MUSAudio: Multi-Source Spatial Audio Generation via MLLM-Based Planning and Flow-Matching

Anonymous ACL submission

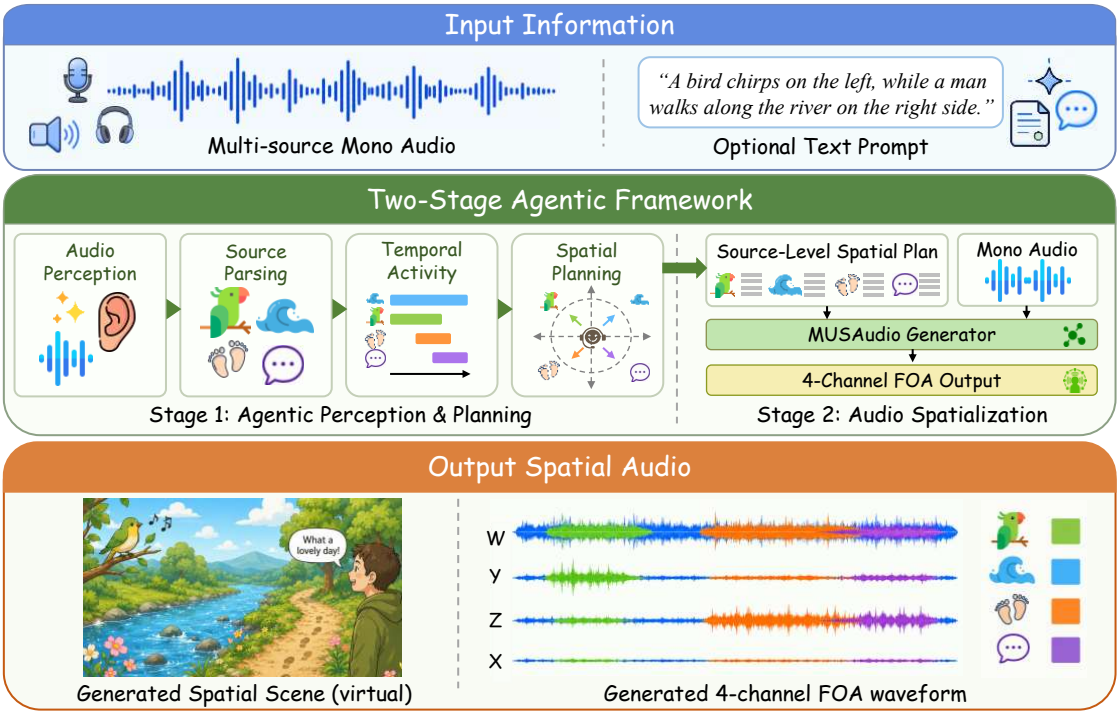

Figure 1: **Overview of MUSAudio.** Given a mono multi-source audio mixture and an optional text prompt, our proposed agentic framework automatically generates immersive 4-channel First-Order Ambisonics (FOA) spatial audio via two stages: (1) *Perception & Planning*, where a tool-augmented MLLM agent accurately parses acoustic events and schedules source-level spatial trajectories; and (2) *Spatial Generation*, where a multi-source latent flow-matching model renders the high-fidelity spatial sound field conditioned on the planned layout.

Abstract

Spatial audio enhances immersion, yet most available audio is recorded in mono, lacking explicit spatial cues. Existing audio spatialization methods typically require manually specified source positions and struggle to model complex acoustic scenes with overlapping, dynamic sources. In this paper, we present MUSAudio, a novel agentic framework for automatic multi-source First-Order Ambisonics (FOA) generation from mono audio. Instead of treating spatialization as a naive signal-to-signal mapping, MUSAudio operates as a two-stage agent that seamlessly integrates audio perception, hierarchical planning, and conditional generation. Given a mono input and an optional text prompt, the agent leverages a multi-modal LLM to parse acoustic event attributes, reason over temporal activities, and synthesize a structured spatiotemporal plan. To support efficient plan-conditioned generation, we introduce a latent flow-matching generator with multi-source attention routing mechanism that aligns source-wise spatial conditions with temporally relevant latent regions, and an FOA-aware VAE that preserves cross-channel spatial coherence through spatially informed supervision. Extensive experiments on a mixed simulated-real dataset show that the agentic planning stage substantially improves source-level spatial grounding, enabling MUSAudio to achieve superior spatial accuracy and perceptual quality. Our code and demos are available at https://musaudio.github.io/.

