Audio Super-Resolution with Latent Bridge Models

Chang Li[1,2,3,*] **Zehua Chen**[1,2,*,†] **Liyuan Wang**[1] **Jun Zhu**[1,2,†]

[1]Department of CST, Tsinghua University, Beijing, China
[2]Shengshu AI, Beijing, China
[3]USTC, Hefei, China

{zhc23thuml, liyuanwang, dcszj}@tsinghua.edu.cn
lc_lca@mail.ustc.edu.cn

Abstract

Audio super-resolution (SR), *i.e.*, upsampling the low-resolution (LR) waveform to the high-resolution (HR) version, has recently been explored with diffusion and bridge models, while previous methods often suffer from sub-optimal upsampling quality due to their uninformative generation prior. Towards high-quality audio super-resolution, we present a new system with latent bridge models (LBMs), where we compress the audio waveform into a continuous latent space and design an LBM to enable a *latent-to-latent* generation process that naturally matches the *LR-to-HR* upsampling process, thereby fully exploiting the instructive prior information contained in the LR waveform. To further enhance the training results despite the limited availability of HR samples, we introduce frequency-aware LBMs, where the prior and target frequency are taken as model input, enabling LBMs to explicitly learn an *any-to-any* upsampling process at the training stage. Furthermore, we design cascaded LBMs and present two prior augmentation strategies, where we make the first attempt to unlock the audio upsampling beyond 48 kHz and empower a seamless cascaded SR process, providing higher flexibility for audio post-production. Comprehensive experimental results evaluated on the VCTK, ESC-50, Song-Describer benchmark datasets and two internal testsets demonstrate that we achieve state-of-the-art objective and perceptual quality for *any-to-48*kHz SR across speech, audio, and music signals, as well as setting the first record for *any-to-192*kHz audio SR. Demo at https://AudioLBM.github.io/.

1 Introduction

Audio super-resolution (SR) systems aim to generate the high-resolution (HR) audio waveform from the observed low-resolution (LR) waveform [47, 56, 109]. Audio SR plays a crucial roles in various applications, including historical recording restoration [73, 59, 74], hearing aids [30], and improvement of perceptual quality [62]. Previous works have explored audio SR with diverse methods such as mapping-based [60, 59], generative adversarial network (GAN)-based [69, 65], diffusion-based [72, 62], and recently proposed bridge-based [54, 45] methods. However, these efforts focus on the audio SR task in a limited scope, *e.g.*, the speech signals in a benchmark dataset, without generalizing to a broader domain of different audio types. To extend audio SR to multiple domains spanning speech signals, sound effects, and music samples, Liu et al. [62] has recently explored diffusion models in the latent space of mel-spectrogram, enabling *any-to-48 kHz* audio SR. A^2SB [45] has recently explored bridge models to synthesize the short-time Fourier transformation (STFT) representation, achieving *any-to-44.1 kHz* music SR.

[*]Equal Contribution.
[†]Corresponding Authors: Zehua Chen and Jun Zhu.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

However, the quality of audio SR at scale is still limited. One major cause is the potential mismatch between their generative frameworks and the audio SR process. As shown in Fig. 2, AudioSR [62] synthesizes the missing part of the mel-spectrogram latent representation from an uninformative Gaussian prior, ignoring the fact that the LR waveform contains informative cues about the HR target. A^{2}SB [45] formulates the audio SR analogous to image inpainting [58], where the removed regions are filled with Gaussian priors that remain uninformative about the target signal. Aiming for a high-quality audio SR system, we propose a latent bridge model (LBM) named AudioLBM, learning the *LR-to-HR* waveform SR process with a *latent-to-latent* generative framework. Specifically, we directly compress the audio waveform into a continuous latent representation, where the latent of the LR waveform demonstrates instructive information for the latent of the HR waveform and avoids the removed areas in previous works [41, 45]. Then, we design bridge models [14, 130] to enable a *latent-to-latent* generation process, which is inherently matched with the *LR-to-HR* waveform SR task and fully exploits the informative prior in the latent space.

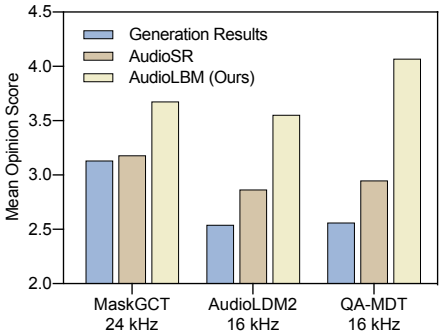

Figure 1: AudioLBM significantly improves the perceptual quality of text-to-speech [111], text-to-audio [63], and text-to-music [53] generation, and outperforms the state-of-the-art *any-to-48 kHz* SR system AudioSR [62].

Given the scarcity of high-resolution audio samples, which limits the generalization and scalability of audio SR systems, we propose frequency-aware LBMs that incorporate the prior and target frequencies as explicit conditioning inputs. The awareness of the frequency at both boundary distributions enables AudioLBMs to explicitly learn an *any-to-any* upsampling process across different frequency bands at the training stage, leading to improved upsampling performance to the target resolution at the sampling stage. Furthermore, upsampling the audio signal to a higher sampling rate, such as 96 or 192 kHz, provides enhanced high-frequency detail and allows flexibility in audio post-production. In this regard, we demonstrate that AudioLBM can be naturally extended to unlock the audio upsampling beyond 48 kHz with a cascaded design, where we present the prior augmentation strategies to reduce the cascading error in inference and the fine-tuning techniques to facilitate 192 kHz upsampling.

In summary, we make the following contributions in this work:

- We present a high-quality audio SR system across speech, sound effects, and music samples through modeling the *LR-to-HR* process with a *latent-to-latent* generative framework.
- We propose frequency-aware LBMs and cascaded LBMs, enabling an *any-to-any* training process to overcome data scarcity and empowering SR beyond 48 kHz for the first time.
- Zero-shot *any-to-48 kHz* audio SR evaluated on VCTK [118], Song-Describer-Dataset [68], and ESC-50 [79] demonstrate that our method outperforms prior systems with an average improvement of 21.5% in LSD [25] and 3.05% in ViSQOL [15] across audio and music, as well as achieves state-of-the-art objective and perceptual quality [87] for speech upsampling.

2 Related Work

Audio super-resolution. Previous works on audio SR can be broadly categorized into spectral-based and waveform-based methods. Spectral-based methods, often referred to as bandwidth extension (BWE), formulate the task as a spectral inpainting problem in either time-frequency space [60, 59, 69, 96, 72, 41, 122, 126, 19, 46] or latent spectral space [62, 37, 110]. In contrast, waveform-based methods directly generate high-resolution signals in the time domain [47, 76, 49, 32, 125, 51, 54]. However, most existing works remain limited in specific domains such as speech or isolated music genres [73, 71], and are evaluated under fixed input resolutions [69, 65, 96], simplified degradation assumptions [32, 55], or limited output sampling rates [72, 46, 69]. These limitations hinder generalization to real-world scenarios involving diverse content and degradation types, such as polyphonic pop music that combines vocals, background instrumentation, and sound effects, or handling extremely low-resolution inputs. Moreover, performing SR beyond 48 kHz (*e.g.*, 96 kHz and 192 kHz) remains unexplored, which provides further benefits in various professional applications,

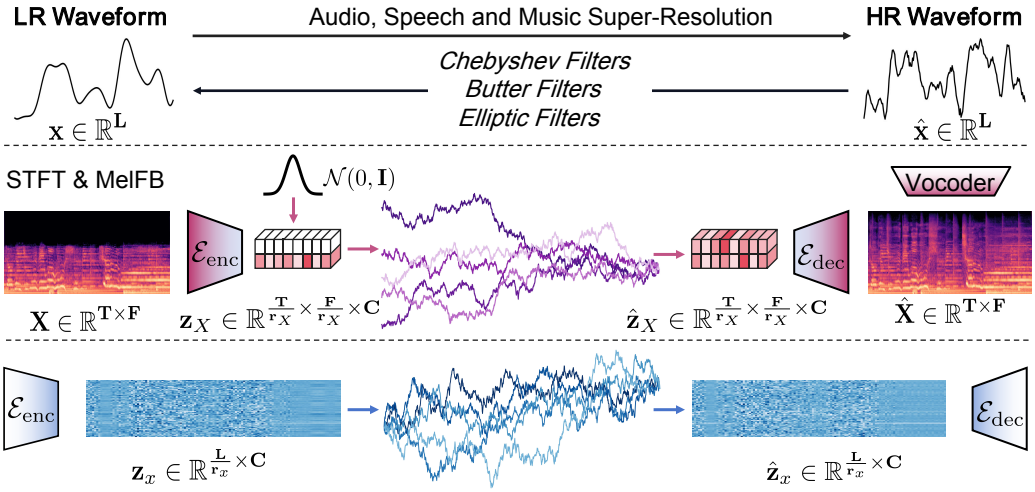

Figure 2: The top part shows how the low-resolution waveform is simulated during training via low-pass filtering, the middle part depicts the baseline method AudioSR [62] that synthesizes high-resolution content from Gaussian noise, and the bottom part presents overview of our proposed AudioLBM. It learns a *latent-to-latent* generation process between the low- and high-resolution waveform latent representations, namely $\boldsymbol{z}_{\boldsymbol{x}}^{\text{LR}} \in \mathbb{R}^{c_x \times \frac{L}{r_x}}$ and $\boldsymbol{z}_{\boldsymbol{x}}^{\text{HR}} \in \mathbb{R}^{c_x \times \frac{L}{r_x}}$, where L is waveform length. c_x and r_x are the channel dimension and compression ratio of waveform latent. In contrast, AudioSR operates in latent space $\boldsymbol{z}_{\boldsymbol{X}} \in \mathbb{R}^{c_X \times \frac{F}{r_X} \times \frac{T}{r_X}}$ of mel-spectrogram $X \in \mathbb{R}^{F \times T}$ with a *noise-to-latent* generation process, where T and F denote time and frequency bins of mel-spectrogram; c_X and r_X denote channel dimension and compression ratio of mel-spectrogram latent.

including mastering [57], post-production [7], spatial audio [8], and immersive content creation [101]. Although AudioSR [62] stands out as a scalable method achieving *any-to-48 kHz* SR across diverse domains, it suffers from two-stage compression pipeline and sub-optimal generative modeling paradigms for the SR task, leading to sub-optimal low-frequency fidelity, misalignment across frequency bands, and high-frequency artifacts [131], which ultimately limit perceptual SR quality. Therefore, building a unified and high-fidelity audio SR model that scales across speech signals, sound effects, and music samples remains a critical challenge. Detailed discussion of audio SR baseline methods is provided in Appendix H.2.2.

Bridge models. Tractable bridge models [130, 14, 64, 52] have received increasing attention for enabling more effective and efficient data-to-data generation paradigms. Unlike diffusion models [34, 98] that generate data by reversing a Markov process that gradually injects noise into the input, bridge models begin from a deterministic and informative prior, and interpolate stochastically toward the target distribution. Recent studies have explored bridge models in various applications, including image translation [52, 103], image restoration [58, 18], dense prediction [38], image editing [11], quality assessment [128, 129] and text-to-speech synthesis [14]. Building on this paradigm, BridgeSR [54] applies the Schrödinger Bridge (SB) framework directly in the waveform domain for speech SR, using a lightweight WaveNet-based score estimator [106, 44]. In parallel, A^{2}SB [45] employs the SB formulation in the STFT domain for music bandwidth extension and inpainting using a U-Net backbone [89, 22]. However, directly modeling in the data space inherently limits generalization and poses challenges for stable training and flexible scaling to higher-resolution SR. These limitations motivate us to develop a more adaptive generation framework tailored to audio SR.

3 Method

In this section, we introduce a latent bridge model (LBM) named AudioLBM to achieve high-quality super-resolution (SR). The key innovations include bridge-based audio upsampling in the continuous latent space of waveform, the frequency-aware model training process, and the cascaded design that unlocks audio SR beyond 48 kHz.

3.1 AudioLBM

Considering that the observed LR waveform has been an informative prior of the target HR waveform in the time-domain [54], we train a convolution-based variational autoencoder (VAE) [26] to directly compress the audio waveform into continuous representations, preserving the prior contained in the LR waveform for target generation. The details of VAE architecture and training configurations are introduced in Appendix B. Given the HR audio waveform $\boldsymbol{x}^{\text{HR}} \in \mathbb{R}^L$, we compute the LR counterpart $\boldsymbol{x}^{\text{LR}} \in \mathbb{R}^L$ with various *low-pass filters* to simulate the real-world degradations. Then, we compress them into latent representations $\boldsymbol{z}^{\text{HR}} \in \mathbb{R}^{c\times l}$ and $\boldsymbol{z}^{\text{LR}} \in \mathbb{R}^{c\times l}$ with a pre-trained encoder $\mathcal{E}(\boldsymbol{x})$, where $l = \frac{L}{r_x}$ and r_x is the down-sampling factor, constructing the boundary distributions and establishing the bridge process as follows.

Bridge process. Given $\boldsymbol{z}^{\text{LR}} \in \mathbb{R}^{c\times l}$ as the prior $\boldsymbol{z}_T$ at time step $t = T$ and $\boldsymbol{z}^{\text{HR}} \in \mathbb{R}^{c\times l}$ as the target $\boldsymbol{z}_0$ at time step $t = 0$, we build a bridge process [14] to connect the boundary distributions (see details in Appendix C):

$$\boldsymbol{z}_t = \frac{\alpha_t \bar{\sigma}_t^2}{\sigma_1^2}\boldsymbol{z}_0 + \frac{\bar{\alpha}_t \sigma_t^2}{\sigma_1^2}\boldsymbol{z}_T + \frac{\alpha_t \bar{\sigma}_t \sigma_t}{\sigma_1}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}), \tag{1}$$

where $\boldsymbol{z}_t$ denotes the noisy representation at time step t in the forward process of bridge models, and $\alpha_t, \bar{\alpha}_t, \sigma_t, \bar{\sigma}_t$ define the drift and diffusion terms in the associated stochastic differential equation (SDE), thereby controlling the noise schedule of the bridge process. Different from the forward process of diffusion models transforming the clean representation at the beginning $t = 0$ into standard Gaussian noise at the boundary $t = T$, bridge models replace the uninformative Gaussian prior with a Dirac prior $\delta_{\boldsymbol{z}^{\text{LR}}}$, and therefore facilitate the generation process to fully exploit the instructive information contained in the prior distribution *i.e.*, $\boldsymbol{z}^{\text{LR}}$ in AudioLBM.

Training objective. There exist multiple equivalent training objectives for bridge models. We empirically find that in the latent space, our AudioLBM achieves higher synthesis quality by using a noise predictor, compared to the data predictor used by recent bridge-related works in waveform [54] or mel-spectrogram domain [14]. In this regard, we formulate the denoising objective as:

$$\mathcal{L}_{\text{bridge}}(\theta) = \mathbb{E}_{\boldsymbol{z}_0=\boldsymbol{z}^{\text{HR}}, \boldsymbol{z}_T=\boldsymbol{z}^{\text{LR}}, t\sim\mathcal{U}[0,T]}\left[\left\|\boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t, t, \boldsymbol{z}_T) - \frac{\boldsymbol{z}_t - \alpha_t \boldsymbol{z}_0}{\alpha_t \sigma_t}\right\|_2^2\right], \tag{2}$$

where $\boldsymbol{z}_t$ is calculated with Eq. (1) at each training iteration.

Sampling process. The forward process of bridge models has a reverse process sharing the same marginal distribution $p_t(\boldsymbol{z}_t|\boldsymbol{z}_0, \boldsymbol{z}_T)$ [14, 130]. For signal generation, starting from the prior $\boldsymbol{z}^{\text{LR}}$, we use a first-order SDE-based sampler. From the time step s to the time step $t \in [0, s)$, the first-order discretization gives:

$$\boldsymbol{z}_t = \frac{\alpha_t \sigma_t^2}{\alpha_s \sigma_s^2}\boldsymbol{z}_s + \alpha_t\left(1 - \frac{\sigma_t^2}{\sigma_s^2}\right)\hat{\boldsymbol{z}}_0(\boldsymbol{z}_s, s) + \alpha_t \sigma_t \sqrt{1 - \frac{\sigma_t^2}{\sigma_s^2}}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}). \tag{3}$$

Given a noise predictor $\boldsymbol{\epsilon}_\theta$ well-trained with Eq. (2), we estimate the target $\hat{\boldsymbol{z}}_0$ in Eq. (3) with $\hat{\boldsymbol{z}}_0 = \frac{\boldsymbol{z}_t}{\alpha_t} - \sigma_t \boldsymbol{\epsilon}_\theta$ at each sampling step, and iteratively synthesize the generation target $\boldsymbol{z}^{\text{HR}}$. The pre-trained decoder $\mathcal{D}(\boldsymbol{z}^{\text{HR}})$ of VAE is then used to reconstruct the waveform $\boldsymbol{x}^{\text{HR}}$.

Bridge process vs. diffusion process. As shown in Fig. 2, the diffusion-based audio upsampling employs a conditional *noise-to-latent* sampling trajectory, where the prior provides limited information for the target. In comparison, our method employs a *latent-to-latent* sampling trajectory from $\boldsymbol{z}^{\text{LR}}$ to $\boldsymbol{z}^{\text{HR}}$, which has been aligned with the *LR-to-HR* audio upsampling. Moreover, in the continuous latent space directly compressed from the audio waveform, the representation of LR waveform provides instructive information for the generation target, rather than suffering from area removal in a latent space compressed by the STFT representation [45] or mel-spectrogram [122].

