sample steps are flawed," *arXiv preprint:2305.08891*, 2023.

[14] T. Salimans and J. Ho, "Progressive distillation for fast sampling of diffusion models," *International Conference on Learning Representations*, 2022.

[15] H. Liu, Q. Tian, Y. Yuan, X. Liu, X. Mei, Q. Kong, Y. Wang, W. Wang, Y. Wang, and M. D. Plumbley, "AudioLDM 2: Learning holistic audio generation with self-supervised pretraining," *arXiv preprint arXiv:2308.05734*, 2023.

[16] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *International Conference on Learning Representations*, 2020.

[17] J. Kong, J. Kim, and J. Bae, "HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," *Advances in Neural Information Processing Systems*, vol. 33, pp. 17 022–17 033, 2020.

[18] J. You, D. Kim, G. Nam, G. Hwang, and G. Chae, "GAN Vocoder: Multi-resolution discriminator is all you need," *arXiv preprint:2103.05236*, 2021.

[19] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, and R. Bittner, "MUSDB18-HQ - an uncompressed version of MUSDB18," Aug 2019.

[20] I. Pereira, F. Araújo, F. Korzeniowski, and R. Vogl, "MoisesDB: A dataset for source separation beyond 4-stems," *arXiv preprint:2307.15913*, 2023.

[21] R. M. Bittner, J. Salamon, M. Tierney, M. Mauch, C. Cannam, and J. P. Bello, "MedleyDB: A multitrack dataset for annotation-intensive mir research." in *ISMIR*, vol. 14, 2014, pp. 155–160.

[22] X. Mei, C. Meng, H. Liu, Q. Kong, T. Ko, C. Zhao, M. D. Plumbley, Y. Zou, and W. Wang, "WavCaps: A ChatGPT-assisted weakly-labelled audio captioning dataset for audio-language multimodal research," *arXiv preprint:2303.17395*, 2023.

[23] A. Agostinelli, T. I. Denk, Z. Borsos, J. Engel, M. Verzetti, A. Caillon, Q. Huang, A. Jansen, A. Roberts, M. Tagliasacchi *et al.*, "MusicLM: Generating music from text," *arXiv preprint:2301.11325*, 2023.

[24] C. D. Kim, B. Kim, H. Lee, and G. Kim, "AudioCaps: Generating captions for audios in the wild," in *NAACL-HLT*, 2019, pp. 119–132.

[25] K. Ito and L. Johnson, "The LJSpeech dataset," https://keithito.com/LJ-Speech-Dataset/, 2017.

[26] K. J. Piczak, "ESC: Dataset for environmental sound classification," in *International Conference on Multimedia*, 2015, pp. 1015–1018.