1 Introduction

Spatial audio plays a crucial role in creating immersive listening experiences for virtual reality, augmented reality, gaming, and live broadcasting (Poeschl et al., 2013; Broderick et al., 2018; Murphy and Neff, 2011). Among various spatial audio formats, First-Order Ambisonics (FOA) has become a widely adopted representation due to its playback format agnosticism and compatibility with diverse rendering systems. However, despite the increasing demand for immersive audio, the majority of existing audio content, including podcasts, video soundtracks, speech recordings, and user generated media, is still produced in mono and lacks explicit spatial information. Take a multi-party podcast as an example: when participants share akin vocal timbres, a downmixed track inevitably blurs the identity-to-voice correspondence. The absence of spatial separation makes it exceedingly difficult for listeners to track the conversational flow and distinguish overlapping speech. This gap has motivated growing interest in automatically converting conventional audio into spatial audio.

Recent years have witnessed substantial progress in audio spatialization (Gao and Grauman, 2019; Leng et al., 2022) and spatial audio generation (Kim et al., 2025; Liu et al., 2025). Traditional digital signal processing pipelines (Zhang et al., 2017; Chen et al., 2020, 2022) typically rely on predefined rendering rules, such as panning, reverberation design, or HRTF (Head Related Transfer Functions) based convolution method, to assign spatial cues to a source signal. While effective in controlled settings, these methods often require manually specified source positions and temporal activity, and are therefore difficult to scale to complex acoustic scenes with multiple overlapping sources and dynamic spatial layouts. More recently, learning based approaches have advanced from neural signal transformation models to autoregressive, diffusion, and flow based generative models, showing promising results in spatial audio synthesis. Nevertheless, existing methods mainly focus on single source or globally conditioned spatialization, and often assume that spatial conditions are explicitly provided. As a result, they remain limited for the more practical setting of automatic multi-source spatialization, where a system must infer what sound sources are present, when they are active, and where they should be placed in space before generating spatially coherent audio.

Converting mono audio into multi-source spatial audio presents two coupled challenges, namely understanding and generation. On the understanding side, the system must parse a mixed audio stream into source-level semantic events, identify their temporal activity, and determine plausible spatial layouts that are consistent with the input content and optional user instructions. This requires a structured reasoning process beyond low level acoustic transformation. On the generation side, the model must synthesize spatial audio that preserves both perceptual quality and spatial correctness, while handling multiple sources with distinct spatial assignments that may change over time.

To address these challenges, we propose MUSAudio, a unified framework for multi-source spatial audio generation that combines MLLM-based spatial planning with flow-matching generation. As illustrated in Figure 1, MUSAudio decomposes the underdetermined mono-to-FOA problem into two stages. First, a planning agent externalizes audio understanding into a structured source-level spatial plan by identifying constituent sources, grounding their active time spans, assigning plausible spatial positions, and verifying the resulting plan. Second, a latent flow-matching generator synthesizes FOA audio conditioned on this interpretable plan, where a multi-source attention routing mechanism aligns source-specific spatial conditions with temporally relevant latent regions. To preserve coherent spatial structure across FOA channels, we further design an FOA-aware VAE with spatially informed supervision, including virtual-direction decoded MRSTFT reconstruction and intensity-direction consistency losses. Finally, we construct a mixed simulated-real spatial audio dataset and benchmark for both training and evaluation. Extensive experiments demonstrate that MUSAudio significantly outperforms existing baselines in generation quality and spatial accuracy.