3.2 Frequency-aware LBMs

Compared to most text-to-speech [111, 24] or text-to-audio [61, 63, 27, 50] generation systems that synthesize the audio samples at a sampling rate below 24 or 44.1 kHz, it is more expensive to collect

the audio samples at a sampling rate of 48 kHz for training audio up-sampling models. To address the limitation of dataset scale, we propose frequency-aware LBMs, enabling an *any-to-any* upsampling process at the training stage. Specifically, given an audio signal $\boldsymbol{x} \in \mathbb{R}^L$, we first filter an HR waveform $\boldsymbol{x}^{\text{HR}} \in \mathbb{R}^L$ with a sampling rate $\text{SR}_{\boldsymbol{x}^{\text{HR}}}$[3] that is lower than the sampling rate of $\boldsymbol{x}$, while it has already provided the frequency band where the audio information predominantly concentrates (detailed in Appendix C). Then, we compute the LR version $\boldsymbol{x}^{\text{LR}} \in \mathbb{R}^L$ with a sampling rate $\text{SR}_{\boldsymbol{x}^{\text{LR}}}$ uniformly sampled from $\mathcal{U}\left(0, \text{SR}_{\boldsymbol{x}^{\text{HR}}}\right)$, constructing a new LR-HR data pair rather than using a fixed sampling rate for $\boldsymbol{x}^{\text{HR}}$.

At each training iteration, we first compress the $\boldsymbol{x}^{\text{HR}}$ and $\boldsymbol{x}^{\text{LR}}$ into latent $\boldsymbol{z}^{\text{HR}}$ and $\boldsymbol{z}^{\text{LR}}$, and then take their sampling rate $\text{SR}_{\boldsymbol{x}^{\text{HR}}}$ and $\text{SR}_{\boldsymbol{x}^{\text{LR}}}$ as model input, encouraging LBMs to explicitly learn an *any-to-any* upsampling process. Specifically, we extract a sinusoidal embedding of quantized $f_{\text{target}} = \text{Quantize}(\text{SR}_{\boldsymbol{x}^{\text{HR}}}/2)$ and continuous $f_{\text{prior}} = \text{SR}_{\boldsymbol{x}^{\text{LR}}}/2$ and prepend them as two additional tokens of our DiT [78, 4]-based noise prediction network, expanding the feature space to $\mathbb{R}^{c\times(l+2)}$. Towards stronger training performance, we leverage the constant scaling factor s for training bridge models in the waveform domain [54], rescaling the latent with $\tilde{\boldsymbol{z}} = s * \boldsymbol{z}$ for stable training, leading to the training objective of our frequency-aware LBMs:

$$\mathcal{L}_{\text{AudioLBM}}(\theta) = \mathbb{E}_{\tilde{\boldsymbol{z}}_0=\tilde{\boldsymbol{z}}^{\text{HR}},\tilde{\boldsymbol{z}}_T=\tilde{\boldsymbol{z}}^{\text{LR}},t\sim\mathcal{U}[0,T]}\left[\left\|\boldsymbol{\epsilon}_\theta(\tilde{\boldsymbol{z}}_t, t, \tilde{\boldsymbol{z}}_T, f_{\text{prior}}, f_{\text{target}}) - \frac{\tilde{\boldsymbol{z}}_t - \alpha_t\tilde{\boldsymbol{z}}_0}{\alpha_t\sigma_t}\right\|_2^2\right], \quad (4)$$

where the noisy latent $\tilde{\boldsymbol{z}}_t$ is computed from rescaled $\tilde{\boldsymbol{z}}_0$ and $\tilde{\boldsymbol{z}}_T$ with Eq. (1). At inference, the detected prior frequency f_{prior} and the target frequency $f_{\text{target}} = 48\text{ kHz}/2 = 24\text{ kHz}$ are both used to condition the model. Specifically, the input waveform is *low-pass filterd* by f_{prior} to suppress high-frequency artifacts, while f_{target} guides the model to generate full-band output up to the desired resolution.

3.3 Audio SR beyond 48 kHz

Cascaded LBMs. Beyond 48 kHz, the audio at an ultra-high sampling rate (*e.g.*, 96 kHz or 192 kHz) provides engineering advantages and post-processing flexibility [85]. However, scaling the upsampling system to such a sampling rate is constrained by the limited capacity of a single model and the scarcity of high-resolution training data, which has not been addressed to date. Here, we propose an extended version of AudioLBM that progressively improves the reconstructed signals from 48 kHz to higher sampling rates. Taking the upsampling to 96 kHz as an example, we introduce cascaded LBMs. We first train a VAE-based compression network $\mathcal{E}(\boldsymbol{x}^{\text{UHR}})$ for the ultra HR audio waveform $\boldsymbol{x}^{\text{UHR}}$, enabling us to design a new AudioLBM $\boldsymbol{\epsilon}_\theta^{\text{UHR}}$ in the latent space $\boldsymbol{z}^{\text{UHR}}$. At generation stage, the waveform $\boldsymbol{x}^{\text{LR}}$ is first upsampled to $\boldsymbol{x}^{\text{HR}}$ with the first-stage LBM, and then taken as the input of the second-stage model, further upsampled to $\boldsymbol{x}^{\text{UHR}}$ with a *latent-to-latent* process from $\boldsymbol{z}^{\text{HR}}$ to $\boldsymbol{z}^{\text{UHR}}$ with Eq. (3), as shown in Figure 3. In the scenario of 192 kHz upsampling, a third-stage LBM is cascaded. However, as the data has been extremely expensive to collect, which restricts the model performance, we propose a fine-tuning method to leverage the pretrained LBM at the second stage, improving the synthesis quality.

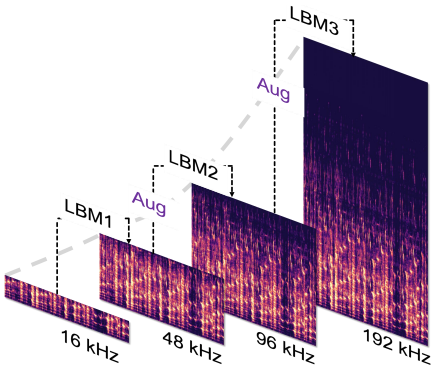

Figure 3: LBMs can be naturally extended into higher-resolution waveform generation with a cascaded paradigm, where prior augmentation is utilized to avoid cascading artifacts and accumulating errors between stages.

Prior augmentation. One of the potential limitations of cascaded generative models is the cascading errors in inference [35]. In cascaded LBMs, the LBM at the first stage approximates the ground-truth distribution of $\boldsymbol{x}^{\text{HR}}$, while we inevitably suffer from a mismatch between the synthesized data and the GT signal because of the imperfect learning of $\boldsymbol{\epsilon}_\theta$. Hence, we propose two prior

[3]To avoid ambiguity, we define the *low-pass filter (LPF)* by its sampling rate, corresponding to filtering with cutoff at its Nyquist frequency.

augmentation strategies in the waveform and latent space, respectively, for the cascaded bridge models $\boldsymbol{\epsilon}_\theta^{\text{UHR}}$, alleviating the mismatch issue with the first-stage LBM $\boldsymbol{\epsilon}_\theta$ and therefore strengthening the upsampling performance. Firstly, considering the difficulty for generative models to reconstruct fine-grained waveform features, we introduce a degradation operation in the waveform domain. Specifically, in the second upsampling stage, we randomly remove a small portion of high-frequency details near the Nyquist boundary of the prior and obtain a *low-pass-filtered* waveform, denoted as $\text{LPF}(\boldsymbol{x}^{\text{HR}})$. Furthermore, solely simulating the previous stage's output via waveform degradation is insufficient to address potential large-scale artifacts, which may be reflected in unclear harmonics or imbalanced energy distribution. To mitigate this, we further introduce latent-space blurring, which perturbs the audio by applying dynamic Gaussian smoothing to its latent representation along the time axis. Unlike previous diffusion-based [35] or flow-based [95] methods where the latent is corrupted with random noise for augmentation, the blurring strategy provides a deterministic degradation that aligns with the Dirac boundary distributions in bridge models [14], leading to the following training objective for the cascaded bridge model $\boldsymbol{\epsilon}_\theta^{\text{UHR}}$:

$$\mathcal{L}_{\text{CLBM}}(\theta) = \mathbb{E}_{\tilde{\boldsymbol{z}}_0=\tilde{\boldsymbol{z}}^{\text{UHR}},\tilde{\boldsymbol{z}}_T=\text{Blur}(\tilde{\boldsymbol{z}}^{\text{HR}}),t\sim\mathcal{U}[0,T]}\left[\left\|\boldsymbol{\epsilon}_\theta^{\text{UHR}}(\tilde{\boldsymbol{z}}_t, t, \tilde{\boldsymbol{z}}^{\text{HR}}, f_{\text{prior}}, f_{\text{target}}, b_r) - \frac{\tilde{\boldsymbol{z}}_t - \alpha_t\tilde{\boldsymbol{z}}_0}{\alpha_t\sigma_t}\right\|_2^2\right], \tag{5}$$

where $\tilde{\boldsymbol{z}}_t$ is calculated with the the ground-truth target $\tilde{\boldsymbol{z}}^{\text{UHR}}$ and the blurred latent prior $\text{Blur}(\tilde{\boldsymbol{z}}^{\text{HR}})$; the condition $\tilde{\boldsymbol{z}}^{\text{HR}}$ is the rescaled latent compressed from the degraded waveform $\text{LPF}(\boldsymbol{x}^{\text{HR}})$; the condition $b_r \sim \mathcal{U}(0, b_r^{\max})$ is the blurring ratio. Hence, the degradation level is also conditioned into the LBM with f_{prior} and b_r. In the training of cascaded LBMs $\boldsymbol{\epsilon}_\theta^{\text{UHR}}$, the small-scale details of the ground-truth prior $\boldsymbol{x}^{\text{HR}}$ has been removed and then the latent prior is blurred, enforcing the model to model the generation of $\boldsymbol{z}^{\text{UHR}}$ from a degraded prior $\text{Blur}(\tilde{\boldsymbol{z}}^{\text{HR}})$ and therefore facilitating the cascaded upsampling process in a robust manner. Further training details of cascaded are provided in Appendix D. At inference time, we select the optimal parameters b_r^* and waveform filtering extent via grid search to maximize cascading quality.

4 Experiment

In this section, we first describe the experimental setups and then present the experimental results of *any-to-48 kHz* upsampling and upsampling beyond 48 kHz with an in-depth analysis.

4.1 Experimental setup

Training setup. We train our compression and SR models on a mixed corpus comprising speech, audio, and music data. All recordings with an original sampling rate below 32 kHz are filtered out, resulting in a total of approximately 5,000 hours of training data. The detailed dataset information is summarized in Appendix G. For each SR stage, all data are resampled to the corresponding target sampling rate and randomly cropped for 5.12-second for the *any-to-48 kHz* and 48→96 kHz models and 2.56-second segments for the 96→192 kHz upsampling stage.

We apply dynamic low-pass filtering to simulate real-world complexity. For the *any-to-48 kHz* stage, the cutoff frequency is sampled from $\mathcal{U}(1{,}000, 20{,}000)$ Hz, the filter type is randomly chosen from {Chebyshev, Butterworth, Bessel, Elliptic}, and the order is sampled from $\mathcal{U}(2, 10)$. For the 48→96 kHz and 96→192 kHz stages, we exclusively use a Chebyshev Type-I filter with order 8, and sample cutoff frequencies from $\mathcal{U}(16{,}000, 48{,}000)$ Hz and $\mathcal{U}(32{,}000, 96{,}000)$, respectively. This design supports arbitrary input sampling rates in any stage and naturally serves as a deterministic waveform-based degradation strategy, as described in Sec. 3.3. For cascading inference, we filter off a frequency range of 4 kHz as waveform degradation and $b_r^{\max} = 1.0$, $b_r^* = 0.3$ for latent degradation for both 96 kHz and 192 kHz SR.

Evaluation setup. For the *any-to-48 kHz* task, we randomly sample 500 speech clips from the VCTK [118] test set, 300 audio samples from the ESC-50 fold-5 [79] and 300 music samples from the Song-Describer-Dataset (SDS) [68]. Since both ESC-50 and Song-Describer are natively recorded at 44.1 kHz, we resample all model outputs to 44.1 kHz to ensure a fair comparison. To evaluate performance on native 48 kHz content, we additionally use 300 randomly selected clips from our internal 48 kHz dataset (48Audio). For the 96 kHz and 192 kHz settings, we select 300 audio clips

VCTK	8 kHz→48 kHz											
Metric	Input	UDM+	NW1*	NW2	NVSR	Frep	FlowH	APBE	BriSR*	AuSR	Ours	Ours†
LSD↓	4.069	1.232	1.417	1.141	1.003	0.907	0.816	1.003	1.047	0.940	0.753	**0.742**
LSD(L)↓	0.187	0.216	0.268	0.294	0.357	0.272	0.194	0.224	**0.172**	0.486	0.773	0.708
LSD(H)↓	4.456	1.345	1.544	1.239	1.085	0.985	0.889	1.093	1.143	0.994	0.724	**0.712**
SSIM↑	0.519	0.691	0.661	0.654	0.734	0.755	0.784	0.742	0.660	0.809	0.893	**0.906**
SigMOS↑	3.136	3.068	2.936	2.838	2.831	2.743	2.792	3.082	2.998	2.846	3.023	**3.095**
Metric	Input	AuSR	Ours	Input	AuSR	Ours	Input	AuSR	Ours	Input	AuSR	Ours
48Audio	8 kHz→48 kHz			12 kHz→48 kHz			16 kHz→48 kHz			24 kHz→48 kHz		
LSD↓	2.931	1.468	**1.066**	2.637	1.365	**0.981**	2.351	1.304	**0.893**	1.788	1.234	**0.845**
ViSQOL↑	2.714	3.156	**3.281**	2.905	3.242	**3.331**	3.055	3.303	**3.509**	3.454	3.622	**3.801**
ESC-50	8 kHz→44.1 kHz			12 kHz→44.1 kHz			16 kHz→44.1 kHz			24 kHz→44.1 kHz		
LSD↓	3.042	1.537	**1.190**	2.720	1.412	**1.087**	2.435	1.292	**0.999**	1.810	1.067	**0.947**
ViSQOL↑	2.604	2.961	**3.003**	2.642	3.031	**3.089**	2.777	3.108	**3.234**	3.609	3.602	**3.641**
SDS	8 kHz→44.1 kHz			12 kHz→44.1 kHz			16 kHz→44.1 kHz			24 kHz→44.1 kHz		
LSD↓	4.515	1.728	**1.338**	4.070	1.501	**1.223**	3.632	1.352	**1.160**	1.788	1.166	**1.110**
ViSQOL↑	1.712	2.714	**2.744**	1.833	2.905	**2.939**	2.155	3.055	**3.168**	3.377	3.454	**3.603**

Table 1: Objective and perceptual metrics across speech, audio, and music. Baselines marked with * are our re-implementations. **Ours**† denotes our model trained exclusively on the VCTK training set.

(96/192 Audio) and 300 music (96/192 Music) excerpts from our internal dataset. Each 192 kHz clip is 2.56 seconds long, while all other evaluation samples are 5.12 seconds.

Long duration inference can be readily implemented using the MultiDiffusion [5, 20, 39] framework, and we empirically find that employing larger overlap sizes in the early sampling steps, while using smaller ones but shifting window positions in the later steps (even 0), is more effective in improving consistency and quality with LBMs.

For objective evaluation, we report the widely used Log-Spectral Distance (LSD) [25] and Spectral Structural Similarity (SSIM) [113, 59]. To assess perceptual quality, we use ViSQOL [15] for 48 kHz general audio and music, and SigMOS [87] for 48 kHz speech.

Baseline method. For evaluation under the *any-to-48 kHz* setting, unless specifically noted, we report only our zero-shot results without post-processing[62, 60] for low-frequency replacement. For speech, in addition to AudioSR (AuSR) [62], we compare against speech-specific baselines trained on the VCTK-train set, including UDM+ [120], NU-Wave (NW1) [49], NU-Wave2 (NW2) [32], NVSR [60], FlowHigh (FlowH) [122], Frepainter (Frep) [41], AP-BWE [65], and Bridge-SR (BriSR*) [54]. We follow the original Bridge-SR configuration, scale up the network to 10.6M parameters, and retrain it on both the VCTK-train set and our training dataset. For general audio and music, we compare with AudioSR under the *any-to-48 kHz* configuration. As there are no publicly available baselines for the 96 kHz and 192 kHz settings, we conduct only ablation studies using our own models. We scale our *any-to-48 kHz* model to 0.5B parameters, while the two subsequent models are scaled to 0.3B. All models are trained with an effective batch size of 128 and 1M iterations, while the fine-tuning procedure described in Sec. 3.3 takes an additional 0.5M steps. Unless otherwise specified, all ablation experiments are conducted with the same model size, using an effective batch size of 32 and trained for 0.2M steps without speech data. We use First-Order SDE sampling [14] for 50 steps for all stages. The compression architecture and additional model details are further discussed in Appendix B.

4.2 *Any-to-48 kHz* upsampling

Overall performance. As shown in Table 1, we evaluate our method across speech (48 kHz), general audio (48/44.1 kHz), and music datasets (44.1 kHz) using both objective metrics and perceptual scores. For the 8 kHz→48 kHz speech SR task, probabilistic generative approaches [62, 122] as

Dataset	ESC-50 (Audio, 16 kHz→48 kHz)					SDS (Music, 16 kHz→48 kHz)				
Metric	w/o Filter	w/o Input-A	w/o Target-A	Ours	only 48 kHz	w/o Filter	w/o Input-A	w/o Target-A	Ours	only 48 kHz
LSD↓	1.366	1.052	1.022	**0.994**	1.127	1.461	1.187	1.166	**1.124**	1.198
LSD-HF↓	1.448	1.093	1.069	**1.026**	1.173	1.567	1.262	1.239	**1.192**	1.275
SSIM↑	0.701	0.715	0.721	**0.722**	0.711	0.466	0.477	0.478	**0.484**	0.478

Table 2: Ablation studies on ESC-50 and SDS for the 16 kHz→48 kHz SR setting. **Input-A** denotes input-frequency awareness; **Target-A** denotes target-frequency awareness.

