AudioMorphix: Training-free audio editing with diffusion probabilistic models

Anonymous submission

Abstract

Precise and context-aware audio editing remains a foundational yet underexplored problem in audio content creation. While prior methods rely on text prompts or exemplar audio pairs for guidance, they often struggle with spatially precise manipulation in user-specified time-frequency regions and fail to maintain the fidelity of the original audio. In this work, we introduce **AudioMorphix**, a training-free audio editing framework that enables fine-grained manipulation in user-specified spectrogram regions by referencing another recording. Unlike conventional approaches, AudioMorphix allows targeted manipulation of specific time-frequency regions while leaving the remaining audio intact with high fidelity. Inspired by morphing theory, we formulate audio editing as a morphing process in the latent space: during morphing, the raw and reference audio components are blended into a mixed latent; during demorphing, the target latent is disentangled from the mixture to produce the desired edit. AudioMorphix performs conditional denoising in the latent space, optimizing a noised latent guided by raw and reference audio, and further rectifies the diffusion trajectory using task-specific energy functions. To retain subtle characteristics of the original recording, we introduce a key-value cache mechanism into the self-attention layers of the diffusion model. We further propose a benchmark to systematically evaluate diverse audio editing tasks. Experiments show that AudioMorphix achieves high fidelity and precise control across a range of tasks, including addition, removal, replacement, time shifting and stretching, and pitch shifting. Demo and code are available online.

Demo — https://anonymous.4open.science/w/AM-Demo/

Code — https://anonymous.4open.science/r/AudioMorphix

Extended version — https://anonymous.4open.science/r/AudioMorphix/asset/full_paper.pdf

Introduction

Audio acts an essential role in shaping immersive experiences across many forms of content, including movies, audiobooks, and podcasts. In creative workflows, people often need to make selective edits, for instance, replacing a specific sound event or shifting the pitch of an instrument, while leaving the rest of the recording unchanged. Traditional digital signal processing (DSP) tools, such as filtering, equalization, or spectral subtraction, provide low-level manipulations but require extensive manual effort for complex edits. This process is slow and unintuitive, limiting the overall productivity of audio creation.

Neural generative models promise a more content-aware editing experience, yet precise and context-aware control remains an open challenge. Denoising diffusion models have revolutionized generative modeling (Ho, Jain, and Abbeel 2020; Song, Meng, and Ermon 2021a; Liang et al. 2025), particularly in the image domain where spatially targeted edits are well established. Recent advances in audio diffusion (Liu et al. 2023; Ghosal et al. 2023; Majumder et al. 2024) have demonstrated diverse and realistic sound synthesis. However, instruction-guided and region-specific audio editing without unintended alterations elsewhere remains underexplored.

Recent neural audio editors provide only partial solutions to precise and context-aware edits. Text-guided diffusion method (Wang et al. 2023) learned to modify raw audio from text instructions in an end-to-end manner. Exemplar-based approach (Cheng, Li, and Anumanchipalli 2025) transferred sound characteristics from paired audio examples. While both approaches are capable of context-aware edits covering a limited set of tasks, they rely heavily on task-specific training data, which is often hard to collect in practice. Alternatively, inversion-based strategies (Manor and Michaeli 2024; Liu et al. 2023) presented training-free editing by projecting an audio recording back into the noisy latent space and resampling new audio with a modified text prompt. However, these current neural editors lack spatial specificity, often introducing undesirable alterations beyond the target region. Achieving instruction-followed, region-specific editing with high fidelity and minimal unintended changes remains underexplored in neural audio editing.

In this work, we introduce AudioMorphix, a training-free neural audio editor that combines regional control with the semantic generative capabilities of diffusion models. AudioMorphix directly manipulates time-frequency (T-F) spectrograms using a user-defined binary mask and reference audio. We formulate editing as a latent morphing process on the diffusion manifold, where tasks such as addition or removal are framed as controlled trajectory transversals. To achieve precise and context-aware edits, we design energy-based guidance functions to steer diffusion sampling and introduce cross-attention feature substitution to preserve fine details from the reference audio. We further release a new audio editing benchmark dataset that covers three primary

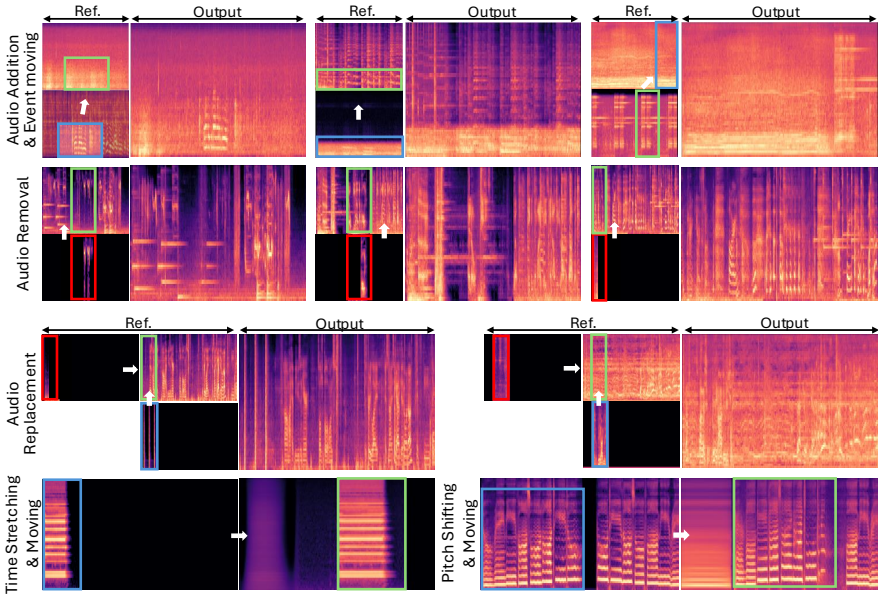


Figure 1: Audio editing tasks of which our AudioMorphix is capable with no training cost. We use green to highlight the editing region on the source audio while blue and red indicate the regions for addition and removal, respectively. The arrows represent the direction of the editing process, showing the flow from the reference audio to the source audio.

tasks, including addition, removal, and replacement, to assess editing performance under various forms of instructions, such as text description, task instruction, and reference audio. As illustrated in Figure 1, AudioMorphix delivers higher localization accuracy and greater editing flexibility than existing editing methods.

Experiment results demonstrate that AudioMorphix outperforms state-of-the-art audio editing models in audio addition, removal, and replacement on both objective and subjective evaluation metrics. Furthermore, AudioMorphix shows strong performance on more editing operations, including time stretching, time shifting, and pitch shifting.

The contributions of this paper are summarized as follows:

- We characterize localized, context-aware audio editing as a spectrogram-masked generation task and further formulate diffusion-based editing as a latent morphing process, enabling structured and flexible control over the editing trajectories in the latent space.
- We propose AudioMorphix, a training-free diffusion-based framework that integrates latent optimization, energy-based guidance and cross-attention substitution for spatially precise and context-aware edits.
- We release a benchmark dataset to evaluate a broad range of neural audio editors under diverse forms of instructions and show that AudioMorphix outperforms current state-of-the-art methods on multiple editing tasks, including addition, removal, and replacement.

Preliminaries

Denoising Diffusion Models

With a predefined forward process $\{q_t\}_{t\in[0,T]}$ that gradually adds noise to a clean sample $\mathbf{x}_0 \in \mathbb{R}^d$, a diffusion model learns to approximate the corresponding reverse process $\{p_t\}_{t\in[0,T]}$ to recover the original sample from its noisy counterpart. At time step t, the noisy sample is constructed as: $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$, where $\bar{\alpha}_t \in (0,1)$ is a predefined variance schedule. See more details in Appendix .

Denoising diffusion implicit models (DDIM) In this work, we applied DDIM (Ho, Jain, and Abbeel 2020) by using a deterministic update:

$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\left(\frac{\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)}{\sqrt{\bar{\alpha}_t}}\right) + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) + \sigma_t\boldsymbol{\epsilon}_t, \quad (1)$$

where $\boldsymbol{\epsilon}_\theta$ is a learned noise predictor and σ_t controls stochasticity. This work adopts the deterministic setting with $\sigma_t = 0$.

Velocity prediction. Salimans and Ho (2022) reformulate the sampling using a velocity vector:

$$\mathbf{v}_\phi = cos(\phi)\boldsymbol{\epsilon} - \sin(\phi)\mathbf{x}_0, \quad (2)$$

where $\phi_t = \arctan(\sigma_t/\alpha_t)$. The DDIM sampling process can be re-written with the trigonometric identities by:

$$\mathbf{z}_{\phi_t-\delta} = \cos(\delta)\mathbf{z}_{\phi_t} - \sin(\delta)\mathbf{v}_\phi(\mathbf{z}_{\phi_t}). \quad (3)$$

This work implements the proposed methods on two representative models: the noise estimator AudioLDM (Liu et al. 2023) and the velocity predictor Tango (Ghosal et al. 2023).