Our main contributions are as follows:

1. We propose MUSAudio, an agentic mono-to-FOA generation framework that converts ambiguous audio understanding into a structured and verifiable source-level spatial plan for conditional spatial audio synthesis.

2. We develop a latent flow-matching generator with multi-source attention routing and an FOA-aware VAE, enabling source-level spatial conditioning while preserving cross-channel spatial coherence.

3. We build a mixed simulated-real spatial audio dataset and benchmark, and show that MUSAudio outperforms existing baselines in both perceptual quality and spatial accuracy.

2 Related Work

Audio Spatialization. Audio spatialization (Li et al., 2024) aims to generate spatial audio from conventional inputs, often conditioned on text, trajectories, spatial cues, or visual context. Early methods mainly relied on digital signal processing techniques (Zhang et al., 2017; Chen et al., 2020, 2022) and typically required manually specified spatial parameters, which limits their flexibility in complex scenes with multiple sources and dynamic layouts. More recent neural approaches have explored binaural generation conditioned on trajectories (Richard et al., 2021; Leng et al., 2022), video (Gao and Grauman, 2019), or text (Pan et al., 2025). Despite encouraging progress, most existing methods focus on binaural audio or single source spatialization, and often assume that spatial conditions are explicitly provided. Compared with binaural audio, FOA (Zhu et al., 2025) provides a scene level sound field representation that is more suitable for flexible rendering across playback systems. Therefore, our method targets four-channel FOA generation and automatically infers source-level spatial layouts for multi-source scenes.

Audio Diffusion Generative Models. Diffusion based models have achieved strong performance in audio generation. Works such as AudioLDM 2 (Liu et al., 2024), Make An Audio 2 (Huang et al., 2023), and MMAudio (Cheng et al., 2025) demonstrate high quality audio synthesis from text or video based multimodal conditions. In the spatial audio domain, prior works (Liu et al., 2025; Luo et al., 2026) have extended generative modeling to immersive audio synthesis. However, these methods are not designed for automatic spatialization from mono audio input, and they do not explicitly model source-level spatial planning in multi-source settings. In contrast, our method focuses on FOA generation, explicitly models spatial consistency across four channels, and conditions generation on structured source-level spatial plans with multi-source attention routing.

Agent for Audio. Multimodal large language models (MLLM) (Team, 2026; Comanici et al., 2025) have shown strong capabilities in transcription, captioning, event recognition, and audio reasoning. Systems such as AudioAgent (Wang et al., 2024) further use large models to organize audio editing and processing pipelines. Sonic4D (Xie et al., 2026) uses an MLLM to plan sound locations from field-of-view video. Different from these works, we use an MLLM to resolve the planning ambiguity in mono-to-spatial audio conversion. The agent does not generate waveforms; instead, it produces a structured and verifiable source-level plan that specifies source identities, active intervals, and spatial positions, which then serves as the conditioning interface for FOA generation.

3 Method

In this section, we present MUSAudio, an agent for multi-source spatial audio generation from mono audio and text prompts. Section 3.1 formulates the task as converting a mono signal and a user prompt into FOA spatial audio. Section 3.2 describes the agentic workflow, where an MLLM-based planner analyzes the input, produces a structured spatial prompt, and invokes a latent flow-matching generator. Section 3.3 introduces the latent flow-matching generation objective. Sections 3.4 and 3.5 then describe the FOA-aware VAE and the multi-source conditioning mechanism used by the generator.

3.1 Problem Formulation

Given a mono audio $x_{\mathrm{m}} \in \mathbb{R}^{1\times L}$ and a text prompt q, our goal is to generate an FOA signal $\hat{x}_{\mathrm{f}} \in \mathbb{R}^{4\times L}$ with channels $[W, Y, Z, X]$, representing omnidirectional and directional components. We formulate this agentic process as $\hat{x}_{\mathrm{f}} = \mathcal{A}(x_{\mathrm{m}}, q)$.