16→96 kHz on 96Audio and 96Music				16→192 kHz on 192Audio and 192Music					
Metric	Input	Direct	Ours	Metric	Input	Direct	48 kHz	96 kHz	Ours
LSD↓	3.068	1.406	**1.216**	LSD↓	2.929	1.913	2.744	2.314	**1.365**
LSD (0–48)↓	4.006	1.498	**1.083**	LSD (16–96)↓	3.191	1.452	2.304	1.372	**1.160**
ViSQOL↑	2.562	3.010	**3.330**	LSD (96–192)↓	2.880	2.244	2.879	2.879	**1.474**

Table 3: Comparison of super-resolution performance on 96 kHz and 192 kHz targets. **Direct** refers to directly trained *any-to-96/192 kHz* models. The **48 kHz** and **96 kHz** columns indicate intermediate outputs in the cascaded pipeline.

well as our method demonstrate clear advantages in capturing global spectral structure (SSIM) and high-frequency quality (LSD(H)). Notably, our zero-shot model significantly outperforms all previous methods in objective evaluation, reducing LSD-HF from FlowHigh's 0.889 to 0.724, improving SSIM from AudioSR's 0.809 to 0.893, and achieving higher perceptual quality (3.023 vs. 2.846 in SigMOS). Due to the domain diversity of the zero-shot training set, we find that the low-frequency noise of speech in some cases is amplified and mistaken for texture in sound effects, which can slightly degrade the perceptual quality. As a point of comparison, we further evaluate a variant trained only on VCTK-train (Ours†), which obtains even better performance and surpasses the GAN-based perceptual SoTA [65] (3.095 vs. 3.082 in SigMOS).

It is worth noting that the methods without compression networks [120, 49, 32, 54] often preserve low-frequency components almost perfectly, as indicated by LSD(L) scores close to the input. However, they often struggle to generate high-quality high-frequency contents. Alternatively, other methods [122, 60, 41, 62] employ post-processing strategies that directly replace the low-frequency band with the original input, thereby achieving near-perfect preservation. In our case, since the model already demonstrates strong overall LSD performance, we refrain from using such replacement strategies (detailed in Appendix E).

Ablation studies. We compare our AudioLBMs without filtering any low sampling-rate data (w/o Filter), without any frequency-awareness (w/o Input A), sorely with input-awareness (w/o Target A) and with both input and target frequency awareness (Ours). As shown in Table 2, the use of simple dataset filtering, input-frequency awareness, and output-frequency awareness leads to a steady and progressive improvement in 48 kHz SR performance. This leads to approximately a 20% reduction in LSD and a noticeable increase in SSIM across both audio and music domains, demonstrating the proposed frequency-awareness mechanism under a moderately filtered dataset. Furthermore, the training solely on 48 kHz data significantly reduces the diversity and scale of the training set, leading to sub-optimal performance and a notable drop in both LSD and SSIM. These results further validate the superiority of the proposed *any-to-any* training paradigm.

4.3 Upsampling beyond 48 kHz

Overall performance. Table 3 presents the comparison results under the 96 kHz and 192 kHz SR settings, where the input waveform is sampled at 16 kHz. We evaluate both cascaded LBMs and directly trained *any-to-96/192 kHz* LBMs. It can be seen that the cascaded LBMs consistently outperform the directly trained counterparts for SR beyond 48kHz. Benefiting from specific trained

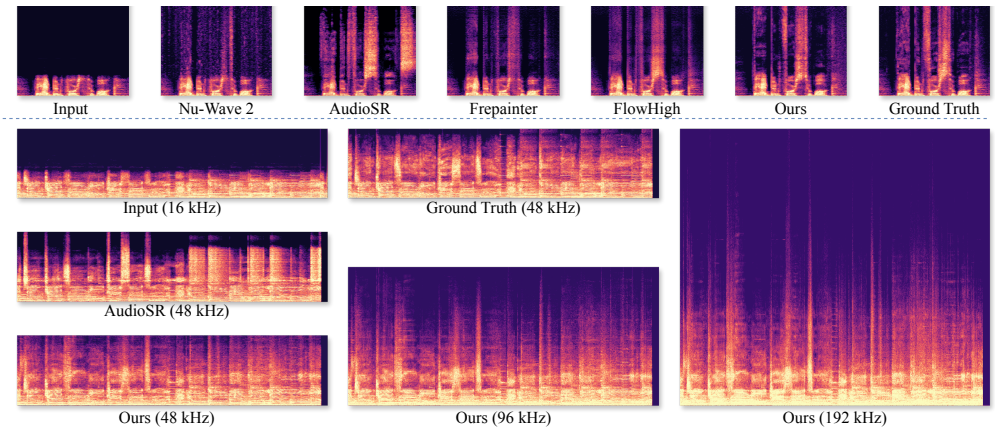

Figure 5: For case studies, we present the linear-amplitude STFT spectrograms of a 1.5-second speech segment from the VCTK-test set (sample *p360_102*) and a 5.12-second music clip.

models with sufficient quantity of corresponding audio data, the cascaded 96kHz LBMs achieves better content generation below 48kHz, reflected in a 0.415 reduction in LSD (0–48) and a 0.32 improvement in ViSQOL, which demonstrates the necessity of the cascaded paradigm. Furthermore, a 0.212 reduction in LSD(16-96) under the 192 kHz setting indicates that the proposed prior and conditioning augmentation strategy effectively enhances and refines the content generated by the previous stage.

To validate the effectiveness of the frequency-aware training paradigm in higher-resolution SR scenarios and further investigate the cascading augmentation, we perform extensive ablation studies conducted on the 96Music and 96Audio datasets under the 16→96 kHz SR setting (see Fig. 4). Specifically, we ablate the following components: (i) dataset filtering by removing samples with original sampling rates below 64 kHz, (ii) input/output frequency-awareness, (iii) waveform-domain filtering-based augmentation, (iv) latent-space blurring-based augmentation, and (v) the effect of dynamically sampled blurring ratios during training. Across both datasets, each component consistently contributes to performance improvement, as reflected in the overrall LSD. While the dynamic blur ratio provides additional gains, it also offers the flexibility to support both the cascaded SR pipeline and the standalone 48→96 kHz setting, where no distortion or degradation needs to be applied to the input signal. The SR experiments under real-world scenarios and a detailed comparison of augmentation parameters are provided in the Appendix E.

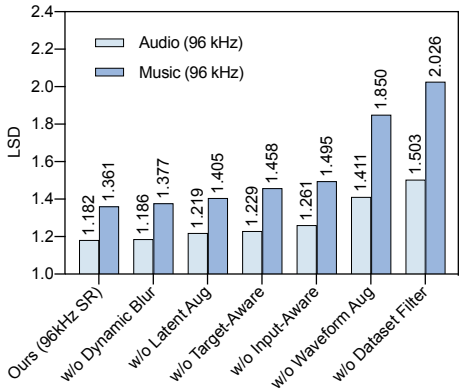

Figure 4: Ablation results on 96Audio and 96Music in the 16→96 kHz setting.

Case studies. Fig. 5 presents case studies on 8→48kHz speech and 16→192kHz music SR using our zero-shot model. For speech, we compare with recent SoTA methods [122, 62, 41, 32]. NU-Wave 2 demonstrates strong low-frequency preservation but struggles to recover high-frequency components under the challenging 6× upsampling setting. Frepainter and FlowHigh capture the general high-frequency contour, yet lack sufficient spectral energy and fine-grained details. AudioSR generates adequate high-frequency content, but often introduces over-boost and undesired artifacts. In contrast, our method achieves detailed full-band reconstruction, maintaining a better energy balance across frequency bands. For music, AudioSR fails to retain low-frequency consistency due to its spectrogram-based design, leading to spectral mismatches and weak harmonic modeling. In contrast, our method takes advantages of LBM modeling and frequency-awareness techniques to generate full-band audio with high fidelity at 48kHz. Further cascading to 96/192kHz introduces clearer high-frequency detail and harmonic structure, confirming the effectiveness of our architecture design and augmentation strategy.

5 Conclusion

In this work, we present AudioLBM, a novel audio upsampling system with bridge models in the continuous latent space of audio waveform, modeling the *LR-to-HR* audio upsampling process with a *latent-to-latent* generative framework. By proposing frequency-aware LBMs, we enable the learning of *any-to-any* upsampling process at the training stage to enlarge training data, thereby enhancing the super-resolution quality. We further present cascaded LBMs, empowering the audio upsampling beyond 48 kHz, and propose the prior augmentation strategies to reduce cascading errors. Comprehensive experimental results demonstrate that our AudioLBM outperforms previous audio upsampling systems by a large margin across speech, sound effects, and music signals, and enables high-quality upsampling to 96 kHz and 192 kHz for the first time in audio community. In the future, we plan to extend our super-resolution approach to additional modalities (*e.g.* image, video or time series), and further generalize it to a broader range of restoration tasks. As with other generative audio models, our method may raise concerns about potential misuse, including unauthorized synthesis of speech or music, imitation of artists' vocal identities, and challenges in verifying the authenticity of audio content, which could contribute to misinformation or undermine creative labor.

References

[1] Andrea Agostinelli, Timo I Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, et al. Musiclm: Generating music from text. *arXiv preprint arXiv:2301.11325*, 2023.

[2] Yuval Atzmon, Maciej Bala, Yogesh Balaji, Tiffany Cai, Yin Cui, Jiaojiao Fan, Yunhao Ge, Siddharth Gururani, Jacob Huffman, Ronald Isaac, et al. Edify image: High-quality image generation with pixel space laplacian diffusion models. *arXiv preprint arXiv:2411.07126*, 2024.

[3] AudiogenAI. AGC: Audio Generative Compression. `https://github.com/AudiogenAI/agc`, 2024. Accessed: 2025-05-21.

[4] Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22669–22679, 2023.

[5] Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. 2023.

[6] Rachel M Bittner, Justin Salamon, Mike Tierney, Matthias Mauch, Chris Cannam, and Juan Pablo Bello. Medleydb: A multitrack dataset for annotation-intensive mir research. In *Ismir*, volume 14, pages 155–160, 2014.

[7] Richard Black. Anti-alias filters: the invisible distortion mechanism in digital audio? In *Audio Engineering Society Convention*. Audio Engineering Society, 1999.

[8] Jens Blauert. *Spatial hearing: the psychophysics of human sound localization*. MIT press, 1997.

[9] Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, et al. Audiolm: a language modeling approach to audio generation. *IEEE/ACM transactions on audio, speech, and language processing*, 31:2523–2533, 2023.

[10] Jianrui Cai, Hui Zeng, Hongwei Yong, Zisheng Cao, and Lei Zhang. Toward real-world single image super-resolution: A new benchmark and a new model. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3086–3095, 2019.

[11] Clément Chadebec, Onur Tasar, Sanjeev Sreetharan, and Benjamin Aubin. Lbm: Latent bridge matching for fast image-to-image translation. *arXiv preprint arXiv:2503.07535*, 2025.

[12] Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-σ: Weak-to-strong training of diffusion transformer for 4k text-to-image generation. In *European Conference on Computer Vision*, pages 74–91. Springer, 2024.

[13] Tianrong Chen, Guan-Horng Liu, and Evangelos A Theodorou. Likelihood training of schr\" odinger bridge using forward-backward sdes theory. *arXiv preprint arXiv:2110.11291*, 2021.

[14] Zehua Chen, Guande He, Kaiwen Zheng, Xu Tan, and Jun Zhu. Schrodinger bridges beat diffusion models on text-to-speech synthesis. *arXiv preprint arXiv:2312.03491*, 2023.

[15] Michael Chinen, Felicia SC Lim, Jan Skoglund, Nikita Gureev, Feargus O'Gorman, and Andrew Hines. Visqol v3: An open source production ready objective speech and audio metric. In *International Conference on Quality of Multimedia Experience*, pages 1–6. IEEE, 2020.

[16] Youngwoo Cho, Minwook Chang, Gerard Jounghyun Kim, and Jaegul Choo. Progressive upsampling audio synthesis via effective adversarial training.

[17] Hyungjin Chung, Byeongsu Sim, Dohoon Ryu, and Jong Chul Ye. Improving diffusion models for inverse problems using manifold constraints. *Advances in Neural Information Processing Systems*, 35:25683–25696, 2022.

[18] Hyungjin Chung, Jeongsol Kim, and Jong Chul Ye. Direct diffusion bridge using data consistency for inverse problems. *Advances in Neural Information Processing Systems*, 36: 7158–7169, 2023.

[19] Marco Comunita, Zhi Zhong, Akira Takahashi, Shiqi Yang, Mengjie Zhao, Koichi Saito, Yukara Ikemiya, Takashi Shibuya, Shusuke Takahashi, and Yuki Mitsufuji. Specmaskgit: Masked generative modelling of audio spectrogram for efficient audio synthesis and beyond.

[20] Yusheng Dai, Chenxi Wang, Chang Li, Chen Wang, Jun Du, Kewei Li, Ruoyu Wang, Jiefeng Ma, Lei Sun, and Jianqing Gao. Latent swap joint diffusion for long-form audio generation. *arXiv e-prints*, pages arXiv–2502, 2025.

[21] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression. *arXiv preprint arXiv:2210.13438*, 2022.

[22] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[23] Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*, 2020.

[24] Zhihao Du, Yuxuan Wang, Qian Chen, Xian Shi, Xiang Lv, Tianyu Zhao, Zhifu Gao, Yexin Yang, Changfeng Gao, Hui Wang, et al. Cosyvoice 2: Scalable streaming speech synthesis with large language models. *arXiv preprint arXiv:2412.10117*, 2024.

[25] Adoram Erell and Mitch Weintraub. Estimation using log-spectral-distance criterion for noise-robust speech recognition. In *International Conference on Acoustics, Speech, and Signal Processing*, pages 853–856. IEEE, 1990.

[26] Zach Evans, Julian D Parker, CJ Carr, Zachary Zuckowski, Josiah Taylor, and Jordi Pons. Long-form music generation with latent diffusion.

[27] Zach Evans, CJ Carr, Josiah Taylor, Scott H Hawley, and Jordi Pons. Fast timing-conditioned latent audio diffusion. In *Forty-first International Conference on Machine Learning*, 2024.

[28] Zach Evans, Julian D Parker, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons. Stable audio open. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[29] Eduardo Fonseca, Xavier Favory, Jordi Pons, Frederic Font, and Xavier Serra. Fsd50k: an open dataset of human-labeled sound events. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:829–852, 2021.

[30] Christian Füllgrabe, Thomas Baer, Michael A Stone, and Brian CJ Moore. Preliminary evaluation of a method for fitting hearing aids with extended bandwidth. *International Journal of Audiology*, 49(10):741–753, 2010.

[31] Jiatao Gu, Shuangfei Zhai, Yizhe Zhang, Joshua M Susskind, and Navdeep Jaitly. Matryoshka diffusion models. In *The Twelfth International Conference on Learning Representations*, 2023.

[32] Seungu Han and Junhyeok Lee. Nu-wave 2: A general neural audio upsampling model for various sampling rates. *arXiv preprint arXiv:2206.08545*, 2022.

[33] Eric Harper, Somshubra Majumdar, Oleksii Kuchaiev, Li Jason, Yang Zhang, Evelina Bakhturina, Vahid Noroozi, Sandeep Subramanian, Koluguri Nithin, Huang Jocelyn, Fei Jia, Jagadeesh Balam, Xuesong Yang, Micha Livne, Yi Dong, Sean Naren, and Boris Ginsburg. NeMo: a toolkit for Conversational AI and Large Language Models. URL `https://github.com/NVIDIA/NeMo`.

[34] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[35] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *Journal of Machine Learning Research*, 23(47):1–33, 2022.

[36] Qingqing Huang, Daniel S Park, Tao Wang, Timo I Denk, Andy Ly, Nanxin Chen, Zhengdong Zhang, Zhishuai Zhang, Jiahui Yu, Christian Frank, et al. Noise2music: Text-conditioned music generation with diffusion models. *arXiv preprint arXiv:2302.03917*, 2023.

[37] Jaekwon Im and Juhan Nam. Flashsr: One-step versatile audio super-resolution via diffusion distillation. *arXiv preprint arXiv:2501.10807*, 2025.

[38] Haorui Ji, Taojun Lin, and Hongdong Li. Dpbridge: Latent diffusion bridge for dense prediction. *arXiv preprint arXiv:2412.20506*, 2024.

[39] Yuxuan Jiang, Zehua Chen, Zeqian Ju, Chang Li, Weibei Dou, and Jun Zhu. Freeaudio: Training-free timing planning for controllable long-form text-to-audio generation. *arXiv preprint arXiv:2507.08557*, 2025.

[40] Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. Audiocaps: Generating captions for audios in the wild. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 119–132, 2019.

[41] Seung-Bin Kim, Sang-Hoon Lee, Ha-Yeong Choi, and Seong-Whan Lee. Audio super-resolution with robust speech representation learning of masked autoencoder. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:1012–1022, 2024.

[42] Diederik P Kingma, Max Welling, et al. Auto-encoding variational bayes, 2013.

[43] Oddur Kjartansson, Supheakmungkol Sarin, Knot Pipatsrisawat, Martin Jansche, and Linne Ha. Crowd-sourced speech corpora for javanese, sundanese, sinhala, nepali, and bangladeshi bengali. In *SLTU*, pages 52–55, 2018.

[44] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. *arXiv preprint arXiv:2009.09761*, 2020.

[45] Zhifeng Kong, Kevin J Shih, Weili Nie, Arash Vahdat, Sang-gil Lee, Joao Felipe Santos, Ante Jukic, Rafael Valle, and Bryan Catanzaro. A2sb: Audio-to-audio schrodinger bridges. *arXiv preprint arXiv:2501.11311*, 2025.

[46] Pin-Jui Ku, Alexander H Liu, Roman Korostik, Sung-Feng Huang, Szu-Wei Fu, and Ante Jukić. Generative speech foundation model pretraining for high-quality speech extraction and restoration. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[47] Volodymyr Kuleshov, S Zayd Enam, and Stefano Ermon. Audio super resolution using neural networks. *arXiv preprint arXiv:1708.00853*, 2017.

[48] Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. High-fidelity audio compression with improved rvqgan. *Advances in Neural Information Processing Systems*, 36:27980–27993, 2023.

[49] Junhyeok Lee and Seungu Han. Nu-wave: A diffusion probabilistic model for neural audio upsampling. *arXiv preprint arXiv:2104.02321*, 2021.

[50] Sang-gil Lee, Zhifeng Kong, Arushi Goel, Sungwon Kim, Rafael Valle, and Bryan Catanzaro. Etta: Elucidating the design space of text-to-audio models. *arXiv preprint arXiv:2412.19351*, 2024.