Classifier-free guidance (CFG). CFG is applied to guide the sampling process of diffusion models with an extra condition, such as text description. With CFG, a conditional and an unconditional diffusion model are jointly trained. In the inference stage, the noise prediction can be obtained from conditional and unconditional estimates by

$$\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, \mathbf{y}) = w\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, \mathbf{y}) + (1-w)\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, \varnothing), \quad (4)$$

where w is the guidance scale controlling the strength of the condition signal, and $\varnothing$ denotes the null token.

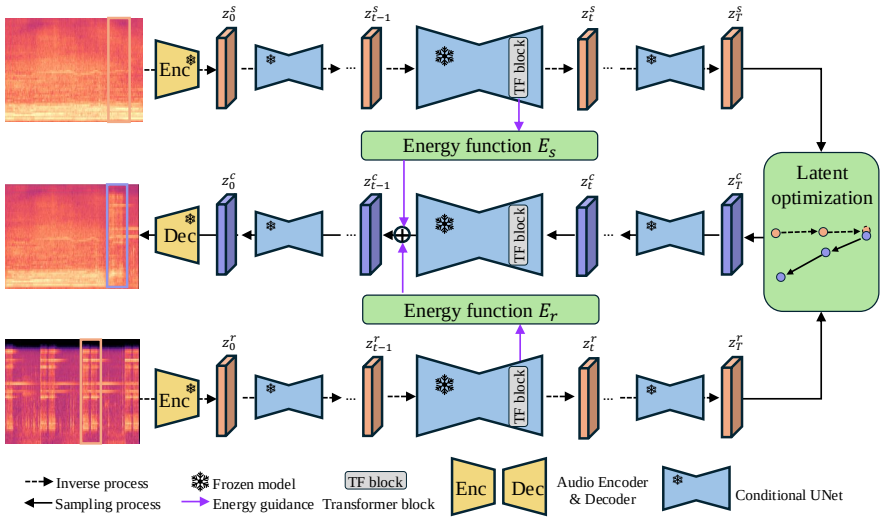


Figure 2: Overview of the proposed AudioMorphix. The AudioMorphix generates the clean latent z_0^c by rectifying the sampling process with latent optimization and energy guidance: After obtaining the noisy latents z_T^s and z_T^r from the spectrograms of the raw and reference audio, along with the corresponding text descriptions s and r, AudioMorphix updates the noisy estimate z_T^c, where c represents the target text description, during the latent morphing process; Throughout the sampling process, AudioMorphix estimates the latent z_{t-1}^c via a frozen latent diffusion model, guided by energy-based functions.

Existing Works on Audio Generation

Audio generation aims to synthesize realistic waveforms from simple priors, such as Gaussian noise, as presented in early work (van den Oord et al. 2016). Recent works have introduced more advanced learning objectives, including autoregressive modeling (Kreuk et al. 2023; Copet et al. 2023a), diffusion methods (Huang et al. 2023; Liu et al. 2023), and agent-based strategies (Liu et al. 2025), to enable high-fidelity audio generation conditioned on diverse signals, such as images and text descriptions. See Appendix for details. However, these methods often struggle with fine-grained control, often requiring multiple rounds of trial and error to achieve user-intended result. To address this limitation, we propose a training-free guidance approach that adapts pretrained audio diffusion networks, for flexible audio editing tasks.

Training-Free Guidance Diffusion

Training-free guidance methods in diffusion models aim to control the output by intervening in the sampling process. In the image domain, DDIM inversion (Mokady et al. 2023) and prompt-to-prompt editing (Hertz et al. 2023) enable image manipulation by changing prompts or modifying cross-attention maps. Noise adjustment techniques (Mokady et al. 2023; Huberman-Spiegelglas, Kulikov, and Michaeli 2024) reuse noise variables from the source image to guide that of the current image in the sampling process. Energy-based functions (Mou et al. 2024; He et al. 2024) direct the sampling toward desired outcomes, while attention substitution (Mou et al. 2024; Chung, Hyun, and Heo 2024) preserves detailed features from a source by replacing key-value components during denoising.

These approaches have shown strong performance in visual editing, where objects are spatially separable. However, directly applying such techniques to audio is ill-suited because *sounds are transparent and often overlap in time and frequency*, making localized editing more challenging. In this work, we handle sound mixtures by operating through a latent morphing process.

Existing Works on Audio Editing

Audio editing has been explored through both training-intensive and training-free approaches (more details in Appendix). Diffusion models such as AudioBox (Vyas et al. 2023), Audit (Wang et al. 2023), and InstructMe (Han et al. 2024) were trained to perform a range of editing tasks. More recent works adapted pre-trained models, e.g., MusicGen for editing (Lin et al. 2024), and image-based personalization methods like DreamBooth (Ruiz et al. 2023) and Textual Inversion (Gal et al. 2023) applied to audio (Plitsis et al. 2024). Despite of their effectiveness in audio editing, they still depend on task-specific finetuning.

More recently, zero-shot audio editing approaches have emerged by combining diffusion inversion and word-level manipulation (Liu et al. 2023; Manor and Michaeli 2024). However, these methods rely heavily on precise text prompts, which limits their flexibility in more general scenarios. Furthermore, they typically edit the entire audio in an end-to-end manner, compromising the fidelity of the original recording through the editing process. This work aims to overcome these limitations by enabling localized edits while reducing the reliance on explicit textual guidance.

Audio Latent Manipulation

Objective

This work introduces AudioMorphix, a neural audio editor that rectifies the diffusion sampling trajectory using a binary mask and a reference audio signal. Compared to previous

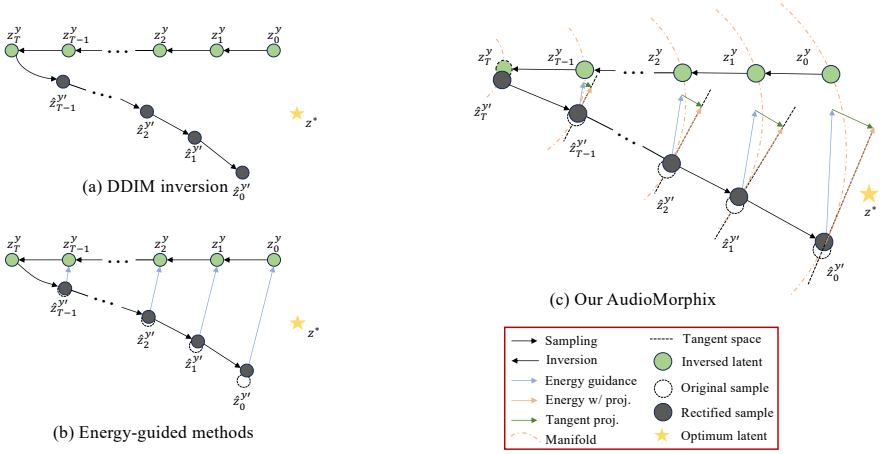


Figure 3: A schematic overview of our AudioMorphix in comparison with DDIM inversion (Mokady et al. 2023) and energy-guided methods (Mou et al. 2024). Let z_t^y be the latent z, correlated with text description y, at the time step t. We omit the process of encoding input audio x into latent z_0^y for simplicity. AudioMorphix refines the sampling processing by updating the noisy latent z_T^y with latent optimization and performing the energy guidance at each time step.

methods (Wang et al. 2023; Liu et al. 2023), it performs zero-shot editing on pre-trained diffusion models, leverages reference audio for context-aware guidance, supports diverse editing tasks (e.g., time-stretching, shifting), and localizes modifications via spectrogram masks while preserving unaffected regions. These capabilities are realized through latent-space optimization and energy-based guidance, allowing precise, semantically meaningful edits without additional training or extensive prompt engineering.

Let $Enc(\cdot)$ be the transformation function mapping an input signal x to latent state z in the diffusion process. While previous methods (Zhang et al. 2024b; Chung, Hyun, and Heo 2024) directly control the trajectory of the generation process, empirically we found:

Property (Latent Spatial Consistency). *The spatial information of x can be inferred from the latent representation z, such that:*

$$\text{sim}(\mathbf{z}_i, \mathbf{z}_j) = \text{sim}(Enc(\mathbf{x}_i), Enc(\mathbf{x}_j)) \propto \text{sim}(\mathbf{x}_i, \mathbf{x}_j). \quad (5)$$

This premise is in line with the finding in (Yang et al. 2024). However, in contrast to visual modalities, manipulating the latent of a sound or a spectrogram is even harder: *Sound tracks are always entangled with each other in a mixture*, resulting in one pixel in a T-F spectrogram-like representation is correlated to more than one sound track.

Manipulate Latent in the Morphing Process

Suppose reference audio $\mathbf{x}_r$ be the interested sound and context audio $\mathbf{x}_c$ be the remaining sound in the mixture. The mixture $\mathbf{x}_m$ is the combination of reference sound $\mathbf{x}_r$ and context $\mathbf{x}_c$, such that $\mathbf{x}_m = \mathbf{x}_r + \mathbf{x}_c$. According to our observation, the latent of the mixture $\mathbf{z}_m$ also correlates with the latent of foreground and background sounds, $\mathbf{z}_r$ and $\mathbf{z}_c$, by: $\mathbf{z}_m \propto \mathbf{z}_r + \mathbf{z}_c$.

In this work, we consider a mixture latent be an interpolation of the reference and context components in the morphing path and reformat three basic audio editing operations from the perspective of the latent morphing process.