Since mono-to-FOA conversion is underdetermined, the agent first uses an MLLM-based planner $\mathcal{P}$ to convert (x_{mono}, q) into a structured prompt

$$c = \mathcal{P}(x_{\mathrm{m}}, q) = \{s, \{(d_i, a_i, p_i)\}_{i=1}^{N}\}, \quad (1)$$

where $\mathcal{P}$ is the overall MLLM-based planning module, s is the scene description, N is the number of detected sources, d_i is the description of source i, a_i is its temporal activity annotation, and p_i is its target spatial trajectory or position. The agent then invokes a latent flow-matching generator $\mathcal{G}$:

$$\hat{x}_{\mathrm{f}} = \mathcal{G}(x_{\mathrm{m}}, c) = \mathcal{G}(x_{\mathrm{m}}, \mathcal{P}(x_{\mathrm{m}}, q)). \quad (2)$$

Thus, the planner handles audio-text perception and spatial prompt generation, while the flow generator renders the FOA waveform conditioned on mono audio signal and the structured prompt.

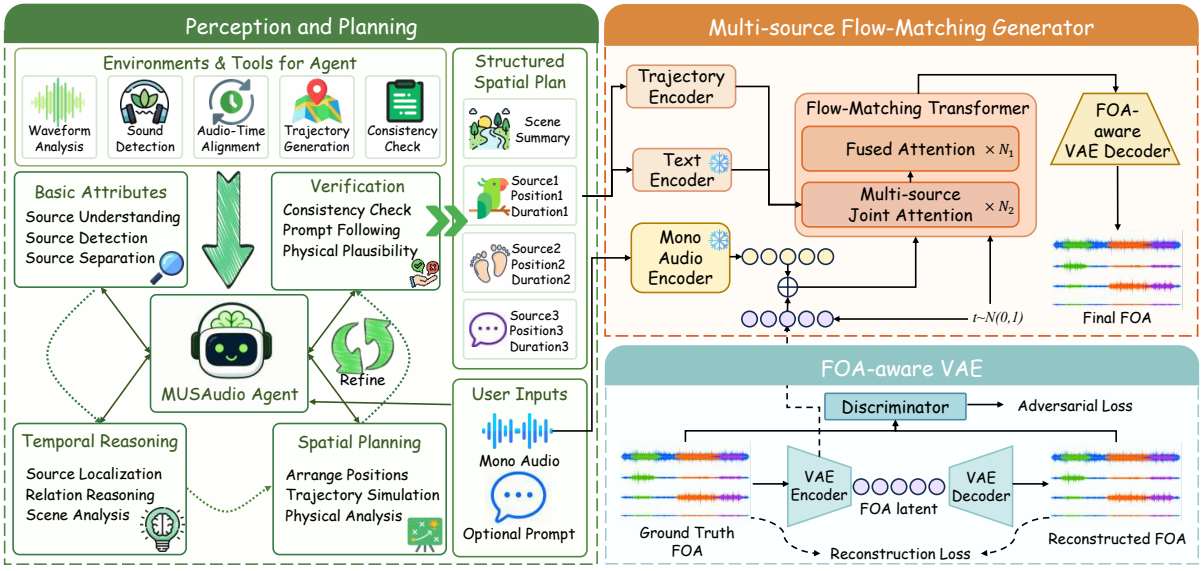

Figure 2: **Overall architecture of MUSAudio.** Our framework operates in two stages: (1) *Perception and Planning* (Left), where a tool-augmented MLLM agent detects acoustic attributes, performs temporal reasoning, and applies spatial planning with a verification loop to yield a *Structured Spatial Plan*. (2) *Spatial Audio Generation*, where a conditional *Multi-source Flow-Matching Generator* (Top Right) synthesizes high-fidelity FOA waveforms within a low-dimensional latent space compressed by an *FOA-aware VAE* (Bottom Right).