[51] Yongjoon Lee and Chanwoo Kim. Wave-u-mamba: an end-to-end framework for high-quality and efficient speech super resolution. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[52] Bo Li, Kaitao Xue, Bin Liu, and Yu-Kun Lai. Bbdm: Image-to-image translation with brownian bridge diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1952–1961, 2023.

[53] Chang Li, Ruoyu Wang, Lijuan Liu, Jun Du, Yixuan Sun, Zilu Guo, Zhenrong Zhang, Yuan Jiang, Jianqing Gao, and Feng Ma. Quality-aware masked diffusion transformer for enhanced music generation. *arXiv preprint arXiv:2405.15863*, 2024.

[54] Chang Li, Zehua Chen, Fan Bao, and Jun Zhu. Bridge-sr: Schrödinger bridge for efficient sr. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[55] Kai Li and Yi Luo. Apollo: Band-sequence modeling for high-quality audio restoration. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[56] Teck Yian Lim, Raymond A Yeh, Yijia Xu, Minh N Do, and Mark Hasegawa-Johnson. Time-frequency networks for audio super-resolution. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 646–650. IEEE, 2018.

[57] Martin Link. Digital audio at 96 khz sampling frequency-pros and cons of a new audio technique. In *Audio Engineering Society Convention*. Audio Engineering Society, 1999.

[58] Guan-Horng Liu, Arash Vahdat, De-An Huang, Evangelos A Theodorou, Weili Nie, and Anima Anandkumar. I2sb: image-to-image schrödinger bridge. In *Proceedings of the 40th International Conference on Machine Learning*, pages 22042–22062, 2023.

[59] Haohe Liu, Qiuqiang Kong, Qiao Tian, Yan Zhao, DeLiang Wang, Chuanzeng Huang, and Yuxuan Wang. Voicefixer: Toward general speech restoration with neural vocoder. *arXiv preprint arXiv:2109.13731*, 2021.

[60] Haohe Liu, Woosung Choi, Xubo Liu, Qiuqiang Kong, Qiao Tian, and DeLiang Wang. Neural vocoder is all you need for speech super-resolution. *arXiv preprint arXiv:2203.14941*, 2022.

[61] Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley. Audioldm: text-to-audio generation with latent diffusion models. In *Proceedings of the 40th International Conference on Machine Learning*, pages 21450–21474, 2023.

[62] Haohe Liu, Ke Chen, Qiao Tian, Wenwu Wang, and Mark D Plumbley. Audiosr: Versatile audio super-resolution at scale. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1076–1080. IEEE, 2024.

[63] Haohe Liu, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong, Qiao Tian, Yuping Wang, Wenwu Wang, Yuxuan Wang, and Mark D Plumbley. Audioldm 2: Learning holistic audio generation with self-supervised pretraining. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024.

[64] Xingchao Liu and Lemeng Wu. Learning diffusion bridges on constrained domains. In *International Conference on Learning Representations*, 2023.

[65] Ye-Xin Lu, Yang Ai, Hui-Peng Du, and Zhen-Hua Ling. Towards high-quality and efficient speech bandwidth extension with parallel amplitude and phase prediction. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024.

[66] Ye-Xin Lu, Yang Ai, Zheng-Yan Sheng, and Zhen-Hua Ling. Multi-stage speech bandwidth extension with flexible sampling rate control. *arXiv preprint arXiv:2406.02250*, 2024.

[67] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11461–11471, 2022.

[68] Ilaria Manco, Benno Weck, Seungheon Doh, Minz Won, Yixiao Zhang, Dmitry Bogdanov, Yusong Wu, Ke Chen, Philip Tovstogan, Emmanouil Benetos, et al. The song describer dataset: a corpus of audio captions for music-and-language evaluation. In *Workshop on Machine Learning for Audio, Neural Information Processing Systems (NeurIPS)*. Neural Information Processing Systems, 2023.

[69] Moshe Mandel, Or Tal, and Yossi Adi. Aero: Audio super resolution in the spectral domain. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[70] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. librosa: Audio and music signal analysis in python. *SciPy*, 2015:18–24, 2015.

[71] Eloi Moliner and Vesa Välimäki. Behm-gan: Bandwidth extension of historical music using generative adversarial networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:943–956, 2022.

[72] Eloi Moliner, Jaakko Lehtinen, and Vesa Välimäki. Solving audio inverse problems with a diffusion model. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[73] Eloi Moliner, Filip Elvander, and Vesa Välimäki. Blind audio bandwidth extension: A diffusion-based zero-shot approach. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024.

[74] Eloi Moliner, Maija Turunen, Filip Elvander, and Vesa Välimäki. A diffusion-based generative equalizer for music restoration. *arXiv preprint arXiv:2403.18636*, 2024.

[75] Tu Anh Nguyen, Wei-Ning Hsu, Antony d'Avirro, Bowen Shi, Itai Gat, Maryam Fazel-Zarani, Tal Remez, Jade Copet, Gabriel Synnaeve, Michael Hassid, et al. Expresso: A benchmark and analysis of discrete expressive speech resynthesis. *arXiv preprint arXiv:2308.05725*, 2023.

[76] Viet-Anh Nguyen, Anh HT Nguyen, and Andy WH Khong. Tunet: A block-online bandwidth extension model based on transformers and self-supervised pretraining. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 161–165. IEEE, 2022.

[77] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210. IEEE, 2015.

[78] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

[79] Karol J Piczak. Esc: Dataset for environmental sound classification. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 1015–1018, 2015.

[80] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

[81] KR Prajwal, Bowen Shi, Matthew Lee, Apoorv Vyas, Andros Tjandra, Mahi Luthra, Baishan Guo, Huiyu Wang, Triantafyllos Afouras, David Kant, et al. Musicflow: Cascaded flow matching for text guided music generation. *arXiv preprint arXiv:2410.20478*, 2024.

[82] Zafar Rafii, Antoine Liutkus, Fabian-Robert Stöter, Stylianos Ioannis Mimilakis, and Rachel Bittner. The musdb18 corpus for music separation. 2017.

[83] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

[84] Chandan KA Reddy, Harishchandra Dubey, Kazuhito Koishida, Arun Nair, Vishak Gopal, Ross Cutler, Sebastian Braun, Hannes Gamper, Robert Aichner, and Sriram Srinivasan. Interspeech 2021 deep noise suppression challenge. *arXiv preprint arXiv:2101.01902*, 2021.

[85] Joshua D Reiss. A meta-analysis of high resolution audio perceptual evaluation. *Journal of the Audio Engineering Society*, 2016.

[86] Julius Richter, Simon Welker, Jean-Marie Lemercier, Bunlong Lay, and Timo Gerkmann. Speech enhancement and dereverberation with diffusion-based generative models. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:2351–2364, 2023.

[87] Nicolae-Cătălin Ristea, Babak Naderi, Ando Saabas, Ross Cutler, Sebastian Braun, and Solomiya Branets. Icassp 2024 speech signal improvement challenge. *IEEE Open Journal of Signal Processing*, 2025.

[88] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[89] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.

[90] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

[91] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4713–4726, 2022.

[92] Robin San Roman, Yossi Adi, Antoine Deleforge, Romain Serizel, Gabriel Synnaeve, and Alexandre Défossez. From discrete tokens to high-fidelity audio using multi-band diffusion. *Advances in Neural Information Processing Systems*, 36:1526–1538, 2023.

[93] Flavio Schneider, Ojasv Kamal, Zhijing Jin, and Bernhard Schölkopf. Moûsai: Efficient text-to-music diffusion models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8050–8068, 2024.

[94] Erwin Schrödinger. Sur la théorie relativiste de l'électron et l'interprétation de la mécanique quantique. In *Annales de l'institut Henri Poincaré*, volume 2, pages 269–310, 1932.

[95] Johannes Schusterbauer, Ming Gui, Pingchuan Ma, Nick Stracke, Stefan Andreas Baumann, Vincent Tao Hu, and Björn Ommer. Fmboost: Boosting latent diffusion with flow matching. In *European Conference on Computer Vision*, pages 338–355. Springer, 2024.

[96] Chenhao Shuai, Chaohua Shi, Lu Gan, and Hongqing Liu. mdctgan: Taming transformer-based gan for speech super-resolution with modified dct spectra. *arXiv preprint arXiv:2305.11104*, 2023.

[97] Ivan Skorokhodov, Sharath Girish, Benran Hu, Willi Menapace, Yanyu Li, Rameen Abdal, Sergey Tulyakov, and Aliaksandr Siarohin. Improving the diffusability of autoencoders. *arXiv preprint arXiv:2502.14831*, 2025.

[98] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[99] Christian J Steinmetz and Joshua D Reiss. auraloss: Audio focused loss functions in pytorch. In *Digital music research network one-day workshop (DMRN+ 15)*, 2020.

[100] Christian J Steinmetz, Jordi Pons, Santiago Pascual, and Joan Serrà. Automatic multitrack mixing with a differentiable mixing console of neural audio effects. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 71–75. IEEE, 2021.

[101] J Robert Stuart. Coding high quality digital audio. *Japan Audio Society*, 1997.

[102] Jiaqi Su, Yunyun Wang, Adam Finkelstein, and Zeyu Jin. Bandwidth extension is all you need. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 696–700. IEEE, 2021.

[103] Xuan Su, Jiaming Song, Chenlin Meng, and Stefano Ermon. Dual diffusion implicit bridges for image-to-image translation. *arXiv preprint arXiv:2203.08382*, 2022.

[104] Jiayan Teng, Wendi Zheng, Ming Ding, Wenyi Hong, Jianqiao Wangni, Zhuoyi Yang, and Jie Tang. Relay diffusion: Unifying diffusion process across resolutions for image synthesis. *arXiv preprint arXiv:2309.03350*, 2023.

[105] Qiao Tian, Yi Chen, Zewang Zhang, Heng Lu, Linghui Chen, Lei Xie, and Shan Liu. Tfgan: Time and frequency domain based generative adversarial network for high-fidelity speech synthesis. *arXiv preprint arXiv:2011.12206*, 2020.

[106] Aaron Van Den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, Koray Kavukcuoglu, et al. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 12, 2016.

[107] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.

[108] Gefei Wang, Yuling Jiao, Qian Xu, Yang Wang, and Can Yang. Deep generative learning via schrödinger bridge. In *International Conference on Machine Learning*, pages 10794–10804. PMLR, 2021.

[109] Heming Wang and DeLiang Wang. Towards robust speech super-resolution. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:2058–2066, 2021.

[110] Yuancheng Wang, Zeqian Ju, Xu Tan, Lei He, Zhizheng Wu, Jiang Bian, et al. Audit: Audio editing by following instructions with latent diffusion models. *Advances in Neural Information Processing Systems*, 36:71340–71357, 2023.

[111] Yuancheng Wang, Haoyue Zhan, Liwei Liu, Ruihong Zeng, Haotian Guo, Jiachen Zheng, Qiang Zhang, Xueyao Zhang, Shunsi Zhang, and Zhizheng Wu. Maskgct: Zero-shot text-to-speech with masked generative codec transformer. *arXiv preprint arXiv:2409.00750*, 2024.

[112] Zhifeng Wang, Diqun Yan, Rangding Wang, Li Xiang, and Tingting Wu. Speech resampling detection based on inconsistency of band energy. *Computers, Materials & Continua*, 56(2), 2018.

[113] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004.

[114] Simon Welker, Matthew Le, Ricky TQ Chen, Wei-Ning Hsu, Timo Gerkmann, Alexander Richard, and Yi-Chiao Wu. Flowdec: A flow-based full-band general audio codec with high perceptual quality. *arXiv preprint arXiv:2503.01485*, 2025.

[115] Yi-Chiao Wu, Israel D Gebru, Dejan Marković, and Alexander Richard. Audiodec: An open-source streaming high-fidelity neural audio codec. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[116] Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, et al. Sana: Efficient high-resolution image synthesis with linear diffusion transformers. *arXiv preprint arXiv:2410.10629*, 2024.

[117] Wanghan Xu, Xiaoyu Yue, Zidong Wang, Yao Teng, Wenlong Zhang, Xihui Liu, Luping Zhou, Wanli Ouyang, and Lei Bai. Exploring representation-aligned latent space for better generation. *arXiv preprint arXiv:2502.00359*, 2025.

[118] Junichi Yamagishi, Christophe Veaux, Kirsten MacDonald, et al. Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit (version 0.92). *University of Edinburgh. The Centre for Speech Technology Research (CSTR)*, pages 271–350, 2019.

[119] Zhuoyi Yang, Heyang Jiang, Wenyi Hong, Jiayan Teng, Wendi Zheng, Yuxiao Dong, Ming Ding, and Jie Tang. Inf-dit: Upsampling any-resolution image with memory-efficient diffusion transformer. In *European Conference on Computer Vision*, pages 141–156. Springer, 2024.

[120] Chin-Yun Yu, Sung-Lin Yeh, György Fazekas, and Hao Tang. Conditioning and sampling in variational diffusion models for speech super-resolution. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[121] Ruibin Yuan, Hanfeng Lin, Shuyue Guo, Ge Zhang, Jiahao Pan, Yongyi Zang, Haohe Liu, Yiming Liang, Wenye Ma, Xingjian Du, et al. Yue: Scaling open foundation models for long-form music generation. *arXiv preprint arXiv:2503.08638*, 2025.

[122] Jun-Hak Yun, Seung-Bin Kim, and Seong-Whan Lee. Flowhigh: Towards efficient and high-quality audio super-resolution with single-step flow matching. *arXiv preprint arXiv:2501.04926*, 2025.

[123] Chong Zhang, Yukun Ma, Qian Chen, Wen Wang, Shengkui Zhao, Zexu Pan, Hao Wang, Chongjia Ni, Trung Hieu Nguyen, Kun Zhou, et al. Inspiremusic: Integrating super resolution and large language model for high-fidelity long-form music generation. *arXiv preprint arXiv:2503.00084*, 2025.

[124] Kai Zhang, Jingyun Liang, Luc Van Gool, and Radu Timofte. Designing a practical degradation model for deep blind image super-resolution. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4791–4800, 2021.

[125] Kexun Zhang, Yi Ren, Changliang Xu, and Zhou Zhao. Wsrglow: A glow-based waveform generative model for audio super-resolution. *arXiv preprint arXiv:2106.08507*, 2021.

[126] Ting-Wei Zhang and Shanq-Jang Ruan. Vm-asr: A lightweight dual-stream u-net model for efficient audio super-resolution. *IEEE Transactions on Audio, Speech and Language Processing*, 2025.

[127] Wendi Zheng, Jiayan Teng, Zhuoyi Yang, Weihan Wang, Jidong Chen, Xiaotao Gu, Yuxiao Dong, Ming Ding, and Jie Tang. Cogview3: Finer and faster text-to-image generation via relay diffusion. In *European Conference on Computer Vision*, pages 1–22. Springer, 2024.

[128] Kanglei Zhou, Zikai Hao, Liyuan Wang, and Xiaohui Liang. Adaptive score alignment learning for continual perceptual quality assessment of 360-degree videos in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 31(5):2880–2890, 2025.

[129] Kanglei Zhou, Hubert P. H. Shum, Frederick W. B. Li, Xingxing Zhang, and Xiaohui Liang. Phi: Bridging domain shift in long-term action quality assessment via progressive hierarchical instruction. *IEEE Transactions on Image Processing*, 34:3718–3732, 2025. ISSN 1057-7149. doi: 10.1109/TIP.2025.3574938.

[130] Linqi Zhou, Aaron Lou, Samar Khanna, and Stefano Ermon. Denoising diffusion bridge models. *arXiv preprint arXiv:2309.16948*, 2023.

[131] Ge Zhu, Juan-Pablo Caceres, Zhiyao Duan, and Nicholas J Bryan. Musichifi: Fast high-fidelity stereo vocoding. *IEEE Signal Processing Letters*, 2024.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We confirm that the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper has discussed the limitations of the work, see Sec. 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: The paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The authors have described necessary information to reproduce the main experimental results of the paper, see Sec. 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [NA]

Justification: The authors will provide detailed implementation specifics in the supplementary materials, but the code is not available at this stage.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper has specified all the training and test details necessary to understand the results, see Sec. 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification:

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper has provided sufficient information on the computer resources, see Section 4.1.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The authors have discussed the broader impacts of this paper, see Section 5 Guidelines.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper does not pose such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [NA]

    Justification: The paper does not pose such risks.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and only permissively licensed data are selected the to train our model.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not release new assets

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

    Justification: TThe authors provide detailed descriptions of the subjective evaluation setup in the Appendix G.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The paper does not suffer these risks.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

Appendix

A Data processing

A.1 Selecting f_{target} in *any-to-any* training paradigm

Our *any-to-any* training strategy adopts a flexible target sampling rate f_{target}, where the input rate is randomly sampled as $f_{\text{prior}} \sim \mathcal{U}(f_{\min}, f_{\text{target}})$. While this design improves generalization and robustness, it may also suffer from the presence of unexpected distortion, device artifacts, or environmental noise in high-frequency regions [84], which frequently occur in in-the-wild recordings. As a result, training SR models with f_{target} exceeding the meaningful frequency range may potentially lead to overfitting to compression artifacts or aliasing patterns, rather than learning useful structure. Such failure cases are shown in Figure 6.

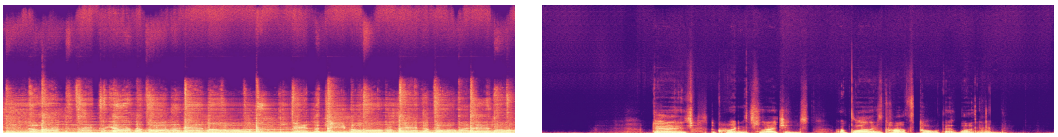

Figure 6: The impact of high-frequency noise on super-resolution outputs. Without an effective choice of f_{target}, the SR model hallucinates unnatural noise and artifacts in the high-frequency band.

These hallucinations not only degrade perceptual quality [102], but also propagate errors in subsequent cascade stages. To mitigate this, we restrict training targets to a more reliable range by enforcing $f_{\text{target}} \leq f_{\text{eff}}(x)$, where $f_{\text{eff}}(x)$ denotes the estimated effective bandwidth of input x that identifies the frequency range where signal energy is concentrated [10, 124, 102].