Audio addition. Provided a raw audio clip $\mathbf{x}_c$ and a reference audio clip $\mathbf{x}_r$, audio addition is to obtain the interpolation of the two sounds. He et al. (2024) and Yang et al. (2024) argued that latent states are distributed on a manifold, suggesting the infeasibility of linearly combining two latent states. Therefore, we interpolate between the latent state $\mathbf{z}_c$ and $\mathbf{z}_r$ [1] via spherical linear interpolation (SLERP) to obtain a "meaningful" intermediate latent state:

$$\mathbf{z}_m = \frac{\sin((1-\alpha)\omega)}{\sin\omega}\mathbf{z}_c + \frac{\sin(\alpha\omega)}{\sin\omega}\mathbf{z}_r, \quad (6)$$

where ω is defined by $\omega = \arccos(\mathbf{z}_c \cdot \mathbf{z}_r / \|\mathbf{z}_c\| \cdot \|\mathbf{z}_r\|)$. The denoised result $\mathbf{z}_m$ is then updated via the DDIM sampling by using the conditional distribution $p_\theta(\mathbf{x}|\mathbf{y}_c)$.

Audio removal. Audio removal is to remove a sound track $\mathbf{x}_c$ from a mixture $\mathbf{x}_m$ using audio $\tilde{\mathbf{x}}_r$ as reference. In practice, the true reference audio x_r is inaccessible; instead, we assume access to an approximate version $\tilde{\mathbf{x}}_r$, such as a noisy or stylistically similar recording. Since the reference audio $\tilde{\mathbf{x}}_r$ is not unique on the manifold, removing a sound using the reference audio *alone* may not yield a satisfactory editing result. To address this, we estimate an optimal latent state $\hat{\mathbf{z}}_c$, initialized from $\tilde{\mathbf{z}}_c$, to guide the diffusion denoising process. Algorithm 1 outlines the gradient-based optimization procedure used to iteratively refine $\hat{\mathbf{z}}_c$ during the demorphing.

Instead of directly estimating latent state $\tilde{\mathbf{z}}_c$ and $\tilde{\mathbf{z}}_r$, the optimization looks for the optimum direction $\boldsymbol{\epsilon}_c$ and $\boldsymbol{\epsilon}_r$ pointing to $\mathbf{z}_c$ and $\mathbf{z}_r$. Because $\mathbf{z}_c$ and $\mathbf{z}_r$ are distributed on a sphere, we use SLERP function g_S and geodesic distance d_g to calculate the interpolation and similarity, respectively. Assuming $\mathbf{z}_c$ and $\mathbf{z}_r$ are independent from each other, we use the similarity between them as a penalty score to regularize the optimization process. To ensure that the updated latent states remain semantically meaningful, we constrain the optimization direction onto the latent manifold defined by T using the

[1] latent states hereby are referred to as the noise latent at step T in the diffusion process. We ignore the subscription for simplicity.

Algorithm 1: Latent Optimization during the demorphing

Input: $\tilde{\mathbf{z}}_c$, $\tilde{\mathbf{z}}_r$, target $\mathbf{z}_m$, timestep t, step size η, iterations n, weights λ_p, λ_t

Output: Optimized latents $\hat{\mathbf{z}}_c$

Initialize perturbations: $\boldsymbol{\epsilon}_c \leftarrow \mathbf{0}$, $\boldsymbol{\epsilon}_r \leftarrow \mathbf{0}$ (with gradients)
$optimizer \leftarrow \text{SGD}([\boldsymbol{\epsilon}_c, \boldsymbol{\epsilon}_r], \eta)$
for $i = 1$ to n **do**
  $optimizer.zero\_grad()$
  // — Estimate latent states —
  $\hat{\mathbf{z}}_c \leftarrow \tilde{\mathbf{z}}_c + \boldsymbol{\epsilon}_c, \quad \hat{\mathbf{z}}_r \leftarrow \tilde{\mathbf{z}}_r + \boldsymbol{\epsilon}_r$
  $\hat{\mathbf{z}}_m \leftarrow g_S(t, \hat{\mathbf{z}}_c, \hat{\mathbf{z}}_r)$
  // — Compute loss components —
  $\mathcal{L}_{\text{recon}} \leftarrow d_g(\mathbf{z}_m, \hat{\mathbf{z}}_m)$
  $\mathcal{L}_{\text{penalty}} \leftarrow (\sum \hat{\mathbf{z}}_c \cdot \hat{\mathbf{z}}_r)^2$
  $\mathcal{L} \leftarrow \mathcal{L}_{\text{recon}} + \lambda_p \cdot \mathcal{L}_{\text{penalty}}$
  Compute gradients: $\nabla_{\boldsymbol{\epsilon}_c} L$, $\nabla_{\boldsymbol{\epsilon}_r} L$
  if $\lambda_t > 0$ **then**
    // — Project gradients to tangent space —
    $\nabla_{\boldsymbol{\epsilon}_c}\mathcal{L} \leftarrow g_{\tan}(\nabla_{\boldsymbol{\epsilon}_c}\mathcal{L}, T_{\hat{\mathbf{z}}_c}), \quad \nabla_{\boldsymbol{\epsilon}_r}\mathcal{L} \leftarrow g_{\tan}(\nabla_{\boldsymbol{\epsilon}_r}\mathcal{L}, T_{\hat{\mathbf{z}}_r})$
  end if
  Clip gradients: $s_g([\boldsymbol{\epsilon}_c, \boldsymbol{\epsilon}_r])$
  Update: $optimizer.step()$
end for
return $\hat{\mathbf{z}}_c \leftarrow \tilde{\mathbf{z}}_c + \boldsymbol{\epsilon}_c$

mapping $g_{\tan}(\cdot, T)$, following the prior work (He et al. 2024). We empirically set the number of iterations $n = 100$, step size $\eta = 1e^{-4}$ and let $\lambda_p, \lambda_t = 1$.

Audio replacement: Audio replacement is to replace a sound track $\mathbf{x}_{rs}$ from a mixture $\mathbf{x}_m$ with another audio $\mathbf{x}_{rt}$. We decompose the task of audio replacement by removing audio $\mathbf{x}_{rs}$ and adding audio $\mathbf{x}_{rt}$ upon the mixture $\mathbf{x}_m$. We used the same setting as audio addition and removal, respectively.

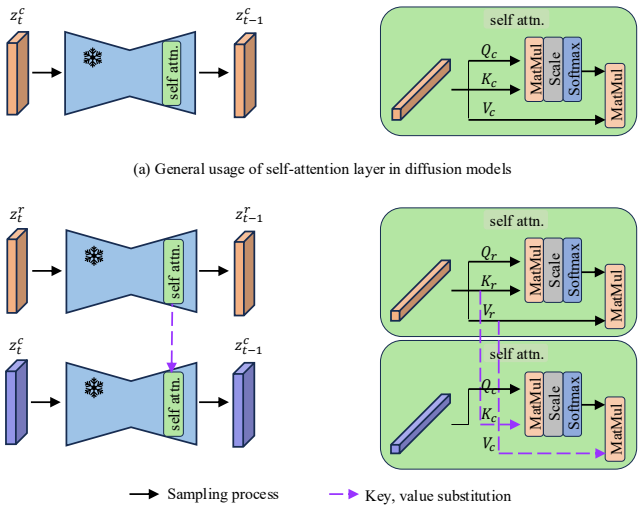
Stepwise Guidance in Sampling Procedure

Overview

This section introduces a stepwise guidance to control the generation procedure using the updated audio latent in Section . Motivated by previous methods (Mou et al. 2024; He et al. 2024), our goal is to decompose a conditional score function $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|c, \mathbf{x}^r)$ into a text-to-audio conditional score function and a differentiable term: $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|c, \mathbf{x}^r) = \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|c) + \nabla_{\mathbf{x}_t} L_t(\mathbf{x}_t; \mathbf{x}^r)$. While there are some works devising energy functions for visual editing, we further improve them by considering latent in the diffusion procedure as T-F representation. Notably, our method is compatible with different prediction objective, such as epsilon (Song, Meng, and Ermon 2021b) and v-prediction (Salimans and Ho 2022).

Guide Audio Editing with Energy Function

In the AudioMorphix, various energy functions are devised as an extra guidance to control the audio generation procedure, mainly focusing on content consistency and contrast between generated audio and reference audio.



(a) General usage of self-attention layer in diffusion models

(b) Our implementation of substituting key and value in self-attention layer

Figure 4: Illustration of adapting self-attention layers to preserve detailed information in the reference latent $\mathbf{z}_r^t$. We cache the key-value components of latent $\mathbf{z}_r^t$ and substitute those of latent $\mathbf{z}_c^t$ during forward process.