3.2 Agent Driven Spatial Audio Generation

Agent Objective. We formulate MUSAudio as an agentic spatial audio generation process. Given mono audio and a text prompt, the agent performs audio-text perception, source-level spatial planning, and flow-matching-based FOA rendering.

Perception-Planning-Generation Workflow. To make the planning process explicit and verifiable, we factorize it into four functional modules: $\mathcal{U}$ identifies "what is sounding" by parsing the acoustic scene and source identities, $\mathcal{T}$ determines "when each source is active" through source-level temporal grounding, $\mathcal{S}$ decides "where each source should be placed" by assigning spatial positions or trajectories, and $\mathcal{V}$ verifies "whether the resulting plan is valid" by enforcing temporal, geometric, and source-separation constraints. This factorization turns the otherwise underdetermined mono-to-FOA mapping into a structured and inspectable planning problem, where intermediate decisions can be verified.

Given x_{m} and q, the agent first calls the audio large language model $\mathcal{U}$ to describe the acoustic scene and identify all audible sources:

$$(s, \{d_i\}_{i=1}^{N}) = \mathcal{U}(x_{\mathrm{m}}, q), \tag{3}$$

where s denotes the scene description and d_i denotes the description of the i-th source. The agent then estimates when each source is active. This step can be implemented by calling either a MLLM or a sound event localization model $\mathcal{T}$:

$$\{a_i\}_{i=1}^{N} = \mathcal{T}(x_{\mathrm{m}}, s, \{d_i\}_{i=1}^{N}), \tag{4}$$

where a_i records the active temporal segments.

Based on the perceived scene, source descriptions, active segments, and user prompt, the agent performs spatial planning with a planner $\mathcal{S}$:

$$\{p_i\}_{i=1}^{N} = \mathcal{S}(q, s, \{(d_i, a_i)\}_{i=1}^{N}). \tag{5}$$

The spatial planner assigns source positions according to user-specified constraints, semantic priors, and spatial audio heuristics, such as separating simultaneously active sources, keeping moving or salient sources perceptually prominent, and placing ambience in a diffuse background region. The preliminary structured plan is then defined as

$$c_0 = \{s, \{(d_i, a_i, p_i)\}_{i=1}^{N}\}. \tag{6}$$

Before generation, the agent applies a lightweight verification module $\mathcal{V}$:

$$c = \mathcal{V}(c_0, x_m), \tag{7}$$

where $\mathcal{V}$ performs temporal boundary clipping, coordinate-range validation, and minimum

angular-separation checks for simultaneously active sources. If any constraint is violated, $\mathcal{V}$ returns diagnostic feedback and triggers refinement of the spatial plan until the verification rules are satisfied. Together, $\mathcal{U}$, $\mathcal{T}$, $\mathcal{S}$, and $\mathcal{V}$ instantiate the planning module $\mathcal{P}$. Finally, the verified structured plan c is passed to the flow-matching generator. Detailed prompts, schema, model calls, and verification rules are provided in Appendix B.1.

Agent Benchmark Construction. To directly evaluate whether the planning agent can produce plausible spatial layouts consistent with the prompt and acoustic content, we constructed an agent-planning benchmark comprising both simulated and real-world recordings. First, we segment the recordings into clips of under 30 seconds and convert the frame-wise DOA annotations into continuous source trajectories. Next, we leverage the metadata alongside spatial hints to generate natural, user-style text prompts. Finally, we employ human annotators to verify the quality and accuracy of the generated data. Detailed construction procedures are provided in Appendix B.2.