Estimating f_{eff}. Traditional methods estimate $f_{\text{eff}}(x)$ using energy-based heuristics, such as computing the log-ratio of total signal energy before and after low-pass filtering [112], or applying spectral roll-off thresholds as implemented in Librosa [70] based on cumulative spectral energy. However, these approaches are often sensitive to broadband or high-frequency noise, which can lead to inaccurate estimates of usable bandwidth (see Figure 7). CQT-Diff++ [73] proposes an iterative refinement strategy to more accurately approximate the true effective frequency band. However, it requires carefully tuned filtering parameters and introduces additional inference overhead.

To address these issues, we adopt a robust data preprocessing pipeline, as shown in Figure 8. Given a raw waveform x, we first estimate the effective cut-off frequency $f_{\text{eff}}(x)$ using curvature-aware spectral analysis, then apply a low-pass filter at $f_{\text{eff}}(x)$ to suppress high-frequency noise and artifacts, effectively boosting the fidelity and stability of downstream super-resolution models.

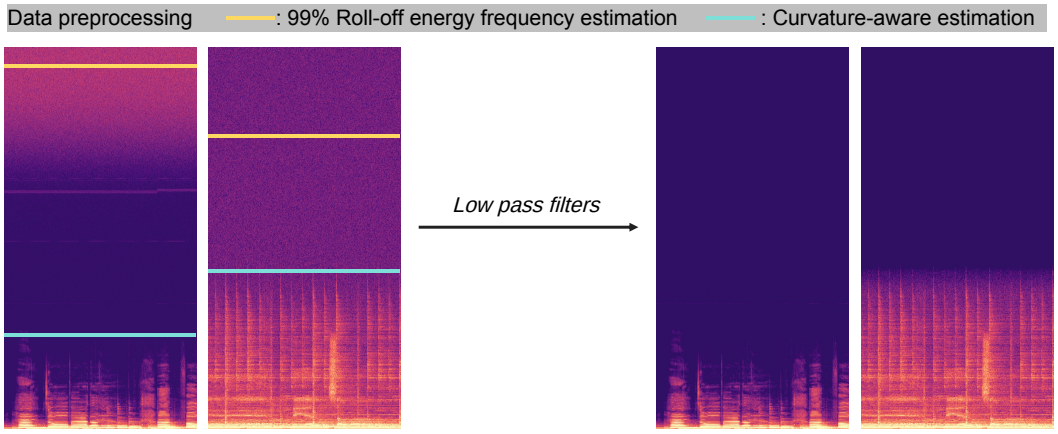

Figure 7: Our proposed data preprocessing pipeline estimates the effective cut-off frequency and removes spectral noise, leading to more suitable samples for SR training.

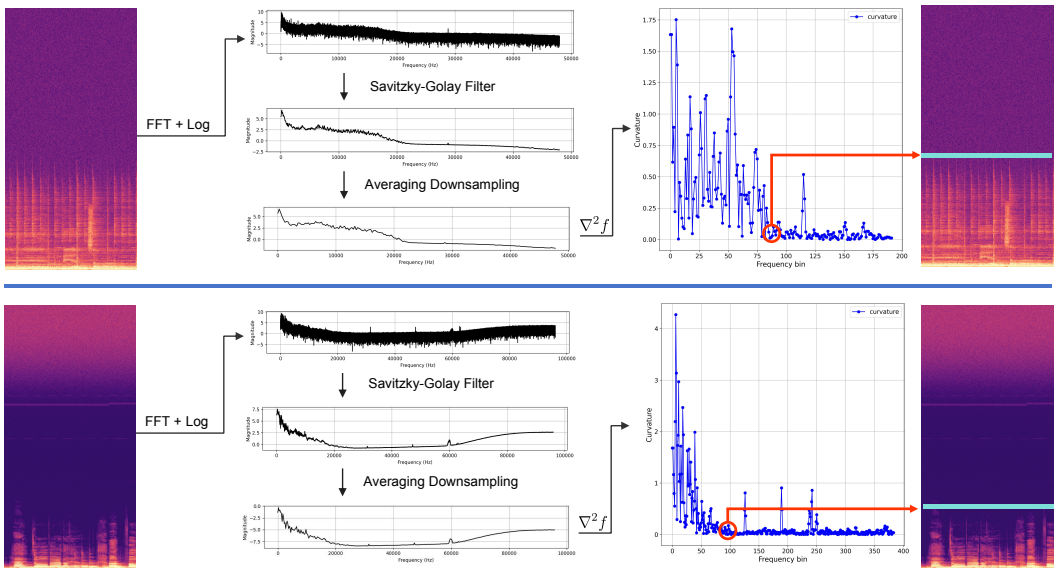

Figure 8: Overview of our proposed curvature-aware estimation pipeline. Given a raw waveform x, we first compute its magnitude spectrum via FFT, followed by Savitzky-Golay smoothing and local downsampling to obtain a denoised spectrum. We then compute the second-order curvature over the log-magnitude spectrum to detect the effective cut-off frequency $f_{\text{eff}}(x)$, which is used for downstream low-pass filtering. As a result, our method is robust to high-frequency noise.

We propose a curvature-aware approach to estimate the effective bandwidth $f_{\text{eff}}(x)$ of an audio signal, as illustrated in Figure 8. Our key motivation is that informative frequency regions exhibit stronger local variation, while noise or silence appears smoother—especially in the log-magnitude spectrum.

Given the densely sampled FFT magnitude $X(f) = |\text{FFT}(x)[f]|$, we apply Savitzky-Golay smoothing and downsampling to obtain $\bar{X}_i$, then define the truncation index i^* as the smallest index where local curvature[4] falls below a threshold ϵ_{sr}, and spectral energy drops below τ:

$$i^* = \arg\min_i \left\{ \max_{j \in [i, i+k]} |\nabla^2 \log \bar{X}_j| < \epsilon_{\text{sr}} \text{ and } \bar{X}_i < \tau \right\}. \tag{6}$$

The estimated bandwidth is then:

$$f_{\text{eff}} = \frac{i^*}{N} \cdot \frac{s_r}{2}. \tag{7}$$

During training, we low-pass filter each input at f_{eff} to suppress spurious high-frequency components as data preprocessing, improving stability and perceptual quality in subsequent SR stages.

A.2 Importance of f_{prior} detection for inference

At inference time, $f_{\text{prior}}(x)$ is estimated for each input sample with the same detection method of f_{eff} individually, allowing adaptive filtering tailored to the true spectral bandwidth of the audio. This ensures that super-resolution is performed on the informative signal subspace, leading to improved perceptual quality and generalization When generative models are used as the source of input audio, such as MaskGCT [111], the nominal output sampling rate is fixed (e.g., 24 kHz). However, the actual frequency content may vary depending on the speech prompt or training data, and may not occupy the full bandwidth of the input sampling rate. Similar phenomena have also been observed in 16 kHz generation models [61], where the effective bandwidth is often narrower than the nominal sampling rate. Consequently, using the declared sampling rate as the low-pass cutoff for super-resolution input may lead to incorrect reconstruction.

As illustrated in Figure 9, although the model declares a 24 kHz sampling rate, its generated waveform exhibits a clear spectral roll-off well below the Nyquist frequency (12 kHz).

[4] $\nabla^2 \log \bar{X}_i \approx \frac{1}{12}\left(-\log \bar{X}_{i+2} + 16 \log \bar{X}_{i+1} - 30 \log \bar{X}_i + 16 \log \bar{X}_{i-1} - \log \bar{X}_{i-2}\right)$.

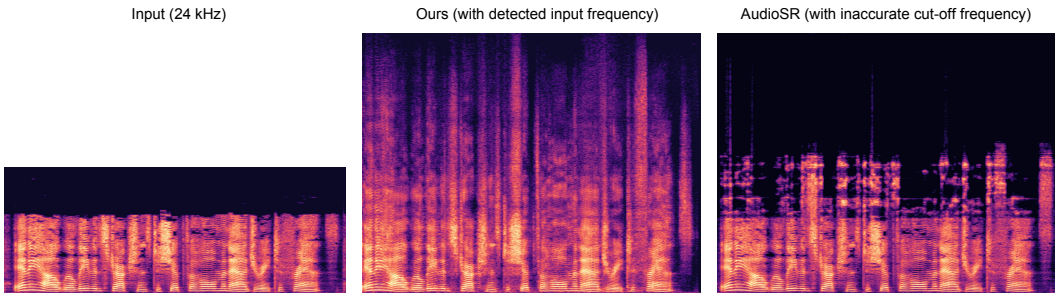

Figure 9: Super-resolution results for samples generated by MaskGCT [111] at 24 kHz, using both AudioSR and our proposed system. While the nominal sampling rate is 24 kHz, the actual spectral content does not span the full bandwidth. As shown in the result of AudioSR, using 24 kHz directly as the cutoff frequency results in incorrect high-frequency generation.

Using 12 kHz as the cutoff frequency by default leads to noticeable spectral discontinuities and artifacts in the output. In contrast, our method first estimates the effective bandwidth of the input audio and adjusts the filtering accordingly, thereby avoiding the generation of spurious high-frequency content and producing a more natural and coherent spectrum.

B Compression network

Variational autoencoders (VAEs) [42] include an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$. The encoder maps an input $\boldsymbol{x}$ to a latent representation $\boldsymbol{z} = \mathcal{E}(\boldsymbol{x})$, and the decoder reconstructs $\boldsymbol{x}$ from $\boldsymbol{z}$ via $\hat{\boldsymbol{x}} = \mathcal{D}(\boldsymbol{z})$. The training loss is

$$\mathcal{L}_{\text{VAE}} = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}}\left[\mathcal{R}(\mathcal{D}(\mathcal{E}(\boldsymbol{x})), \boldsymbol{x})\right] + \text{KL}(q_{\mathcal{E}}(\boldsymbol{z}|\boldsymbol{x}) \,\|\, \mathcal{N}(\mathbf{0}, \boldsymbol{I})), \tag{8}$$

where $\mathcal{R}$ is a reconstruction loss that measures the distance between the original sample $\boldsymbol{x}$ and the reconstructed sample $\mathcal{D}(\mathcal{E}(\boldsymbol{x}))$. $q_{\mathcal{E}}(\boldsymbol{z}|\boldsymbol{x})$ is the approximate posterior distribution over $\boldsymbol{z}$ given $\boldsymbol{x}$, and the KL term regularizes this posterior toward a standard Gaussian prior.

We follow the similar training network structure and pipeline with Stable-Audio-Open [28] and ETTA [50], which combines the following training objectives:

1. A multi-resolution STFT loss [99, 100]:

$$\mathcal{L}_{\text{MRSTFT}}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \sum_{i=1}^{m} \left(\frac{\|\text{stft}_i(\boldsymbol{x}) - \text{stft}_i(\hat{\boldsymbol{x}})\|_F}{\|\text{stft}_i(\boldsymbol{x})\|_F} + \frac{1}{T} \left\| \log \frac{\text{stft}_i(\boldsymbol{x})}{\text{stft}_i(\hat{\boldsymbol{x}})} \right\|_1 \right),$$

2. An adversarial hinge loss and feature matching loss from EnCodec [21]:

$$\mathcal{L}_{\text{adv}}(\hat{\boldsymbol{x}}, \boldsymbol{x}) = \sum_{k=1}^{K} \left[\max(0, 1 - D_k(\boldsymbol{x})) + \max(0, 1 + D_k(\hat{\boldsymbol{x}}))\right],$$

$$\mathcal{L}_{\text{feat}}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \frac{1}{KL} \sum_{k=1}^{K} \sum_{l=1}^{L} \frac{\left\|D_k^l(\boldsymbol{x}) - D_k^l(\hat{\boldsymbol{x}})\right\|_1}{\text{mean}\left(\left\|D_k^l(\boldsymbol{x})\right\|_1\right)},$$

   where D_k^l is the l-th layer of the k-th discriminator D_k.

3. The KL divergence loss:

$$\text{KL}(q_{\mathcal{E}}(\boldsymbol{z}|\boldsymbol{x}) \,\|\, \mathcal{N}(\mathbf{0}, \boldsymbol{I})).$$

B.1 48 kHz compression network

An effective compression network is critical to latent-based super-resolution. However, most existing audio compression models fail to maintain strong reconstruction quality, thereby limiting the upper bound of downstream performance. In this section, we analyze the effect of latent compression rate,

KL divergence, and latent scaling coefficient on both reconstruction and generation. Finally, we also compare our VAE to other baselines.

To ensure a fair evaluation of both reconstruction and generation capabilities, we train each ablated VAE configuration on our full speech corpus (details in Appendix G) for 200K steps, and pair it with a corresponding Latent Bridge Model (LBM) trained for 150K steps. All evaluations are conducted on the VCTK-test set. Throughout, we fix the VAE channel dimension to 64 for consistency, as we found that a higher channel dimension often leads to unstable training.

B.1.1 Compression rate

Typically, for the generation task, higher compression typically improves generative modeling without significantly compromising reconstruction [26]. However, in low-level tasks such as audio super-resolution, we observe that lower compression rates consistently yield better reconstruction and SR performance, as shown in Table 4. Nevertheless, lower compression introduces longer training time and higher inference cost (RTF), thus reducing deployment efficiency.

To this end, we conduct experiments under a fixed LBM training budget (96 A800-hours), and find that a compression rate of 512 achieves the best trade-off between efficiency and performance, delivering consistently superior results across both objective and subjective metrics.

Metric / Compression rate r_x	**2048**	**1024**	**512**	**256**
Reconstruction				
SSIM↑	0.925	0.930	0.946	**0.959**
LSD↓	0.664	0.679	0.665	**0.637**
SigMOS↑	3.099	3.083	**3.169**	3.142
Generation				
SSIM↑	0.911	0.912	0.925	**0.936**
LSD↓	0.730	0.714	0.710	**0.687**
SigMOS↑	3.088	3.025	**3.139**	3.088
Trained in equal time				
SSIM↑	0.900	0.912	**0.925**	0.905
LSD↓	0.730	0.742	**0.711**	0.825
SigMOS↑	3.086	2.958	**3.130**	3.029
RTF (A800, 100 step)	0.180	0.370	0.700	1.140

Table 4: Objective and subjective evaluation under different compression rates r_x. Metrics include SSIM, LSD, and SigMOS across reconstruction, generation, and time equalization settings.

B.1.2 KL divergence

For diffusion-based generation, increasing the KL divergence term is known to improve the alignment between the latent space and the Gaussian prior, thereby enhancing the diffusability of autoencoders and improving generation quality [117, 97]. However, in bridge models, the prior distribution is no longer a standard Gaussian but instead a Dirac distribution induced by the low-resolution waveform latent. In this setting, enforcing KL regularization may no longer be appropriate. Moreover, a smaller—or even zero—KL weight allows the latent space to prioritize reconstruction fidelity, potentially lifting the upper bound of super-resolution quality. To examine this, Table 5 reports results under a fixed compression rate of 512, with KL coefficients set to 0, 1e-7, 1e-5, 1e-3. Consistent with trends observed in Section B.1.1, we find that smaller KL values (as low as 0 or 1e-7) yield better reconstruction and super-resolution performance in the LBM setup—contrasting with the usual observations in diffusion-based generation tasks.

We also observe that reducing the KL weight causes the latent distribution to expand, reflected in significantly increased minimum and maximum values. This can potentially hinder training stability and downstream modeling. To mitigate this, we introduce a simple post-scaling strategy for the latent space. For KL values of {0, 1e-7, 1e-5}, we apply a fixed scaling factor s, chosen such

Metric/KL	0	1e-7	1e-5	1e-3
Reconstruction				
SSIM ↑	**0.942**	0.942	0.940	0.935
LSD ↓	0.651	**0.619**	0.637	0.664
pMOS ↑	**3.111**	3.079	3.103	3.030
mean	0.0506	0.1047	−0.0123	−0.032
std	3.9313	4.4400	1.1670	0.9407
min	−27.403	−29.362	−8.516	−4.619
max	27.615	34.446	8.506	4.774
Generation				
s **= 0.25**				
SSIM ↑	0.907	**0.907**	0.904	–
LSD ↓	0.742	**0.703**	0.723	–
pMOS ↑	**3.095**	3.060	3.073	–
s **= 1.0**				
SSIM ↑	0.905	0.906	0.903	0.899
LSD ↓	0.769	0.704	0.719	0.751
pMOS ↑	3.082	3.044	3.063	2.976

Table 5: Reconstruction and generation performance under varying KL divergence coefficients (VAE) and latent scaling factors (LBM). Reported on VCTK-test.

that $\mathrm{std}(z \cdot s) \approx 1$, thereby constraining the amplitude range of the latent representation. This normalization consistently improves both objective and subjective metrics.

B.1.3 Comparison with baselines

Therefore, we set the compression rate to 512, reduce the KL-divergence to 0, and apply a latent scaling factor of $s = 0.25$, resulting in a latent representation $\mathbf{z}$ with a frame rate of 100 Hz and 64 channels. After training for 1 million steps on the full dataset, we evaluate the reconstruction quality on the ESC-50 and VCTK-test sets and compare it against other baselines. As shown in Table 6, our model achieves the best reconstruction performance both in speech and audio domain across all metrics, which illustrates the advantage of using the compression network with relatively lower compression ratio and KL weight for AudioLBM.

B.2 Beyond 48 kHz compression networks

In our cascaded architecture, audio super-resolution above 48 kHz is achieved progressively across multiple stages. To avoid making the compression model (VAE) at higher sampling rates a bottleneck, or introducing artifacts that compromise details reconstructed in earlier stages, we use a unified compression ratio of 512 across all sampling rates.