The first derivative of an energy function is added to the score obtained from the conditional U-Net $\boldsymbol{\epsilon}_\theta$ for latent update in the sampling process. Suppose $\mathbf{F}_t^c$, $\mathbf{F}_t^r$ are the intermediate features obtained from the conditional U-Net $\boldsymbol{\epsilon}_\theta$ at step t corresponding to the input audio and the reference audio, respectively. Empirically, we collate the intermediate features $\mathbf{F}_{t,l}^c$, $\mathbf{F}_{t,l}^r$ from the l-th self-attention layers of the U-Net decoder Let $\mathbf{m}^c$ and $\mathbf{m}^r$ be the binary masks upon the spectrogram of input audio and reference audio, respectively. The binary masks $\mathbf{m}^c$ and $\mathbf{m}^r$ can constrain the audio editing operating on particular T-F patches. We can measure the consistency between input and reference audio by calculating their cosine similarity over the interested area:

$$\text{sim}(\mathbf{F}_{t,l}^c, \mathbf{m}_c, \mathbf{F}_{t,l}^r, \mathbf{m}_r) = 0.5 \cdot \cos\left(\mathbf{F}_{t,l}^c[\mathbf{m}_c], s_g(\mathbf{F}_{t,l}^r[\mathbf{m}_r])\right) + 0.5, \tag{7}$$

where $s_g(\cdot)$ is the gradient clipping function. Intuitively, we scale the similarity score $\text{sim}(\cdot) \in [0, 1]$ to align with human perception where 0 means the closest distance between two audio. The guidance of consistency term is then defined by:

$$S_{\text{consist}}(\mathbf{F}_t^c, \mathbf{m}_c, \mathbf{F}_t^r, \mathbf{m}_r) = \sum_{l \in L} \frac{1}{1 + 4 \cdot \frac{1}{HW} \sum_{h \in H} \sum_{w \in W} \text{sim}(\mathbf{F}_t^c, \mathbf{m}_c, \mathbf{F}_t^r, \mathbf{m}_r)}, \tag{8}$$

While the contrast concept between two audio can be defined as the reciprocal of the cosine similarity, we argue that in the audio removal use case, the sound track in the reference audio is similar but not the same as that of the input audio. Therefore, the contrast between input and reference audio is measured with the global representation of the input and the reference:

$$S_{\text{contrast}}(\mathbf{F}_t^c, \mathbf{m}_c, \mathbf{F}_t^r, \mathbf{m}_r) = \frac{1}{HW} \sum_{h \in H} \sum_{w \in W} \text{sim}(\mathbf{F}_t^c, \mathbf{m}_c, \mathbf{F}_t^r, \mathbf{m}_r), \tag{9}$$

Notably, the proposed energy functions are capable of generalizing to various prediction objectives, including epsilon and

Table 1: Comparison of various audio editing methods on the AudioSet-E evaluation set, with the highest score highlighted in **bold** and the second highest in underline.

	Addition		Removal		Replacement		Average	
	FAD ↓	KL ↓	FAD ↓	KL ↓	FAD ↓	KL ↓	FAD ↓	KL ↓
DDIM inversion	5.61	1.72	6.24	1.86	8.29	2.05	6.71	1.88
DDPM inversion	19.18	2.27	19.14	2.30	21.25	2.30	19.86	2.29
AUDIT	5.81	3.17	3.47	3.48	5.68	2.81	4.99	3.15
Our method (w/ AudioLDM)	**5.58**	0.83	**2.83**	1.29	**2.67**	2.28	**3.69**	1.47
Our method (w/ Tango)	6.62	**0.57**	6.29	**0.77**	7.27	**0.62**	6.73	**0.65**

Table 2: Subjective scores (%) of various models across different tasks, with the highest score highlighted in **bold** and the second highest in underline.

	Fidelity ↑	Perceptual quality ↑	Consistency ↑	Region specificity ↑	Instruction adherence ↑
Ground truth	**60.84**	**62.27**	**60.25**	56.57	**55.46**
AUDIT	51.79	49.44	47.14	47.10	49.33
DDIM inversion	49.32	50.59	47.63	49.56	50.08
DDPM inversion	46.39	45.96	51.67	49.90	49.21
Our method	56.67	59.21	56.73	**58.76**	52.30

v-prediction, by directly modifying the probability density distribution to rectify the sampling trajectory. In experiments, we let $L = 2, 3$ be the selected self-attention layers of the U-Net decoder.

Energy Guidance for Each Task

Using the consistency measurement S_{consist} and contrast measurement S_{contrast}, we devise various energy-based function:

Audio addition is aim to mix the context audio x^c with the reference audio $\mathbf{x}^r$. $\mathbf{m}_c$ and $\mathbf{m}_r$ are the binary masks of context and reference audio, respectively. Since sound tracks are "transparent", the original sounds in the context audio cannot be replaced with those of reference audio. Therefore, the devised energy function should consider not only the consistency between reference and generated audio, but also the consistency before and after edition. The energy-based guidance can be expressed in the following:

$$\begin{aligned}\boldsymbol{\epsilon}_{add} = w_{content} \cdot S_{\text{consist}}(\mathbf{F}_t, \mathbf{m}_c, \mathbf{F}_t^c, \mathbf{m}_c) \\ + w_{edit} \cdot S_{\text{consist}}(\mathbf{F}_t, \mathbf{m}_c, \mathbf{F}_t^r, \mathbf{m}_r).\end{aligned} \tag{10}$$

Audio removal is to separate a sound track from the input mixture while preserving the rest of sounds. Along with pushing the generated audio away from the reference audio within the interested region of the latent space, we should also maintain the similarity of the global representation between the remaining of the original and synthesized audio:

$$\begin{aligned}\boldsymbol{\epsilon}_{remove} = w_{content} \cdot S_{\text{consist}}(\mathbf{F}_t, \mathbf{m}_c, \mathbf{F}_t^c, \mathbf{m}_c) \\ + w_{edit} \cdot S_{\text{contrast}}(\mathbf{F}_t, \mathbf{m}_c, \mathbf{F}_t^r, \mathbf{m}_r).\end{aligned} \tag{11}$$

Audio replacement is considered as a chain of basic operations. Particularly, we performance removal and addition tasks separately to replace a sound track in the mixture with another one.

Diffusion Procedure with Memory Bank

The combination of latent morphing and energy guidance builds a good posterior in the diffusion sampling process. However, as some works indicate, the gap between generated and reference audio still exists. Following (Mou et al. 2024), we modify the self-attention mechanism in the conditional U-Net. As shown in Figure 4, the key, value of self-attention layers in the decoder are substituted by the original ones obtained from the inversion process. In experiments, we replace the key, value of the second and the third layers with those of the inverted trajectory.

Experiments

Experiment setup

Datasets. To evaluate diverse editing methods, we curated a new dataset *AudioSet-E* based upon the temporally strong labeled part of AudioSet (Gemmeke et al. 2017) for three audio editing tasks, including addition, removal, and replacement. AudioSet-E contains instruction, audio, and pairs of text descriptions as reference for audio editing. Particularly, AudioSet-E contains 1442 samples for audio addition, 1426 samples for audio removal, and 1870 samples for audio replacement. More about data curation in the Appendix We qualitatively evaluated the proposed AudioMorphix on moving, time stretching, and frequency shifting.

Comparison methods. For addition, removal, and replacement tasks, we compared our AudioMorphix against DDIM inversion (Liu et al. 2023), DDPM inversion (Manor and Michaeli 2024), and AUDIT (Wang et al. 2023) on the AudioSet-E. We didn't implement DreamBooth and text inversion methods from (Plitsis et al. 2024) because they are targeted at audio personalization rather than manipulation. In addition to original audio, DDIM and DDPM inversion take a pair of original and target text descriptions as input while for AUDIT an editing instruction is required.

Metrics. We applied Frechet audio distance (FAD) and Kullback–Leibler divergence (KL) to evaluate all audio editing models. FAD measures the fidelity between generated samples and target samples while KL measures the correlation

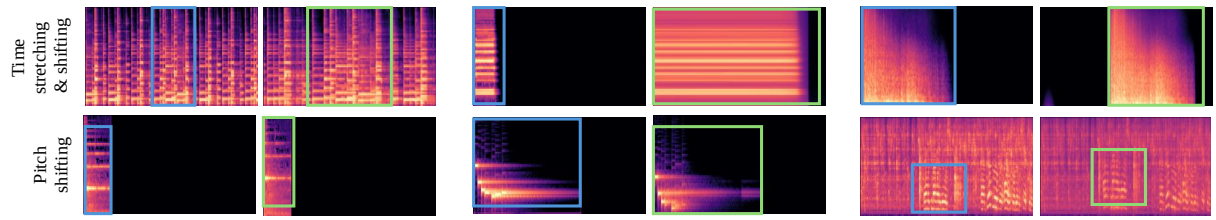


Figure 5: Examples of three audio manipulations: time stretching, time shifting, and pitch shifting.

Table 3: Ablation study on the choices of tangent space projection and text description.

w/ Text	Tan. proj.	Addition		Removal		Replacement	
		FAD ↓	KL ↓	FAD ↓	KL ↓	FAD ↓	KL ↓
		6.46	0.84	**2.46**	**1.03**	6.06	**1.00**
✓		**5.58**	**0.83**	2.83	1.29	**2.67**	2.28
	✓	8.49	3.08	3.08	1.63	8.09	1.06
✓	✓	6.10	1.50	3.38	**1.03**	5.91	1.70

between generated samples and target samples Yuan et al. (2023). We release our evaluation kit [2] to facilitate a fair comparison in the future work. For subjective evaluation, we assess audio editing approaches from five aspects: fidelity, perceptual quality, consistency, region specificity, and instruction adherence. Detailed definitions of these evaluation metrics are provided in the Appendix.