3.3 Latent Flow-Matching

The generator operates in the latent space of the FOA-aware VAE. Let E_{f} and D_{f} denote the FOA VAE encoder and decoder. During training, let x_{f} denote the ground-truth FOA signal paired with x_{m}. Its latent representation is $z_{\mathrm{f}} = E_{\mathrm{f}}(x_{\mathrm{f}})$. Following the original formulation (Lipman et al., 2022), we sample a Gaussian noise latent $z_\epsilon \sim \mathcal{N}(0, I)$ and define a linear path from noise to data:

$$z_t = (1-t)z_\epsilon + t z_{\mathrm{f}}, \quad t \in [0,1]. \tag{8}$$

The target velocity along this path is $z_{\mathrm{f}} - z_\epsilon$. We train a conditional vector field v_θ to predict the velocity with the prompt c and mono condition z_{m}:

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}_{z_{\mathrm{f}}, z_\epsilon, t}\left[\|v_\theta(z_t, t, c, z_{\mathrm{m}}) - (z_{\mathrm{f}} - z_\epsilon)\|_2^2\right], \tag{9}$$

where $z_{\mathrm{m}} = E_{\mathrm{m}}(x_{\mathrm{m}})$ is the mono representation produced by the audio encoder E_{m}. During inference, we start from z_ϵ and integrate the learned vector field from $t = 0$ to $t = 1$ to obtain $\hat{z}_{\mathrm{f}}$, which is decoded as $\hat{x}_{\mathrm{f}} = D_{\mathrm{f}}(\hat{z}_{\mathrm{f}})$. Model architecture and inference details are provided in Appendix C.2.

3.4 FOA-aware Audio VAE

Motivation. The flow-matching generator operates in a compact latent space, which demands an audio autoencoder capable of preserving the delicate spatial structure of FOA signals. Because FOA spatial cues are inherently encoded via the cross-channel amplitude and phase relationships among $[W, Y, Z, X]$, a naive channel-wise application of a mono codec is highly suboptimal. Such independent processing not only disrupts spatial consistency but also inflates the latent space, quadrupling the latent dimensionality and introducing a fourfold information redundancy.

Model Architecture. Our autoencoder is adapted from a pretrained mono DAC-VAE. We replace the single-channel input and output convolutions with four-channel variants so that the encoder E_{f} and decoder D_{f} map between an FOA waveform $x_{\mathrm{f}} \in \mathbb{R}^{4\times L}$ and a latent code z_{f}. Different from the original discrete codec (Kumar et al., 2023), we remove the quantization bottleneck and train a continuous VAE by replacing the quantization loss with a KL regularization term. The encoder predicts a posterior $q_\phi(z_{\mathrm{f}} \mid x_{\mathrm{f}})$, and the decoder reconstructs $\hat{x}_{\mathrm{f}} = D_{\mathrm{f}}(z_{\mathrm{f}})$. We also extend the adversarial branch to FOA audio input. The waveform and spectral discriminators in the DAC training pipeline are modified to process the full four-channel tensor, enabling them to penalize both single-channel artifacts and cross-channel spatial inconsistencies.

Training Objective. The VAE objective is organized into reconstruction and adversarial parts:

$$\mathcal{L}_{\mathrm{VAE}} = \mathcal{L}_{\mathrm{rec}} + \mathcal{L}_{\mathrm{adv}}. \tag{10}$$

The reconstruction part contains channel-wise spectral reconstruction, KL regularization, and two FOA spatial consistency losses: $\mathcal{L}_{\mathrm{rec}} = \lambda_{\mathrm{ch}}\mathcal{L}_{\mathrm{ch}} + \lambda_{\mathrm{kl}}\mathcal{L}_{\mathrm{kl}} + \lambda_{\mathrm{virt}}\mathcal{L}_{\mathrm{virt}} + \lambda_{\mathrm{dir}}\mathcal{L}_{\mathrm{dir}}$. For each FOA channel $c \in \{W, Y, Z, X\}$, we compute a multi-resolution STFT loss:

$$\mathcal{L}_{\mathrm{ch}} = \sum_{c\in\{W,Y,Z,X\}} w_c\, L_{\mathrm{STFT}}(x_c, \hat{x}_c), \tag{11}$$

where x_c and $\hat{x}_c$ are the reference and reconstructed signals of channel c, and w_c is the channel weight.