However, high-resolution datasets (e.g., 96 kHz and 192 kHz) are relatively scarce. Training compression models directly on such data can significantly hurt generalization and lead to poorer reconstruction performance. Fortunately, we find that as long as the compression ratio remains consistent, a VAE trained at a lower sampling rate can be directly and effectively reused for higher sampling rates. Specifically, from the perspective of a 96 kHz VAE, 48 kHz audio can be interpreted as a time-compressed (i.e., faster-played) version of high-resolution audio, and thus still represents a valid subset of the high-resolution distribution. Since low-resolution audio data (such as 48 kHz) is more abundant, pretraining compression models on such data provides a strong and robust initialization, even in the presence of some distributional shift. As shown in Table 7, using a compression model pretrained on 48 kHz already yields strong results on the 96Audio dataset. However, despite the much smaller dataset size, directly pretraining at 96 kHz still achieves better performance under the test dataset with native 96 kHz distribution, suggesting a domain gap between the two resolutions. Nevertheless, a lightweight 200k-step fine-tuning on top of the 48 kHz-pretrained VAE surpasses both models, demonstrating strong transferability with only 20% training resource.

fr	Model	SSIM ↑	LSD ↓	LSD-LF ↓	LSD-HF ↓
VCTK (48 kHz)					
100 Hz	agc [3] (continuous)	0.913	0.762	0.766	0.741
50 Hz	agc [3] (discrete)	0.917	0.789	0.754	0.768
150 Hz	encodec [21]	0.887	1.126	0.761	1.176
75 Hz	audiodec [115]	0.922	0.939	0.944	0.937
75 Hz	flowdec [114]	0.925	0.619	0.560	0.630
100 Hz	**Ours**	**0.951**	**0.580**	**0.428**	**0.602**
ESC-50 (44.1 kHz)					
100 Hz	agc [3] (continuous)	0.730	0.836	0.748	0.840
50 Hz	agc [3] (discrete)	0.741	0.852	0.703	0.862
150 Hz	encodec [21]	0.704	0.925	0.910	0.922
100 Hz	dac [48]	0.734	0.799	0.735	0.801
75 Hz	audiodec [115]	0.695	0.999	0.969	0.997
75 Hz	flowdec [114]	0.717	0.899	0.810	0.908
100 Hz	**Ours**	**0.789**	**0.763**	**0.544**	**0.785**

Table 6: Comparison of reconstruction quality across baselines on VCTK and ESC-50 test sets (resampled to 48 kHz and 44.1 kHz, respectively).

Step	Phase	Model	fr	SSIM ↑	LSD ↓	LSD-LF ↓	LSD-HF ↓
96Audio dataset							
100w	pretrain	48 kHz-vae	50Hz	0.815	0.808	0.815	0.774
100w	pretrain	96 kHz-vae	50Hz	0.818	0.798	0.803	0.772
100w	pretrain	48 kHz-vae	100Hz	0.840	0.811	0.821	0.760
100w	pretrain	96 kHz-vae	100Hz	0.841	0.731	0.747	0.698
20w	finetune	finetune from 48Hz-vae	100Hz	**0.850**	**0.709**	**0.703**	**0.692**
192Audio dataset							
100w	pretrain	192 kHz-vae	100Hz	0.868	0.736	0.643	0.752
100w	pretrain	96 kHz-vae	100Hz	0.862	0.747	0.646	0.763
100w	pretrain	48 kHz-vae	100Hz	0.863	0.750	0.692	0.748
20w	finetune	finetune from 96 kHz-vae	100Hz	0.866	0.722	0.630	0.740
20w	finetune	finetune from 48 kHz-vae	100Hz	**0.871**	**0.713**	**0.603**	**0.737**

Table 7: VAE reconstruction results using different training strategies.

We observe similar trends in the 192Audio dataset: models fine-tuned from either 48 kHz or 96 kHz initialization outperform those trained from scratch at 192 kHz. Interestingly, fine-tuning from 48 kHz-pretrained VAE consistently leads to better results compared to fine-tuning from 96 kHz-pretrained VAE, highlighting that the scale of pretraining data matters more than resolution proximity. In other words, pretraining on larger datasets at lower sampling rates yields more transferable and stable compression models for high-resolution audio.

C Foundation of bridge models

Bridge models provide a principled and efficient solution for inverse problems and conditional generation in settings where the prior is non-Gaussian yet partially aligned with the target distribution. Given a sample from the prior distribution x_{prior} and its correspondence from target distribution x_{target}, In score-based generative models (SGMs) [98], a forward SDE is defined between $x_0 = x_{\text{target}} \sim p_{\text{target}}$, and $x_T = x_{\text{prior}} \sim p_{\text{prior}}$:

$$dx_t = f(x_t, t)\, dt + g(t)\, dw_t. \tag{9}$$

Here, $t \in [0, T]$ represents the current time step, with x_t denoting the state of data in the process. The drift is given by the vector field f, the diffusion by the scalar function g, w_t is the standard Wiener process, and the reference path measure p^{ref} describes the probability of paths from p_{prior} and p_{target}.

Under the above framework with boundary constraints, the Schrödinger bridge (SB) problem [94, 14] seeks to find a path measure p of specified boundary distributions that minimizes the Kullback-Leibler divergence D_{KL} between a path measure and reference path measure p_{ref}:

$$\min_{p \in \mathcal{P}_{[0,T]}} D_{\text{KL}}(p \parallel p^{\text{ref}}) \quad s.t. \quad p_0 = p_{\text{prior}},\ p_T = p_{\text{data}}, \tag{10}$$

where $\mathcal{P}_{[0,T]}$ denotes the collection of all path measures over the time interval $[0, T]$.

In SB theory [108, 13, 14], this specific SB problem can be expressed as a pair of forward-backwords linear SDEs:

$$dx_t = \left[f\left(x_t, t\right) + g^2(t)\nabla \log \Psi_t\left(x_t\right)\right] dt + g(t)dw_t, \tag{11}$$

$$dx_t = \left[f\left(x_t, t\right) - g^2(t)\nabla \log \widehat{\Psi}_t\left(x_t\right)\right] dt + g(t)d\overline{w}_t, \tag{12}$$

where the non-linear drifts $\nabla \log \Psi_t(x_t)$ and $\nabla \log \widehat{\Psi}_t(x_t)$ can be described by coupled partial differential equations (PDEs). A closed-form solution for SB [14] exists when Gaussian smoothing is applied with $p_0 = \mathcal{N}\left(x_{\text{HR}}, \epsilon_0^2 I\right)$ and $p_T = \mathcal{N}\left(x_{\text{LR}}, \epsilon_T^2 I\right)$, to the original Dirac distribution. By defining $\alpha_t = e^{\int_0^t f(\tau)d\tau}$, $\bar{\alpha}_t = e^{\int_1^t f(\tau)d\tau}$, $\sigma_t^2 = \int_0^t \frac{g^2(\tau)}{\alpha_\tau^2} d\tau$, and $\bar{\sigma}_t^2 = \int_t^1 \frac{g^2(\tau)}{\alpha_\tau^2} d\tau$, with boundary conditions and linear Gaussian assumption, tractable form of SB is solved as

$$\widehat{\Psi}_t = \mathcal{N}\left(\alpha_t x, \alpha_t^2 \sigma_t^2 I\right), \quad \Psi_t = \mathcal{N}\left(\bar{\alpha}_t y, \alpha_t^2 \bar{\sigma}_t^2 I\right) \tag{13}$$

under $\epsilon_T = e^{\int_0^T f(\tau)d\tau}\epsilon_0$ and $\epsilon_0 \to 0$ [14]. Therefore, marginal distribution of x_t at state t is also Gaussian:

$$p_t = \Psi_t \widehat{\Psi}_t = \mathcal{N}\left(\frac{\alpha_t \bar{\sigma}_t^2}{\sigma_1^2} x_0 + \frac{\bar{\alpha}_t \sigma_t^2}{\sigma_1^2} x_T, \frac{\alpha_t^2 \bar{\sigma}_t^2 \sigma_t^2}{\sigma_1^2} I\right). \tag{14}$$

D Cascaded LBMs

In this section, we provide a detailed description of the cascaded LBM, including analyses of the proposed waveform-based and latent-based augmentation techniques.

D.1 Waveform-based filtering augmentation

Samples generated from the *any-to-48 kHz* stage often exhibit small-scale high-frequency artifacts, as shown in Figure 10 (a). Directly feeding such artifacts into the next-stage super-resolution model can significantly degrade performance, as shown in Figure 10 (b), where high-frequency content becomes difficult to reconstruct. To address this, we apply Chebyshev Type-I low-pass filters (order=8) with sharp frequency roll-off in the spectral domain, as shown in Figure 10 (c)–(e), removing a frequency content of 8 kHz, 4 kHz, and 2 kHz, respectively, before feeding them into the next-stage model. The resulting outputs are shown in Figure 10 (f)–(h).

As shown in Figures 10 (g) and (h), moderate filtering removes unwanted high-frequency artifacts while preserving useful mid-frequency information, enabling the model to generate high-frequency details more effectively. However, overly aggressive filtering, such as in Figure 10 (g), eliminates too much information. This forces the model to simultaneously reconstruct missing mid-to-low frequency content and synthesize high frequencies, resulting in over-smoothed outputs compared to the result in Figure 10 (h).

The quantitative results presented in Table 8 further validate our findings. We conduct super-resolution experiments on both 48 kHz waveform generated by a preceding 16→48 kHz model (denoted as *Generated*) and 48 kHz audio obtained by downsampling real 96 kHz recordings (denoted as *Real*). For the *Generated* setting, waveform-based augmentation proves consistently effective across all filtering scales, with a moderate filtering ratio achieving the best trade-off between artifact removal and content preservation. Notably, for the *Real* setting, although inference benefits most from models trained without filtering, applying dynamic waveform-based augmentation during training still leads to improved performance. This suggests that waveform-based augmentation remains beneficial in both synthetic and real-data scenarios, highlighting its general applicability and robustness.

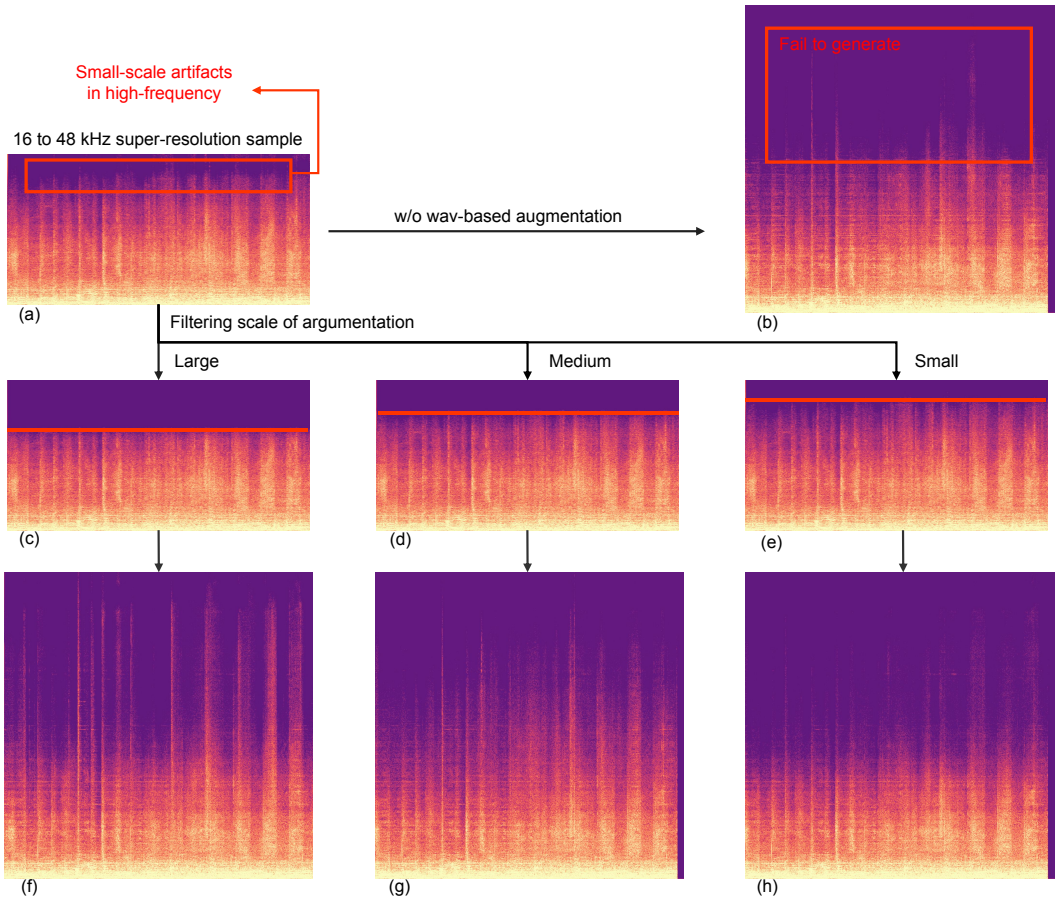

Figure 10: Visual comparison of waveform-based filtering strategies. (a) shows the baseline output from the *any-to-48 kHz* stage with high-frequency artifacts. (b) demonstrates that direct super-resolution from such inputs results in degraded high-frequency reconstruction. (c)–(e) apply *low-pass filters* with different cutoff frequencies, and the corresponding outputs after super-resolution are shown in (f)–(h).

48→96 kHz SR	**SR on Generated 48 kHz waveform**					**SR on Real 48 kHz waveform**				
Filter Ratio: $24-\delta$	18 kHz	19 kHz	20 kHz	22 kHz	w.o.	12 kHz	15 kHz	18 kHz	21 kHz	w.o.
LSD ↓	1.532	**1.495**	1.520	1.534	1.518	1.147	1.193	1.122	**0.978**	1.282
SSIM ↑	0.539	**0.541**	0.539	0.538	0.537	0.763	0.762	0.766	**0.767**	0.757

Table 8: Ablation study on waveform-based filtering augmentation for 48→96 kHz super-resolution on the 96Music dataset.

D.2 Latent-based blurring augmentation

To improve the alignment between the predicted and ground-truth latents in the first stage, we introduce a blurring augmentation strategy applied to the latent of the upsampled waveform from previous stage. Specifically, after performing waveform-based degradation—where a small portion of high-frequency spectral content is removed using *low-pass filtering*—we obtain a degraded latent representation $\tilde{\boldsymbol{z}}^{\mathrm{HR}} \in \mathbb{R}^{c_x \times \frac{L}{r_x}}$.

Specifically, we apply one-dimensional Gaussian blurring to each channel of $\tilde{z}^{\text{HR}}$, which helps smooth local fluctuations and provides a stable initial point for bridge sampling:

$$z_{\text{prior}}[c](t) = \text{Blur}(\tilde{z}^{\text{HR}}) = \sum_{\tau=-k}^{k} w(\tau)\, \tilde{z}^{\text{HR}}[c](t-\tau), \quad w(\tau) = \frac{1}{Z} \exp\left(-\frac{\tau^2}{2b_r^2}\right), \tag{15}$$

where $c \in [1, c_x]$, b_r denotes the blur ratio, k is half the kernel size, and Z is the normalization constant ensuring $\sum_\tau w(\tau) = 1$. We choose a kernel size of 5, which is significantly larger than $3b_r$, ensuring sufficient coverage of the Gaussian support. During training, the blur ratio b_r is uniformly sampled from the range $(0, 1)$.

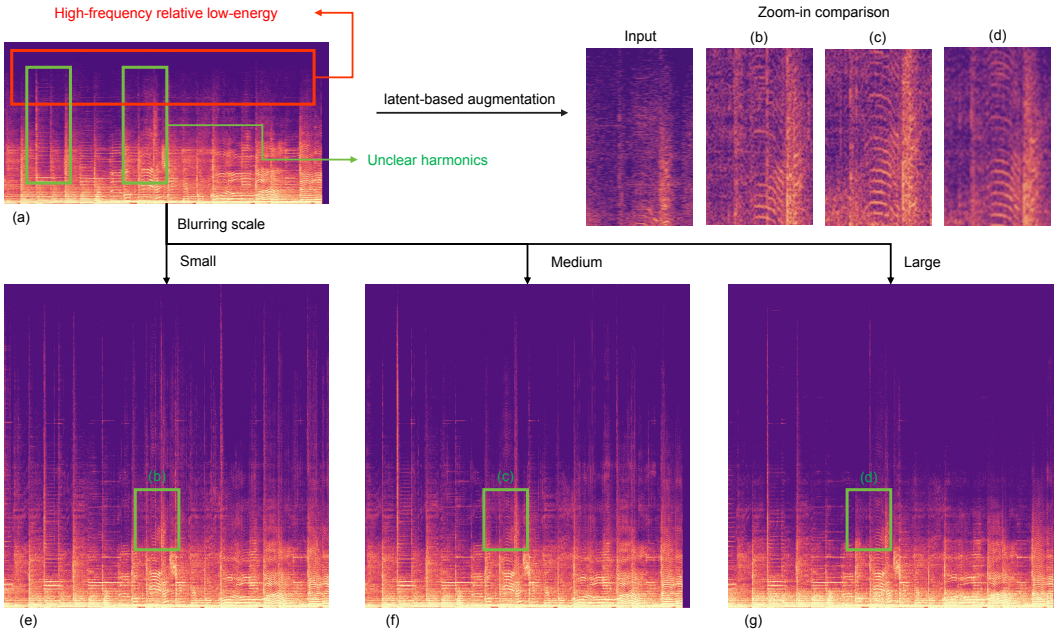

Figure 11: Visual comparison of latent-based blurring strategies. (a) shows the baseline output from the *any-to-48 kHz* stage, where unclear harmonics and low high-frequency energy can be observed. (e)–(g) illustrate the effect of applying different blur ratios $b_r = 0.05, 0.3, 0.5$ to the prior latent during bridge sampling. (b)–(d) show zoom-in views of (e)–(g) respectively for closer inspection.

As shown in Figure 11, in addition to the small-scale high-frequency artifacts described in Section D.1, the generated samples may also exhibit unclear harmonics or relatively low energy in the high-frequency band—both of which can negatively affect downstream stages and lead to cascading errors.

To address this, we apply different degrees of prior blurring with $b_r = 0.05$, $b_r = 0.3$, and $b_r = 0.5$, as shown in Figures 11 (e)–(g), with zoom-in comparisons in (b)–(d). The results demonstrate that latent-based augmentation effectively mitigates the imperfections in the input prior and improves overall reconstruction quality. It not only enhances high-frequency generation but also improves low-frequency reconstruction, leading to more balanced spectral energy and clearer harmonic structures.