Comparisons

Table 1 compares various audio editing methods on the AudioSet-E evaluation dataset. Our AudioMorphix outperforms the comparison methods across all tasks, particularly excelling in terms of FAD and KL metrics, which indicates better fidelity and distribution matching of the edited images. This suggests that AudioMorphix provides more accurate and realistic image edits compared to DDIM inversion, DDPM inversion, and AUDIT methods. The results are especially notable in the addition and removal tasks, where it shows significant improvements in the KL divergence, indicating a more precise alignment with the target distribution.

Table 2 shows that AudioMorphix achieves the best overall performance, outperforming all comparison methods across addition, removal, and replacement tasks. AudioMorphix slightly outperforms the ground truth in region specificity, probably because AudioMorphix adopts an end-to-end generation approach to avoid perceptual artifacts introduced by signal processing operations, such as clipping and concatenation. Additionally, the proposed model demonstrates strong perceptual coherence and localized detail preservation, making it a well-balanced generative approach. While the ground truth retains the highest fidelity score of 60.84, AudioMorphix highlights the trade-offs between strict fidelity and enhanced perceptual quality, meeting the requirement of audio editing.

Fig. 5 illustrates some examples of three audio manipulation operations, including time stretching, time shifting, and pitch shifting. Compared to traditional DSP methods that modify sounds at the waveform level, the spectrogram-based editing approach offers greater flexibility by enabling manipulation within a specified time-frequency region.

Ablation Study

We evaluated each component of AudioMorphix by ablating text description and tangent space projection in the Table 3. Only the text description leads to a notable improvement in the performance of AudioMorphix, achieving an FAD score of 5.58 and KL score of 0.83 on the addition task and an FAD score of 2.67 on the replacement task. This improvement stems from the fact that a good text description helps identify an optimal latent during the inversion process, directly boosting the effectiveness of AudioMorphix. Conversely, introducing tangent space projection led to performance degradation, especially on the addition and replacement tasks. This is likely because tangent space projection requires a well-designed update scheme compared to direct guidance.

Conclusion

This work introduced AudioMorphix, a training-free framework for precise and context-aware audio editing conditioned on user-defined binary masks and reference audio. By framing editing as a latent morphing process in the diffusion latent space, our method enables region-specific, semantically-meaningful modifications without requiring additional training. AudioMorphix integrates latent optimization, energy-guided trajectory rectification, and key-value substitution in self-attention layers to achieve precise edits while preserving the unaltered portions of the original audio.

Experiments on diverse editing tasks, including addition, removal, and replacement, show that AudioMorphix outperforms previous methods, achieving up to an average FAD score of 3.69 and an average KL score of 0.65 across the primary editing tasks. These results highlight the potential of latent-space optimization for controllable and flexible audio editing. Future work will explore broader forms of guidance and real-time adaptation, moving toward reliable, general-purpose, test-time audio editing tools.

[2] https://anonymous.4open.science/r/TAGE

References

Cheng, K. J.; Li, T.; and Anumanchipalli, G. 2025. Audio Texture Manipulation by Exemplar-Based Analogy. In *2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5.

Chung, J.; Hyun, S.; and Heo, J.-P. 2024. Style Injection in Diffusion: A Training-Free Approach for Adapting Large-Scale Diffusion Models for Style Transfer. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8795–8805.

Copet, J.; Kreuk, F.; Gat, I.; Remez, T.; Kant, D.; Synnaeve, G.; Adi, Y.; and Defossez, A. 2023a. Simple and Controllable Music Generation. In *Advances in Neural Information Processing Systems*, volume 36, 47704–47720. Curran Associates, Inc.

Copet, J.; Kreuk, F.; Gat, I.; Remez, T.; Kant, D.; Synnaeve, G.; Adi, Y.; and Défossez, A. 2023b. Simple and Controllable Music Generation. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Dhariwal, P.; and Nichol, A. 2021. Diffusion Models Beat GANs on Image Synthesis. *Advances in Neural Information Processing Systems*, 11: 8780–8794.

Gal, R.; Alaluf, Y.; Atzmon, Y.; Patashnik, O.; Bermano, A. H.; Chechik, G.; and Cohen-or, D. 2023. An Image is Worth One Word: Personalizing Text-to-Image Generation using Textual Inversion. In *The Eleventh International Conference on Learning Representations*.

Gemmeke, J. F.; Ellis, D. P. W.; Freedman, D.; Jansen, A.; Lawrence, W.; Moore, R. C.; Plakal, M.; and Ritter, M. 2017. Audio Set: An ontology and human-labeled dataset for audio events. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 776–780. New Orleans, LA: IEEE. ISBN 978-1-5090-4117-6.

Ghosal, D.; Majumder, N.; Mehrish, A.; and Poria, S. 2023. Text-to-Audio Generation using Instruction Guided Latent Diffusion Model. In *Proceedings of the 31st ACM International Conference on Multimedia*, MM '23, 3590–3598.

Gui, A.; Gamper, H.; Braun, S.; and Emmanouilidou, D. 2024. Adapting Frechet Audio Distance for Generative Music Evaluation. In *2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1331–1335.

Han, B.; Dai, J.; Hao, W.; He, X.; Guo, D.; Chen, J.; Wang, Y.; Qian, Y.; and Song, X. 2024. InstructME: an instruction guided music edit framework with latent diffusion models. In *2024 Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*.

He, Y.; Murata, N.; Lai, C.-H.; Takida, Y.; Uesaka, T.; Kim, D.; Liao, W.-H.; Mitsufuji, Y.; Kolter, J. Z.; Salakhutdinov, R.; and Ermon, S. 2024. Manifold Preserving Guided Diffusion. In *The Twelfth International Conference on Learning Representations*.

Hertz, A.; Mokady, R.; Tenenbaum, J.; Aberman, K.; Pritch, Y.; and Cohen-or, D. 2023. Prompt-to-Prompt Image Editing with Cross-Attention Control. In *The Eleventh International Conference on Learning Representations*.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*.

Huang, J.; Ren, Y.; Huang, R.; Yang, D.; Ye, Z.; Zhang, C.; Liu, J.; Yin, X.; Ma, Z.; and Zhao, Z. 2023. Make-An-Audio 2: Temporal-Enhanced Text-to-Audio Generation. ArXiv:2305.18474 [cs, eess].

Huberman-Spiegelglas, I.; Kulikov, V.; and Michaeli, T. 2024. An Edit Friendly DDPM Noise Space: Inversion and Manipulations. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 12469–12478.

Kreuk, F.; Synnaeve, G.; Polyak, A.; Singer, U.; Défossez, A.; Copet, J.; Parikh, D.; Taigman, Y.; and Adi, Y. 2023. AudioGen: Textually Guided Audio Generation. In *The Eleventh International Conference on Learning Representations*.

Liang, J.; Liu, X.; Wang, W.; Plumbley, M. D.; Phan, H.; and Benetos, E. 2025. Acoustic Prompt Tuning: Empowering Large Language Models With Audition Capabilities. *IEEE Transactions on Audio, Speech and Language Processing*, 33: 949–961.

Liang, J.; Zhang, H.; Liu, H.; Cao, Y.; Kong, Q.; Liu, X.; Wang, W.; Plumbley, M. D.; Phan, H.; and Benetos, E. 2024. WavCraft: Audio Editing and Generation with Large Language Models. In *ICLR 2024 Workshop on LLM Agents*.

Lin, L.; Xia, G.; Zhang, Y.; and Jiang, J. 2024. Arrange, inpaint, and refine: steerable long-term music audio generation and editing via content-based controls. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*. ISBN 978-1-956792-04-1.

Liu, H.; Chen, Z.; Yuan, Y.; Mei, X.; Liu, X.; Mandic, D.; Wang, W.; and Plumbley, M. D. 2023. AudioLDM: Text-to-Audio Generation with Latent Diffusion Models. In Krause, A.; Brunskill, E.; Cho, K.; Engelhardt, B.; Sabato, S.; and Scarlett, J., eds., *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, 21450–21474. PMLR.

Liu, X.; Zhu, Z.; Liu, H.; Yuan, Y.; Cui, M.; Huang, Q.; Liang, J.; Cao, Y.; Kong, Q.; Plumbley, M. D.; and Wang, W. 2025. WavJourney: Compositional Audio Creation with Large Language Models. *IEEE Transactions on Audio, Speech and Language Processing*.

Lu, C.; Zhou, Y.; Bao, F.; Chen, J.; LI, C.; and Zhu, J. 2022. DPM-Solver: A Fast ODE Solver for Diffusion Probabilistic Model Sampling in Around 10 Steps. In *Advances in Neural Information Processing Systems*, volume 35, 5775–5787.

Majumder, N.; Hung, C.-Y.; Ghosal, D.; Hsu, W.-N.; Mihalcea, R.; and Poria, S. 2024. Tango 2: Aligning Diffusion-based Text-to-Audio Generative Models through Direct Preference Optimization. In *ACM Multimedia 2024*.

Manor, H.; and Michaeli, T. 2024. Zero-Shot Unsupervised and Text-Based Audio Editing Using DDPM Inversion. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, 34603–34629.

Mokady, R.; Hertz, A.; Aberman, K.; Pritch, Y.; and Cohen-Or, D. 2023. Null-text Inversion for Editing Real Images

using Guided Diffusion Models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 6038–6047.

Mou, C.; Wang, X.; Song, J.; Shan, Y.; and Zhang, J. 2024. DragonDiffusion: Enabling Drag-style Manipulation on Diffusion Models. In *The Twelfth International Conference on Learning Representations*.

Plitsis, M.; Kouzelis, T.; Paraskevopoulos, G.; Katsouros, V.; and Panagakis, Y. 2024. Investigating Personalization Methods in Text to Music Generation. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1081–1085.

Ruiz, N.; Li, Y.; Jampani, V.; Pritch, Y.; Rubinstein, M.; and Aberman, K. 2023. DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 22500–22510.

Salimans, T.; and Ho, J. 2022. Progressive Distillation for Fast Sampling of Diffusion Models. In *International Conference on Learning Representations*.

Song, J.; Meng, C.; and Ermon, S. 2021a. Denoising Diffusion Implicit Models. In *International Conference on Learning Representations*.

Song, J.; Meng, C.; and Ermon, S. 2021b. Denoising Diffusion Implicit Models. In *International Conference on Learning Representations*.

van den Oord, A.; Dieleman, S.; Zen, H.; Simonyan, K.; Vinyals, O.; Graves, A.; Kalchbrenner, N.; Senior, A.; and Kavukcuoglu, K. 2016. WaveNet: A Generative Model for Raw Audio. In *9th ISCA Workshop on Speech Synthesis Workshop (SSW 9)*, 125.

Vyas, A.; Shi, B.; Le, M.; Tjandra, A.; Wu, Y.-C.; Guo, B.; Zhang, J.; Zhang, X.; Adkins, R.; Ngan, W.; Wang, J.; Cruz, I.; Akula, B.; Akinyemi, A.; Ellis, B.; Moritz, R.; Yungster, Y.; Rakotoarison, A.; Tan, L.; Summers, C.; Wood, C.; Lane, J.; Williamson, M.; and Hsu, W.-N. 2023. Audiobox: Unified Audio Generation with Natural Language Prompts. arXiv:2312.15821.

Wang, Y.; Ju, Z.; Tan, X.; He, L.; Wu, Z.; Bian, J.; and zhao, s. 2023. AUDIT: Audio Editing by Following Instructions with Latent Diffusion Models. In Oh, A.; Neumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 71340–71357. Curran Associates, Inc.

Xie, Y.; Yao, C.-H.; Voleti, V.; Jiang, H.; and Jampani, V. 2025. SV4D: Dynamic 3D Content Generation with Multi-Frame and Multi-View Consistency. In *The Thirteenth International Conference on Learning Representations*.

Yang, Z.; Yu, Z.; Xu, Z.; Singh, J.; Zhang, J.; Campbell, D.; Tu, P.; and Hartley, R. 2024. IMPUS: Image Morphing with Perceptually-Uniform Sampling Using Diffusion Models. In *The Twelfth International Conference on Learning Representations*.

Yuan, Y.; Liu, H.; Liang, J.; Liu, X.; Plumbley, M. D.; and Wang, W. 2023. Leveraging Pre-Trained AudioLDM for Sound Generation: A Benchmark Study. In *2023 31st European Signal Processing Conference (EUSIPCO)*, 765–769.

Zhang, H.; Chowdhury, S.; Cancino-Chacón, C. E.; Liang, J.; Dixon, S.; and Widmer, G. 2024a. DExter: Learning and Controlling Performance Expression with Diffusion Models. *Applied Sciences*, 14(15).

Zhang, Y.; Huang, N.; Tang, F.; Huang, H.; Ma, C.; Dong, W.; and Xu, C. 2023. Inversion-based Style Transfer with Diffusion Models. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*. Vancouver, Canada. ISBN 9798350301298.

Zhang, Y.; Ikemiya, Y.; Xia, G.; Murata, N.; Martínez-Ramírez, M. A.; Liao, W.-H.; Mitsufuji, Y.; and Dixon, S. 2024b. MusicMagus: Zero-shot text-to-music editing via diffusion models. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*.

Appendix

Preliminary in Diffusion Models

Denoising Diffusion Models

Diffusion models, or score-matching networks, have achieved great process in high-quality generation across various domains, such as image (Dhariwal and Nichol 2021; Zhang et al. 2023), video (Xie et al. 2025), symbolic music (Zhang et al. 2024a) and audio generation (Liu et al. 2023). Let $\mathbf{x} \in \mathrm{X} \subset \mathbb{R}^d$ be a d-dimentional sample in the finite set of X, drawn from the "true but unknown" distribution P, and $\mathbf{y} \in \mathrm{Y}$ be the provided condition, such as text description. Diffusion models generate a new sample by a sequence of invocation of time-dependent score function $\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)$ for noisy data $\mathbf{x}_t$. During training, a noise variable $\boldsymbol{\epsilon}$ is sampled from Gaussian distribution $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. The noisy data $\mathbf{x}_t$ is obtained as a linear combination of the noise variable $\boldsymbol{\epsilon}$ and the clean data $\mathbf{x}_0 \sim P(\mathbf{x})$ at the step t, as $\mathbf{x}_t = \sqrt{\overline{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\overline{\alpha}_t}\boldsymbol{\epsilon}$ where $\overline{\alpha}_t > 0$ is a scaling parameter. This conditional probability distribution can be defined by $q(\mathbf{x}_t|\mathbf{x}_0) := \mathcal{N}(\mathbf{x}_t; \sqrt{\overline{\alpha}_t}\mathbf{x}_0, (1-\overline{\alpha}_t)\boldsymbol{I})$. A diffusion model learns a denoiser $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)$ to parameterize the score function with the loss function

$$\mathbb{E}_{\mathbf{x}_0,t,\boldsymbol{\epsilon}_t\sim\mathcal{N}(0,1)}\left[\|\boldsymbol{\epsilon}_t - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\|_2^2\right], \tag{12}$$

where θ is a set of learnable parameters of the denoiser. In the sampling process, we apply the denoiser $\boldsymbol{\epsilon}_\theta$ to estimate the noise variable $\boldsymbol{\epsilon}_{t-1}$ and substitute it from noisy data $\mathbf{x}_t$ iteratively to get the clean data $\mathbf{x}_0$.

Denoising Diffusion Implicit Models (DDIM)

DDIM was proposed to improve the inference speed by using a deterministic generative process (Ho, Jain, and Abbeel 2020). During inference, DDIM obtains noisy data x_{t-1} at the step t with the following update rule:

$$\begin{aligned}\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\left(\frac{\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)}{\sqrt{\bar{\alpha}_t}}\right) \\ + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) + \sigma_t\boldsymbol{\epsilon}_t,\end{aligned} \tag{13}$$

where on the right side the first term is a prediction of the clean data $\mathbf{x}_0$ using the noisy data $\mathbf{x}_t$ and the denoiser $\boldsymbol{\epsilon}_\theta$, the second term represents the estimated direction pointing to $\mathbf{x}_t$, and the last term denotes a random noise. σ_t is a scaling factor controlling the stochasticity in the sampling process: with $\sigma_t = \sqrt{(1-\bar{\alpha}_{t-1})/(1-\bar{\alpha}_t)}\sqrt{1-\bar{\alpha}_t/\bar{\alpha}_{t-1}}$. DDIM is implemented as DDPM while $\sigma_t = 0$ is interpreted as a deterministic sampling process. It is noteworthy that some works (Song, Meng, and Ermon 2021a; Lu et al. 2022) considered the deterministic sampling process as the discretization of a continuous-time probability flow ODE. This ODE-update rule can be reversed to give a deterministic connection between $\mathbf{x}_0$ and its latent state $\mathbf{x}_t$ (Ho, Jain, and Abbeel 2020), given by

$$\frac{\mathbf{x}_{t+1}}{\sqrt{\beta_{t+1}}} - \frac{\mathbf{x}_t}{\sqrt{\beta_t}} = \left(\sqrt{\frac{1-\beta_{t+1}}{\beta_{t+1}}} - \sqrt{\frac{1-\beta_t}{\beta_t}}\right)\boldsymbol{\epsilon}_\theta^{(t)}(\mathbf{x}_t). \tag{14}$$

Velocity Prediction

For inference effiency, Salimans and Salimans and Ho (2022) defined velocity $\mathbf{v}$ as the combination of a clean sample $\mathbf{x}_0$ and noise component $\boldsymbol{\epsilon}$:

$$\mathbf{v}_\phi = cos(\phi)\boldsymbol{\epsilon} - \sin(\phi)\mathbf{x}_0, \tag{15}$$

where $\phi_t = \arctan(\sigma_t/\alpha_t)$. Therefore, the DDIM sampling process can be re-wroten by:

$$\mathbf{z}_{\phi_t} = \cos(\phi_t)\mathbf{x}_0\boldsymbol{\epsilon}(\mathbf{z}_{\phi_t}) + \sin(\phi_t)\hat{\boldsymbol{\epsilon}}(\mathbf{z}_{\phi_t}), \tag{16}$$

where $\hat{\boldsymbol{\epsilon}}(\mathbf{z}_\phi) = (\mathbf{z}_\phi - \cos(\phi)\hat{\mathbf{x}}_\theta(\mathbf{z}_\phi))\sin(\phi)$. By applying the trigonometric identities, the update step can be written as

$$\mathbf{z}_{\phi_t-\delta} = \cos(\delta)\mathbf{z}_{\phi_t} - \sin(\delta)\mathbf{v}_\phi(\mathbf{z}_{\phi_t}). \tag{17}$$