Among them, Figure 11 (f) (with $b_r = 0.3$) yields the best result. A smaller blur ratio, as in Figure 11 (e), may not sufficiently correct low-frequency inconsistencies, while a larger blur ratio, as in Figure 11 (g), may overly smooth useful details, resulting in insufficient high-frequency generation. As shown in Table 9, applying latent-based blurring significantly improves performance over the no-augmentation baseline. A moderate blur ratio ($b_r = 0.3$ for generated, $b_r \to 0$ for real inputs) yields the best trade-off, effectively reducing high-frequency artifacts while preserving essential structure. From the results on *SR on Real 48 kHz waveform*, we observe a similar trend as discussed in Section D.1: applying stronger blurring generally leads to worse performance on real data, as low-frequency content is already accurate and does not require correction. This suggests that during inference, only a small or near-zero blur ratio is needed for real 48 kHz data. Nevertheless, applying a

48→96 kHz SR	**SR on Generated 48 kHz waveform**					**SR on Real 48 kHz waveform**				
Blur Ratio: b_r	0.4	0.3	0.25	0.20	w.o.	0.35	0.25	0.15	0.05	w.o.
LSD ↓	1.399	**1.361**	1.377	1.380	1.405	1.196	0.984	0.978	**0.976**	0.995
SSIM ↑	0.531	0.542	**0.543**	0.538	0.540	0.735	0.769	0.771	**0.771**	0.759

Table 9: Ablation study on latent-based blurring augmentation for 48→96 kHz super-resolution on the 96Music dataset.

mild level of blurring during training still outperforms the no-blur baseline (yielding a 0.02 reduction in LSD and a 0.01 improvement in SSIM), which may be because such operation promotes better understanding of low-frequency content, thereby enabling the model to generate high-frequency components more effectively.

E Additional results

In this section, we first continue our discussion on the cascaded stage and present a comparison of several additional augmentation strategies and their effects. We then analyze the role of the low-frequency replacement technique, comparing its use in our system and in other baseline methods.

E.1 Other augmentation techniques

We additionally compare three alternative augmentation strategies: (1) *latent noise*, where $\boldsymbol{z}_{\text{prior}} = \tilde{\boldsymbol{z}}^{\text{HR}} + n_r \cdot \mathcal{N}(0, \boldsymbol{I})$; (2) *pixel blur*, where $\boldsymbol{x}_{\text{prior}} = \text{Blur}(\tilde{\boldsymbol{x}}^{\text{HR}})$; and (3) *pixel noise*, where $\boldsymbol{x}_{\text{prior}} = \tilde{\boldsymbol{x}}^{\text{HR}} + n_r \cdot \mathcal{N}(0, \boldsymbol{I})$.

Method	latent blur (fixed)	latent blur (dyn)	pixel blur	latent noise	latent noise	latent noise	latent noise	pixel noise	w/o
b_r/n_r	0.3	0.3	0.1	0.1	0.2	0.3	0.4	0.1	–
LSD ↓	1.186	**1.182**	1.234	1.381	1.389	1.385	1.394	3.461	1.219
SSIM ↑	0.733	**0.738**	0.731	0.728	0.728	0.725	0.725	0.397	0.734

Table 10: Ablation study comparing different prior augmentation strategies for bridge-based super-resolution. The second row indicates the blur/noise ratio (b_r/n_r) used in each method.

We observe that latent blurring consistently outperforms other strategies, especially when using dynamic blur ratios during training. Pixel-level noise introduces instability and fails to converge, while random latent noise degrades gradually with increased noise levels. This suggests that randomized augmentation may not be suitable for bridge models, while deterministic methods such as blurring offer more stable and reliable guidance. Compared to pixel blurring, latent blurring provides a more global and structured form of regularization, which better aligns the low-frequency components with the ground-truth during training.

E.2 Low-frequency replacement

Low-frequency replacement is a widely adopted post-processing technique in super-resolution systems, particularly in super-resolution models based on compressed representations [60, 62, 122, 37]. Table 11 presents the results of applying post-processing within our cascaded architecture. Specifically, we apply low-frequency replacement to the output of the any-to-48 kHz model by substituting the sub-16 kHz frequency band with the corresponding band from the input waveform, resulting in the **48 kHz post** variant. This serves as the input for the subsequent 48→96 kHz super-resolution stage, producing **96 kHz post**. Finally, a second round of low-frequency replacement is performed on the output waveform to yield the **Final post** result.

From the table, we observe that performing post-processing at the 48 kHz stage significantly improves low-frequency fidelity, as reflected by improvements in LSD (0–48 Hz: 2.714 → 2.437) and ViSQOL

Band (Hz)	Metric	Direct	Cascaded (w/o post)	48 kHz Upsampling	48 kHz Post	96 kHz Post	Final Post
0–48	LSD ↓	3.785	1.552	2.714	2.437	1.314	**1.291**
0–16	LSD-LF ↓	0.186	0.485	0.811	**0.331**	0.522	0.342
16–48	LSD-HF ↓	5.071	1.738	1.614	1.596	1.221	**1.220**
48–96	LSD-SF ↓	3.329	1.603	3.532	3.141	1.507	**1.487**
0–48	ViSQOL ↑	1.954	2.602	3.073	**3.144**	3.119	3.137

Table 11: Ablation study of low-frequency replacement as a post-processing technique for 16→96 kHz super-resolution on the 96Music dataset. **Direct** denotes *any-to-96 kHz* model; **Cascaded (w/o post)** denotes the raw output from cascaded LBMs without replacement; **Post** indicates low-frequency replacement at corresponding stages.

Method	Modeling Space	Method Type	RTF ↓	NFE
Ours (any → 48 kHz)	Wav-VAE	Bridge	0.369	50
Ours (any → 96 kHz)	Wav-VAE	Cascaded Bridges	0.695	100
Ours (any → 192 kHz)	Wav-VAE	Cascaded Bridges	1.351	150
Bridge-SR [54] 48 kHz	Waveform	Bridge	1.670	50
UDM+ [60] 48 kHz	Waveform	Unconditional Diffusion	2.320	100
AudioSR [62] 48 kHz	Mel-VAE	Conditional Diffusion	0.948	50
Fre-painter [41] 48 kHz	Mel	GAN	0.009	1
NVSR [60] 48 kHz	Mel	GAN	0.033	1

Table 12: Real-time factor (RTF) results on an NVIDIA-A800 under the 48 kHz setting, as well as several baselines.

(3.073 → 3.144). More importantly, starting the 48→96 kHz super-resolution from a more accurate low-frequency foundation leads to better high-frequency generation, shown by the LSD-SF (48–96 Hz) improvement from 1.603 to 1.507.

Although the 48→96 kHz stage may slightly degrade the low-frequency components again, we find that performing an additional round of low-frequency replacement on the final 96 kHz waveform helps recover this degradation. As a result, we achieve a comparable overall fidelity in the 0–48 kHz band (ViSQOL: 3.137 vs. 3.144), confirming the effectiveness of post-processing in both any-to-48 kHz and cascaded LBMs settings.

E.3 Inference computational cost

To comprehensively evaluate the efficiency, we report the real-time factor (RTF) on an NVIDIA-A800 and several baselines under the 48 kHz setting in Table 12.

It is worth noting that under the 48 kHz setting, although our method is not as fast as GAN-based approaches such as [60, 41], due to the iterative sampling nature, it significantly outperforms other iterative-based methods including [54] and the previous state-of-the-art system AudioSR [62].

When scaling to higher sampling rates (e.g., 192 kHz), it naturally leads to slower inference speeds. We mitigate this impact by adopting a lighter network architecture (see Appendix G.2), so the inference speed does not increase linearly with the sampling rate. As shown in Table 12, even when upsampling to 192 kHz, our system is still faster than the 48 kHz waveform-domain bridge system [54].

E.4 Modeling space and other probabilistic methods

To isolate the contribution of the latent-bridge formulation, we conducted additional experiments using the same speech datasets (OpenSLR [43], Expresso [75], EARS [86], and VCTK-Train [118]) for training all models, and evaluated them on the VCTK-Test set. We compare the following variants without any additional modules, each trained with 500K steps on the any-to-48 kHz setting and tested on the 8-to-48 kHz SR setting with 50-step sampling:

Method	Modeling Space	SSIM ↑	LSD ↓	LSD-LF ↓	LSD-HF ↓	SigMOS [87] ↑
Diffusion	Mel-VAE latent	0.809	0.940	0.486	0.994	2.846
Rectified Flow	Mel	0.784	0.816	0.194	0.889	2.792
Bridge	Complex STFT	0.809	1.295	0.414	1.401	2.951
Bridge	Raw waveform	0.660	1.037	0.184	1.101	2.896
Rectified Flow	Ours Wav-VAE latent	0.880	0.751	0.793	0.722	2.892
Diffusion	Ours Wav-VAE latent	0.879	0.758	0.806	0.728	2.741
Bridge (Ours)	Ours Wav-VAE latent	**0.907**	**0.742**	**0.708**	**0.712**	**3.095**

Table 13: Comparison of different generative paradigms under the 8-to-48 kHz SR setting. All models are trained on the same datasets for 500K steps.

- Latent Diffusion (on Wav-VAE latents)
- Latent Rectified Flow (on Wav-VAE latents)
- Bridge-STFT (based on the Nemo [33] framework)
- Bridge-Waveform (based on Bridge-SR [54])
- Latent Bridge (Ours)

As shown in Table 13, our latent-bridge model achieves higher SSIM, lower LSD (objective quality), and better SigMOS (subjective quality), supporting that the latent-domain bridge formulation contributes significantly to the final performance, even when controlling for model architecture and training data.

F Inference techniques

In this section, we will discuss the low-pass filtering preprocessing method designed to handle real-world downsampled data.

F.1 Filtering preprocessing for real-world data

Metric	**Cheby1**	**Ellip**	**Bessel**	**Butter**
LSD ↓	**0.977**	0.996	1.193	1.017
LSD-LF ↓	0.814	0.814	0.811	**0.810**
LSD-HF ↓	**0.989**	1.011	1.236	1.036
SSIM ↑	**0.693**	0.690	0.664	0.686

Table 14: Comparison of different low-pass filters applied for super-resolution inference. Metrics are reported on 16→48 kHz super-resolution.

Even though we adopt various randomized low-pass filters during training to simulate real-world low-resolution audio, we empirically find that it is beneficial to additionally apply a low-pass filter to the upsampled low-resolution waveform during inference. In particular, the filter should ideally exhibit a sharp roll-off in the frequency domain.

As shown in Table 14, we compare four classical low-pass filters—Chebyshev Type I, Butterworth, Bessel, and Elliptic—for post-upsampling filtering, and evaluate their effects on 16→48 kHz super-resolution performance on the ESC-50 test set.

We observe that both the Chebyshev and Elliptic filters lead to better super-resolution results, as they feature faster attenuation in the stopband. In contrast, Butterworth and Bessel filters exhibit slower roll-off. We hypothesize that this is because the model may struggle to distinguish whether the observed high-frequency attenuation originates from the true characteristics of the data or from the filtering artifacts introduced during preprocessing. In the former case, the model may mistakenly learn to continue the attenuation trend, which can impair its ability to effectively generate high-frequency components.

G Detailed experiment settings

G.1 Training dataset

Sampling Rate	Dataset	Duration	Type
48 kHz	VCTK-train [118]	40h	Speech
	OpenSLR [43]	190h	Speech
	EARS [86]	100h	Speech
	Expresso [75]	20h	Speech
	MusDB18 [82]	10h	Music
	Medleydb [6]	10h	Music
	[InternalMusic]	2000h	Music
	FSD50K [29]	100h	Sound
	[InternalSound]	2000h	Sound
96 kHz	[InternalMusic]	90h	Music
	[InternalSound]	150h	Sound
192 kHz	[InternalMusic]	5h	Music
	[InternalSound]	10h	Sound

Table 15: Overview of training datasets used at different sampling rates, including both public and internal sources.

To support multi-rate audio super-resolution, we curate a diverse set of training datasets spanning speech, music, and sound domains across three sampling rates: 48 kHz, 96 kHz, and 192 kHz. As shown in Table 15, As shown in Table 15. We can also observe that high-resolution data becomes increasingly scarce as the sampling rate increases.

G.2 Model architecture

We adopt the Diffusion Transformer (DiT) architecture as the noise predictor for our bridge model, following the design of Stable Audio Open [28]. During training, we employ the Bridge-gmax schedule, which has been proven effective in previous super-resolution and text-to-speech tasks [54, 14]. The network architecture and training configurations for the three-stage cascade are detailed below.

Any-to-48 kHz Stage

Model:

- Depth: 24 layers
- Attention heads: 24
- Hidden dimension: 1152
- Scaling factor: 0.25
- Training sample length: 245760 samples (5.12s)

Training: Model:

- Batch size: 128
- Optimizer: Adam with $\beta_1 = 0.9$, $\beta_2 = 0.99$
- Learning rate: 1×10^{-5}
- Weight decay: 0
- Training sample rate range: 2 kHz to 32 kHz
- Bridge $g^2_{\text{max}} = 1.0$
- Bridge $g^2_{\text{min}} = 0.001$

48-to-96 kHz Stage* and *96-to-192 kHz Stage

Model:

- Depth: 16 layers
- Attention heads: 16
- Hidden dimension: 1152
- Scaling factor: 1.00
- Training sample length: 245760 samples (5.12s for 96 kHz 2.56s for 192 kHz)

Training:

- Batch size: 128
- Optimizer: Adam with $\beta_1 = 0.9$, $\beta_2 = 0.99$
- Learning rate: 1×10^{-5}
- Weight decay: 0
- Training sample rate range for *48-to-96 kHz*: 32 kHz to 96 kHz
- Training sample rate range for *96-to-192 kHz*: 64 kHz to 192 kHz
- Bridge $g^2_{\text{max}} = 1.0$
- Bridge $g^2_{\text{min}} = 0.001$

G.3 Evaluation metrics

Log-Spectral Distance (LSD) Log-Spectral Distance (LSD) [25] is a widely used metric for evaluating the performance of audio super-resolution methods. Given a signal s and its super-resolved counterpart $\hat{s}$, their Short-Time Fourier Transforms (STFT) are computed as $S = \text{STFT}(s)$ and $\hat{S} = \text{STFT}(\hat{s})$, where both $S, \hat{S} \in \mathbb{R}^{F \times T}$, with F denoting the number of frequency bins and T the number of time frames. LSD is defined as:

$$\text{LSD}(S, \hat{S}) = \frac{1}{T} \sum_{t=1}^{T} \sqrt{\frac{1}{F} \sum_{f=1}^{F} \left[\log_{10} \left(\frac{S(f,t)^2}{\hat{S}(f,t)^2} \right) \right]^2}. \tag{16}$$

Here, $S(f,t)$ and $\hat{S}(f,t)$ denote the spectral magnitude at frequency f and time t for the original and super-resolved signals, respectively. LSD penalizes spectral discrepancies, and lower values indicate better perceptual similarity in the frequency domain. To evaluate performance over a specific frequency band $[f_1, f_2]$, we define a band-limited version of LSD as:

$$\text{LSD}_{[f_1,f_2]}(S, \hat{S}) = \frac{1}{T} \sum_{t=1}^{T} \sqrt{\frac{1}{f_2 - f_1 + 1} \sum_{f=f_1}^{f_2} \left[\log_{10} \left(\frac{S(f,t)^2}{\hat{S}(f,t)^2} \right) \right]^2}. \tag{17}$$

Structural Similarity (SSIM) While LSD is a point-wise spectral metric, Structural Similarity (SSIM) [113] overcomes its limitations by incorporating structural and contextual information. Originally designed for perceptual image quality assessment, SSIM compares local statistics of luminance, contrast, and structure. In the context of audio, we apply SSIM on log-magnitude spectrograms to capture spectral texture fidelity.

The SSIM score between reference S and super-resolved $\hat{S}$ is computed as:

$$\text{SSIM}(S, \hat{S}) = \sum_{k=1}^{K} \left(\frac{(2\mu_{S_k}\mu_{\hat{S}_k} + \epsilon_1)(2\text{Cov}(S_k, \hat{S}_k) + \epsilon_2)}{(\mu^2_{S_k} + \mu^2_{\hat{S}_k} + \epsilon_1)(\sigma^2_{S_k} + \sigma^2_{\hat{S}_k} + \epsilon_2)} \right). \tag{18}$$

Here, S_k and $\hat{S}_k$ denote the k-th local 7×7 block in the spectrograms, K is the total number of blocks, and $\epsilon_1 = 0.01$, $\epsilon_2 = 0.02$ are stability constants following VoiceFixer [24]. Higher SSIM values imply better preservation of spectral structure.

Virtual Quality Objective Listener (ViSQOL) To assess perceptual speech quality in an objective manner, we adopt the Virtual Speech Quality Objective Listener (ViSQOL) [15], a signal-based estimator of Mean Opinion Score (MOS) using spectro-temporal similarity.

ViSQOL (in audio mode) assumes a sampling rate of 48 kHz and outputs MOS-LQO scores ranging from 1 to 4.75, with higher scores indicating better perceptual quality. However, since ViSQOL is designed for speech and only supports certain sample rates (e.g., 16 kHz), we additionally employ SigMOS [87] to evaluate speech quality for 48 kHz scenario.

G.4 Subjective experiment

For the subjective listening test, we evaluate super-resolution performance on generative model outputs from three representative systems: MaskGCT [111], AudioLDM2 [63], and QA-MDT [53], which represent modern speech generation at 24 kHz and music/sound generation at 16 kHz, respectively.

For speech, we randomly select 10 samples from the LibriSpeech-test set [77] as both the reference audio and the text prompt for MaskGCT generation. For music and environmental audio, we sample 10 examples each from the MusicCaps [1] and AudioCaps [40] datasets. To further assess performance on real-world music, we additionally include 10 music segments from the Song-Describer dataset [68]. All audio clips are truncated to 5.12 seconds. For each test case, we present three versions to listeners: the input audio (Generated), the output from AudioSR, and the output from our system (AudioLBM), yielding a total of 40 groups × 3 samples = 120 audio clips for evaluation.