Further Discussion on Audio Generation

Audio generation aims to synthesize realistic waveforms from simple prior distributions, such as Gaussian noise, as pioneered by early works like WaveNet (van den Oord et al. 2016). In recent years, this field has seen rapid advancements driven by the development of more sophisticated generative techniques. Autoregressive approaches (Kreuk et al. 2023; Copet et al. 2023a) generate each audio token conditioning each prediction on previously generated outputs. Diffusion-based methods (Huang et al. 2023; Liu et al. 2023) model the generation process as a sequence of denoising steps, gradually transforming a noise-corrupted latent into structured audio. More recently, LLM-based agents (Liu et al. 2025) have been introduced, leveraging the reasoning capabilities of large language models alongside a toolbox of deep neural networks to orchestrate complex generation workflows. These approaches have demonstrated the ability to produce high-fidelity audio conditioned on a wide range of modalities, including text, images, and video.

Despite these advances, there still remains a large margin to achieve fine-grained control over the generated audio content. Most existing methods (Copet et al. 2023a; Liu et al. 2023; **?**) rely on extensive prompt engineering to align the output with specific user intentions. Such trial-and-error approaches are often time-consuming and inefficient, especially when precise and context-aware edits are needed, such as modifying a specific segment of audio or altering stylistic attributes while preserving semantic content. To overcome these limitations, we introduce a training-free guidance framework that enables targeted, interpretable, and flexible audio editing across a variety of editing tasks.

Further Discussion on Training-Free Guidance Diffusion

Recently training-free guidance diffusion methods are introduced to control the generated output by interfering with the sampling process of diffusion models. In the community of images, DDIM inversion was proposed to manipulate an image by inverting it with the corresponding prompt and re-generating a new one conditioned on a reference prompt (Mokady et al. 2023). Prompt-to-prompt framework (Hertz et al. 2023) was introduced to edit images by adjusting text description and attention map in cross-attention

layers. (Mokady et al. 2023) and (Huberman-Spiegelglas, Kulikov, and Michaeli 2024) preserved in the diffusion process of source images the noise variable which is then used to adjust the noise variable of current images. (Mou et al. 2024) and (He et al. 2024) designed energy functions as an extra guidance on the top of noise estimation to control the sampling process of current images. In addition, some works (Mou et al. 2024; Chung, Hyun, and Heo 2024) attempted to preserve the detailed information in source images by substituting the key, value vectors of the current sampling process with those of the source diffusion process. Despite the leap made in the image domain, there remains a non-trivial issue underlying in the community of audio: *sounds are transparent and always overlap with each other*. In this work, we are studying manipulating a sound track from sound mixtures while maintaining the rest of sound tracks in the audio.

Further Discussion on Audio Editing

A straightforward approach for audio editing is to train a controllable audio generative model capable of taking extra conditions as guidance. AudioBox, a flow-matching model conditioned on both text and audio prompts, was proposed to create the audio content by masking and audio infilling (Vyas et al. 2023). Wang et al. (2023) and Han et al. (2024) trained dedicated diffusion models for various audio editing tasks, such as addition, removal, replacement, and remixing. While these methods can be used for audio editing, large-scale training is required for a satisfying result, which could be impractical in some scenarios.

Some recent works focused on fine-tuning off-the-shelf models for audio editing (Wang et al. 2023). Lin et al. (2024) finetuned MusicGen (Copet et al. 2023b) on multiple music editing tasks by introducing extra signals as guidance. Plitsis et al. (2024) investigated several image editing methods, such as DreamBooth (Ruiz et al. 2023) and Textual inversion (Gal et al. 2023), for audio personalization. Despite the minimal training cost, they still need to tune the model on task-specific datasets.

Zero-shot audio editing tasks were introduced by inverting the diffusion process. Liu et al. (2023) firstly demonstrated the potential of text-to-audio diffusion models for editing tasks using DDIM inversion. More recently, Manor and Michaeli (2024) applied an edit-friendly DDPM latent space to edit the audio content by word swapping. However, these methods rely on precise text descriptions for transcription, which limits their applicability in certain editing scenarios.

Dataset Curation

We curated a new dataset to evaluate various audio editing tasks, including addition, removal, and replacement, based on the temporally strong subset of the AudioSet dataset (AudioSet-SL) (Gemmeke et al. 2017). Utilizing the timestamps of sound events in AudioSet-SL, we mixed 2-3 audio tracks together with or without the selected sound events. A separate dataset was created for each task as described below:

Audio addition. We randomly selected a sound event from two audio samples in the database and created two mixtures: one with and one without the selected sound event. The mixture without the selected sound event was used as the raw audio, and the mixture with the event was used as the target audio. Additionally, we used the isolated sound event as the reference audio. For text descriptions, we used a bag of sound event categories from AudioSet-SL, filling predefined templates with the selected sound event's name as the instruction.

Audio removal. The curation of the audio removal dataset follows a similar process to the audio addition task. However, the mixture with the selected sound events in the audio removal was used as the raw audio, and the mixture without those events served as the target audio. To increase the difficulty of the audio-driven editing task, we randomly sampled 1-second clips from the selected events and discarded the remaining portions during preprocessing.

Audio replacement. We randomly selected three audio recordings, labeled A, B, and C, from AudioSet-SL. We ensured that A and B contained overlapping sound events from different categories. For the raw audio, we mixed audio C and the overlapped region from audio A, and for the target audio, we blended audio C and the same region from audio B. Recordings from A and B were used as the reference audio. For text descriptions, we used combinations of sound events from the two tracks (A and B), filling predefined templates with the relevant sound events as the editing instruction.

The resulting dataset, AudioSet-E, contains 1,442 samples for audio addition, 1,426 samples for audio removal, and 1,870 samples for audio replacement. Compared to previous audio editing datasets (Gui et al. 2024; Liang et al. 2024), AudioSet-E provides a more diverse platform to evaluate the quality of generated audio across multiple editing tasks.

Implement Details

We used a single NVIDIA A100 for evaluation. For a fair comparison, our AudioMorphix was provided with no masking information same as the other editing methods. We set the guidance scale to 1 for AudioLDM and 1.2 for Tango. For our AudioMorphix and ddim inversion, we set the number of inference steps as 50 while implementing DDPM inversion with 200 steps.

Subjective Evaluation Setup

Our subjective evaluations were carried out using Amazon Mechanical Turk [3]. We provided raters with task descriptions and detailed instructions to ensure a consistent evaluation process. Each audio sample was assessed by a minimum of 20 different raters. The final score for each system was calculated by averaging scores across all raters and audio samples.

Further Explanation on Latent Optimization

Algorithm 1 performs latent optimization for a removal task by iteratively refining content and removal latent variables using gradient-based optimization. The process involves calculating a loss based on the model's output and a target, with

[3] https://requester.mturk.com

an optional penalty term to control the interaction between content and the removal variables. Additionally, a tangent loss can be applied to the gradients of the latent variables to improve optimization. The optimization is carried out using Stochastic Gradient Descent (SGD) for a predefined number of iterations, and the final latent variables are returned after the optimization loop.

Qualitative Evaluation

Figure 6 compares our proposed methods against other audio editing methods, including DDIM, DDPM, and AUDIT, over audio addition, removal, and replacement tasks. It can be observed that our AudioMorphix follows the instructions best. Additionally, the AudioMorphix remains the details of non-targeted region in the raw audio, indicating its capacity of high-fidelity audio editing.

Figure 7 indicates the output of the AudioMophix w.r.t. the increase of the source-to-reference ratio, the ratio of source audio to the entire mixture. The goal of this experiment is to assess the impact of spherical linear interpolation (SLERP) operations on the audio addition task. It can be observed that the generated sound smoothly morphed from the source audio to the reference audio. This supports our motivation that a latent of sound mixture can be obtained by morphing between those of two different sound tracks.

Definition of Subjective Evaluation Metrics

To evaluate the performance of audio editing methods, we propose five distinct criteria:

- **Fidelity**. This criterion measures how accurately the model preserves the original content, particularly in unedited portions of the audio. It ensures that no unwanted artifacts or distortions are introduced during the editing process.
- **Perceptual quality**. This evaluates the overall listening experience of the edited audio, focusing on its naturalness, clarity, and freedom from degradation. It aims to ensure that the edited audio sounds cohesive and high-quality from a human listener's perspective.
- **Consistency**. This criterion assesses the smoothness of transitions between differently processed segments, such as between edited and unedited parts of the audio. It ensures that these transitions are seamless and imperceptible, maintaining a coherent and fluid listening experience.
- **Region specificity**. This measures whether the model restricts its editing to the designated regions without unintentionally affecting other areas of the audio. It ensures that edits are precisely confined to the intended portions.
- **Instruction adherence**. This evaluates how well the model interprets and follows the specific editing instructions provided by the user, ensuring that the modifications align with the intended changes.