Participants were presented with 40 randomly ordered groups, each containing the three versions described above.They were instructed to rate each sample based on a holistic evaluation of fidelity, clarity, and overall sound quality. Higher scores indicate better perceived quality.

We collected a total of 2400 ratings from 20 participants with diverse backgrounds. We then compute the average ratings across three domains—Speech, Music, and Audio Effects—as shown in Figure 1 in the Introduction. The results show that super-resolution consistently improves the perceptual quality of low-sample-rate audio, confirming the value of upsampling for generative models. Furthermore, our system outperforms AudioSR across all domains, demonstrating the superior audio quality enabled by our approach.

H Related works

H.1 Detailed introduction of audio super-resolution baselines

Nu-Wave. Nu-Wave [49] represents the first attempt at applying diffusion models to waveform-level audio super-resolution. It builds on the WaveNet [106] structure and adopts the DiffWave [44] training paradigm to learn mappings from fixed low-resolution audio to 48 kHz waveform. To support evaluation across various input resolutions, we follow the official implementation[5] and train separate models for 8 kHz, 12 kHz, 16 kHz, and 24 kHz inputs, each for 1.5 million steps.

Nu-Wave 2. Nu-Wave 2 [49] extends Nu-Wave by incorporating two key innovations: Short-Time Fourier Convolution (STFC) and Bandwidth Spectral Feature Transform (BSFT). These additions improve harmonic modeling and allow support for any-to-48 kHz super-resolution. The model also adopts a fast generation scheme using eight predefined non-uniform sampling steps. For fair comparison, we use the pre-trained model made available by the authors[6].

UDM+. UDM+ [120] retains the DiffWave-based architecture but deviates from Nu-Wave in training strategy. It uses unconditional diffusion modeling and injects low-resolution signal guidance at inference through a 50-step uniform sampling process with low-frequency component replacement[67]. Additionally, Manifold Constraint Gradient (MCG) [17] is introduced to better balance low- and high-frequency consistency during generation.

[5] https://github.com/maum-ai/nuwave

[6] https://github.com/maum-ai/nuwave2

Bridge-SR. Bridge-SR [54] adapts the Nu-Wave 2 architecture but introduces a key conceptual shift: instead of sampling from a standard Gaussian prior, the model initializes directly from the low-resolution input. This bridge formulation allows high-quality any-to-48 kHz SR. Furthermore, it employs data normalization strategies and frequency-aware loss functions to improve the final reconstruction quality.

mdctGAN. mdctGAN [96] targets the instability issues of complex-valued neural networks in the STFT domain by operating in the Modified Discrete Cosine Transform (MDCT) domain. Rather than relying on post-vocoders, mdctGAN uses adversarial learning to generate high-fidelity waveform with phase consistency. We evaluate the model using the authors' official checkpoints[7], covering four different input resolutions.

NVSR. NVSR [60] is a neural vocoder-based speech SR system tailored to handle a wide range of upsampling factors and phase reconstruction challenges. It is composed of three main modules: (1) a mel-bandwidth extension network based on ResUNet; (2) a TFGAN-powered [105] neural vocoder; and (3) a post-processing module. We use the official model checkpoint available at [8] for our comparisons.

AP-BWE. AP-BWE [65] is the first bandwidth extension system to model the high-frequency phase explicitly. It leverages a GAN-based framework to jointly predict amplitude and phase spectra using a dual-stream CNN, where the two branches interact to reconstruct full-band audio from narrowband speech. Although it does not support arbitrary-resolution inputs, we evaluate the four fixed-resolution models provided by the authors[9].

AudioSR. AudioSR [62] is a general-purpose diffusion-based super-resolution model capable of handling speech, music, and general audio inputs. It performs super-resolution in the latent space conditioned on low-resolution inputs and uses a two-stage cascade—consisting of a mel-spectrogram VAE and a vocoder—for waveform reconstruction. Benefiting from strong zero-shot generalization, we evaluate the model using the official implementation[10].

FrePainter. Fre-Painter [41] introduces a masked autoencoder (MAE)-based framework for audio super-resolution. It employs an upper-band masking strategy during fine-tuning to simulate low-resolution inputs by masking high-frequency components. Additionally, a mix-ratio masking approach is utilized to enhance robustness across various input sampling rates. The model is pre-trained on large-scale datasets to learn robust speech representations and is fine-tuned jointly with a neural vocoder for waveform reconstruction. We evaluate the model using the official implementation[11].

FlowHigh. FLowHigh [122] proposes an audio super-resolution method based on flow matching, aiming to overcome the sampling inefficiency of traditional diffusion models. It adopts an inpainting-style framework similar to AudioSR but constrains the modeling space to Mel-spectrograms. A lightweight two-layer Transformer [107] is used as the velocity predictor. To further improve sampling efficiency, FLowHigh introduces a data-dependent prior instead of a pure Gaussian initialization, allowing for faster flow-based sampling. We evaluate the model using the official implementation[12].

H.2 Cascaded modeling

H.2.1 Audio domain

Audio waveform are continuous and high-dimensional [64], training a monolithic network to synthesize full-band signals is notoriously unstable and tends to lose high-frequency detail [16]. Recent work therefore favors cascade architectures in which each stage handles a narrower bandwidth or a simpler representation.

[7] https://github.com/neoncloud/mdctGAN

[8] https://github.com/haoheliu/ssr_eval

[9] https://github.com/yxlu-0102/AP-BWE

[10] https://github.com/haoheliu/versatile_audio_super_resolution

[11] https://github.com/FrePainter/code

[12] https://github.com/jjunak-yun/FLowHigh_code

Text/semantic → audio cascades. Early systems such as Jukebox [23] map a text prompt through three VQ-VAE tiers—semantic, acoustic and waveform—before decoding; Moûsai [93] reduces this to two latent-diffusion steps. More recent models, including AudioLM, MusicLM, and MusicFlow [9, 1, 81], first generate semantic tokens, then refine them into acoustic tokens that a vocoder converts to the final audio. InspireMusic [123] pairs a transformer LM with a flow-based super-resolution decoder for long-form, high-resolution music, while YuE [121] uses an LLM-based lyric encoder followed by a lightweight vocoder, forming a two-stage lyric-to-song pipeline.

Super-resolution cascades. On the SR side, models improve fidelity by letting each block focus on a limited frequency band: progressive up-sampling GANs (PU-GAN) [16], coarse-to-fine phase extensions such as MS-BWE [66], and multi-band diffusion that processes sub-bands in parallel [92]. The work most closely related to ours, Noise-to-Music [36], cascades two diffusion models (3.2 kHz $\rightarrow$ 16 kHz) and uses waveform blurring plus stochastic low-pass resampling to perform cascading augmentation.

H.2.2 Vision domain

A similar multi-stage processing trend is evident in the vision domain. Cascade-resolution training progressively feeds the same network with increasingly finer inputs—as seen in models like PixArt-Σ[12] and SANA[116]—enabling a single model to scale seamlessly to generate high-resolution images.

Text-to-Image cascades. Modern text-to-image generation pipelines often employ cascaded architectures, where separate diffusion or autoregressive models operate sequentially at increasing spatial resolutions. Early examples include CDM [35], DALL·E 2 [83], Imagen [90], and Relay-Diffusion [104]. More recent advances, such as CogView 3 [127], Matryoshka Diffusion Model [31], and FMboost [95], incorporate hybrid approaches combining diffusion models or flow models. Pipelines like SDXL [80] enhance the image quality of Stable Diffusion [88] through explicitly incorporating multi-resolution enhancement and refinement stages.

Super-resolution cascades. Super-resolution systems built on diffusion models also adopt cascaded frameworks. Methods such as Inf-DiT [119] and SR3 [91] perform iterative refinement, while Pixel-Space Laplacian Diffusion [2] leverage successive Laplacian pyramid-based enhancements. Together, these works confirm that allocating separate generators to coarse layout and fine detail is crucial for stable training, fast sampling, and high-fidelity ultra-resolution imagery.

I Qualitative results

In this section, we present two subsections of case studies. The first subsection compares the super-resolution performance of bridge models trained in different representation spaces. The second focuses on cherry-picked samples from the demo page of A^2SB [45], which provides a strong baseline for comparison.

I.1 Modeling on different audio spaces

In this section, we compare three bridge models: (1) following Bridge-SR [54], we train a 10.6M-parameter network based on a large WavNet [106] architecture on our dataset; (2) following NVIDIA-NeMo [46][13], we construct a bridge model directly on the STFT spectrogram, treating each time frame as a token and modeling it with a DiT (Diffusion Transformer [4]) architecture of 0.3B, also trained on our dataset; and (3) our proposed *any-to-48 kHz* AudioLBM system. As shown in Figure 12, waveform-based bridge models perform reasonably well on speech data but remain constrained by limited model capacity. When applied to more diverse content such as music and sound effects, these models fail to generalize, resulting in blurred high-frequency components and weak spectral energy. In contrast, STFT-based models can roughly capture the relationship between low and high frequencies; however, the high-frequency details remain overly smooth, and harmonic structures are not well preserved—often leading to visible spectral discontinuities. Our proposed

[13] `https://github.com/NVIDIA/NeMo`

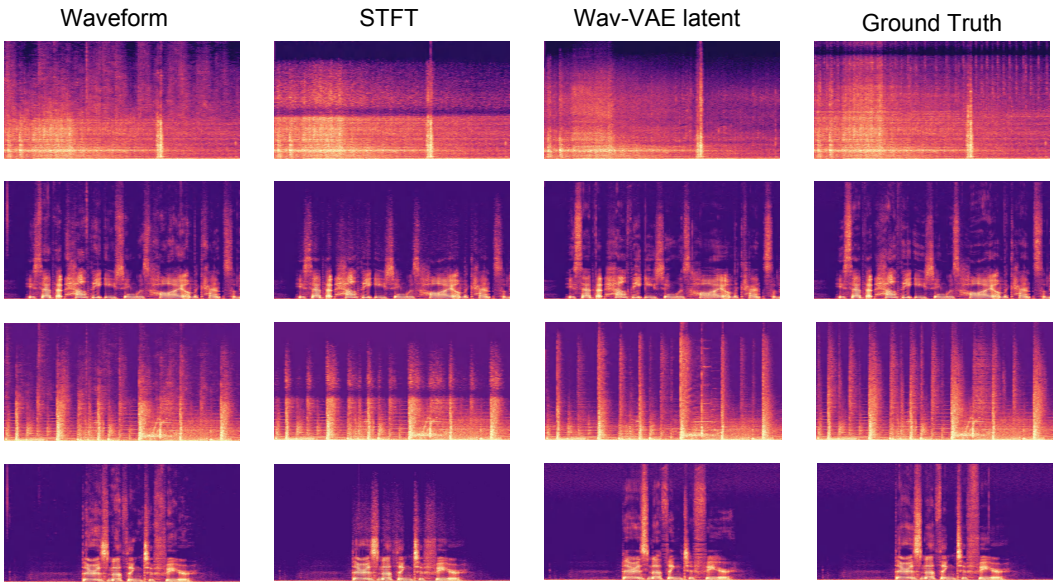

Figure 12: Super-resolution results for bridge models on different modeling spaces.

AudioLBM demonstrates better generalization across all domains, yielding richer spectral detail and more consistent energy distribution across the frequency spectrum.

I.2 Comparison with cherry-picked samples vs. our non-cherry-picked results

In this section, we present a qualitative comparison across five representative super-resolution baselines on the 8 kHz to 48 kHz task: AudioSR [62], A^2SB [45], re-implemented 48 kHz version of Audit, CQTDiff [72] form from A^2SB [41], and our proposed method. As shown in Figure 13,

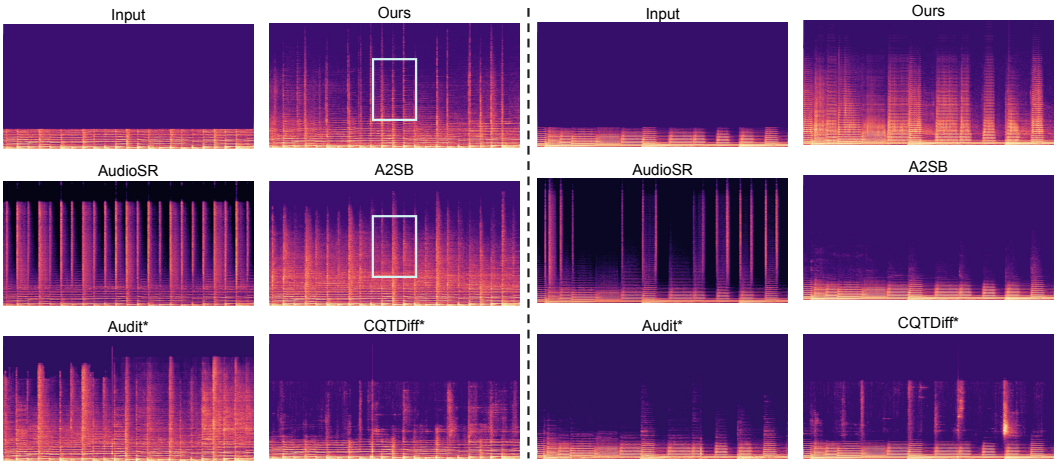

Figure 13: Qualitative comparison from A^2SB's demo page.

our method clearly outperforms other latent-representation-based approaches such as AudioSR and Audit, producing more complete, harmonic-rich, and structurally coherent high-frequency content. In contrast, direct data-space methods like A^2SB and CQTDiff tend to generate overly smooth outputs with blurred high-frequency regions and fail to reconstruct clear harmonic patterns—especially visible in the blue box in the leftmost example.

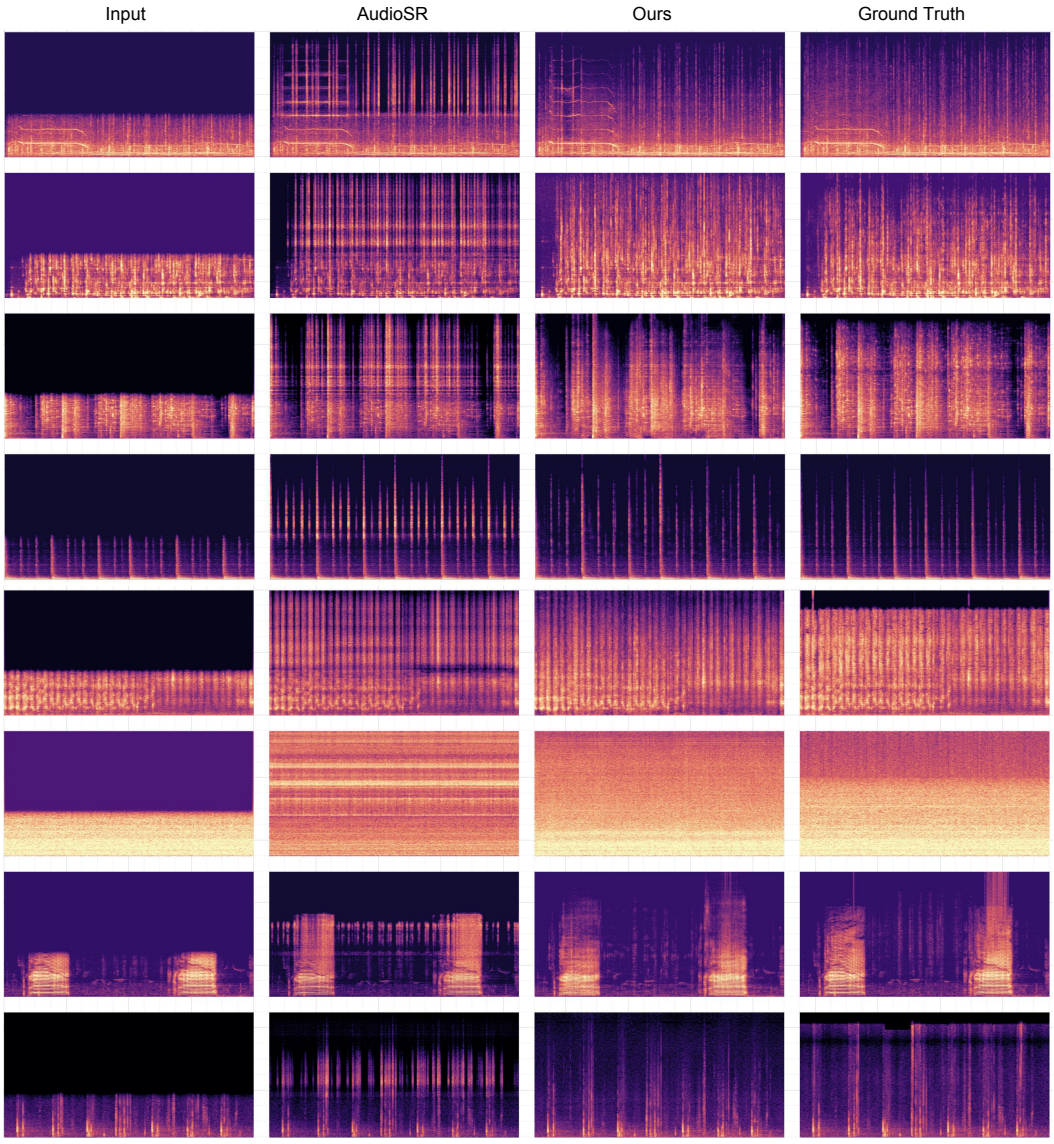

Figure 14: Additional comparison with AudioSR.

I.3 Additional Comparison

As shown in Figure 14, we further compare eight 16 kHz to 48 kHz super-resolution samples between our method and AudioSR. It can be observed that AudioSR, which performs spectral completion in the latent space, often produces excessively strong high-frequency components or misalignments between high- and low-frequency regions. This suggests a limited ability to leverage the low-frequency cues present in the input.

Additionally, due to the use of multi-stage cascading compression network in AudioSR, fine spectral detail is often smoothed out, resulting in overly blurred outputs. In contrast, our method aligns more closely with the ground truth and maintains strong coherence across the full frequency range. Moreover, thanks to the *any-to-any* training paradigm, our system can consistently generate full-bandwidth 48 kHz outputs, whereas AudioSR occasionally fails to fully reconstruct the high-frequency spectrum.