Subjective Evaluation on Different Tasks

Figure 8 illustrates that AudioMorphix outperforms the top-tier model in region specificity and fidelity across various tasks, demonstrating its superior ability to edit specific regions of the target content while preserving the rest of the audio unchanged. This also suggests that AudioMorphix excels in tasks that require precise and context-aware content manipulation. Furthermore, AudioMorphix shows better consistency compared to Ground Truth on the removal and replacement tasks, highlighting the advantages of its end-to-end generative approach over traditional DSP-based methods. The stable and reliable results provided by AudioMorphix emphasize its potential for high-quality, consistent audio editing.

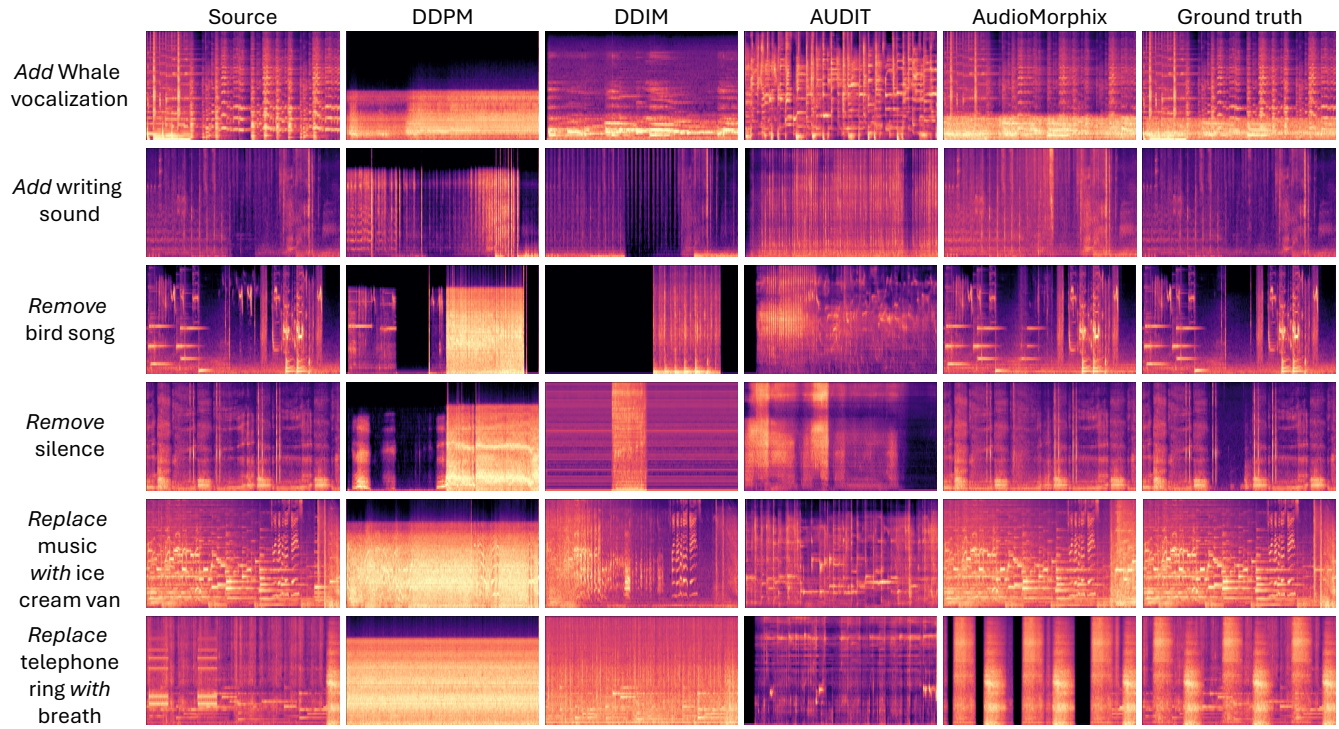


Figure 6: Qualitative evaluation between our AudioMorphix and other audio editing methods.

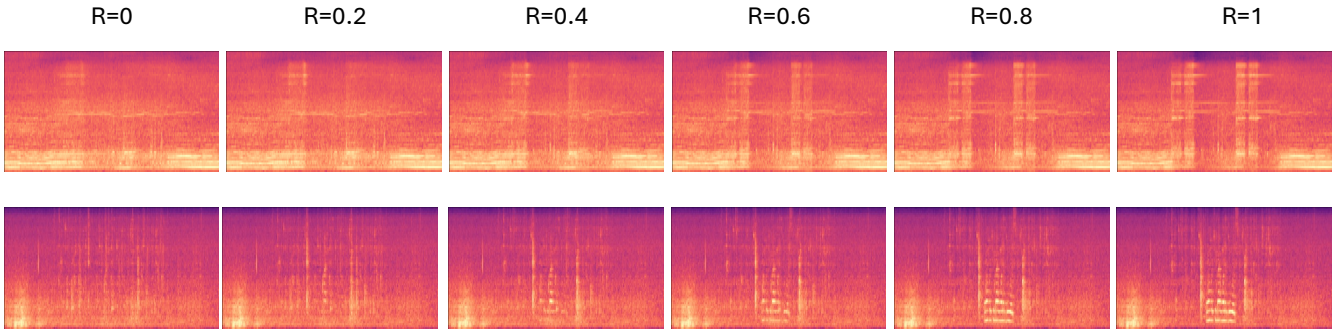


Figure 7: Ablation study on the impact of SLERP in the audio addition task.

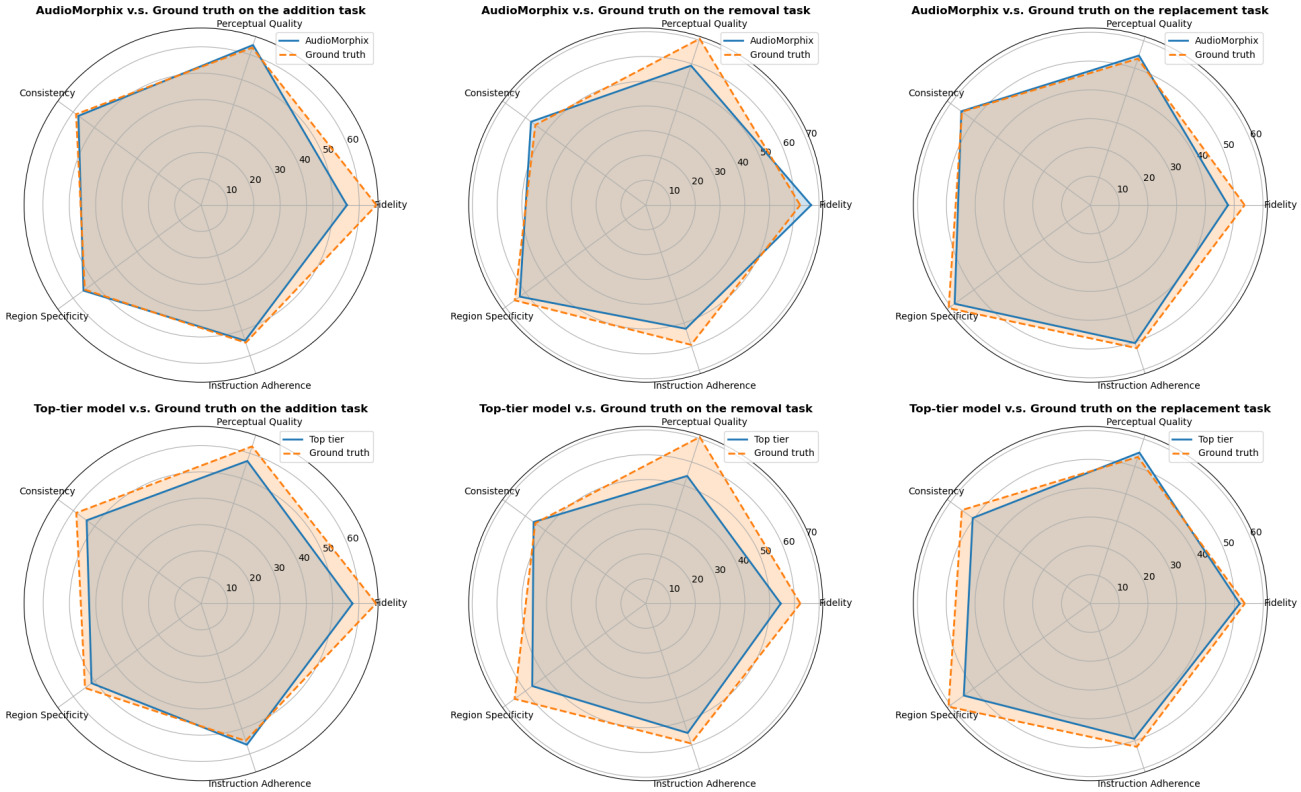


Figure 8: Subjective evaluation across different tasks, with “top tie” denoting the highest value among AUDIT, DDIM inversion, and DDPM inversion.