Spatial Reconstruction Losses. The virtual-direction loss decodes FOA signals along K canonical unit directions $\mathbf{d}_k$ and matches the decoded signals in the spectral domain:

$$\mathcal{L}_{\mathrm{virt}} = \frac{1}{K}\sum_{k=1}^{K} L_{\mathrm{STFT}}\left(W + \mathbf{d}_k^\top \mathbf{v}, \hat{W} + \mathbf{d}_k^\top \hat{\mathbf{v}}\right), \tag{12}$$

where $\mathbf{v} = [Y, Z, X]^\top$ and $\hat{\mathbf{v}} = [\hat{Y}, \hat{Z}, \hat{X}]^\top$. The active-intensity loss matches the local direction of acoustic energy flow. For each time-frequency bin, the active-intensity vector is

$$\mathbf{I} = [\mathrm{Re}(WY^*), \mathrm{Re}(WZ^*), \mathrm{Re}(WX^*)], \quad (13)$$

where W, Y, Z, X here denote complex STFT coefficients at the same time-frequency bin, $\mathrm{Re}(\cdot)$ extracts the real part, and $(\cdot)^*$ denotes complex conjugation. Let $\mathbf{n} = \mathbf{I}/(\|\mathbf{I}\|_2 + \epsilon)$ and define $\hat{\mathbf{n}}$ analogously from the reconstructed signal. We use

$$\mathcal{L}_{\mathrm{dir}} = \frac{1}{|\Omega|} \sum_{(f,\ell)\in\Omega} \left(1 - \hat{\mathbf{n}}_{f,\ell}^\top \mathbf{n}_{f,\ell}\right), \quad (14)$$

where (f, ℓ) indexes a time-frequency bin and Ω contains bins whose intensity magnitude is above an energy threshold. Detailed architecture and configurations are provided in Appendix C.1.

3.5 Multi-Source Spatial Conditioning

Model Architecture. As illustrated in Figure 2, the flow generator conditions on text semantics, mono content, and source-level spatial prompts. To robustly extract semantic features, we employ T5Gemma (Zhang et al., 2025) as the text encoder E_{text}. The scene description s and each source description d_i are encoded separately:

$$\mathbf{H}_{\mathrm{scene}} = E_{\mathrm{text}}(s), \quad \mathbf{H}_i^{\mathrm{src}} = E_{\mathrm{text}}(d_i), \quad (15)$$

where $\mathbf{H}_{\mathrm{scene}}$ denotes scene-level text tokens and $\mathbf{H}_i^{\mathrm{src}}$ denotes source-level text tokens for source i.

For acoustic content, we use a mono audio encoder E_{m} initialized from the encoder part of the pretrained mono VAE. Given x_{m}, it produces the frame-aligned mono condition z_{m}. For spatial prompts, each source trajectory or position p_i is interpolated to latent frames and encoded by a spatial position encoder. At each frame, the model obtains a frame-wise spatial condition r_ℓ, where ℓ denotes the latent frame index.

Condition Injection. Let y_ℓ be the noisy FOA latent token at latent frame ℓ. The model injects mono audio embeddings and spatial conditions by adding them to the projected audio token:

$$h_\ell = \phi_{\mathrm{f}}(y_\ell) + \phi_{\mathrm{m}}(z_{\mathrm{m},\ell}) + r_\ell, \quad (16)$$

where h_ℓ is the conditioned audio hidden state, and ϕ_{f} and ϕ_{m} are learnable projections. The same frame-wise condition is also used in audio-side adaptive normalization together with the timestep embedding and global scene embedding. This keeps acoustic content, spatial layout, and flow time aligned at the latent-frame level.

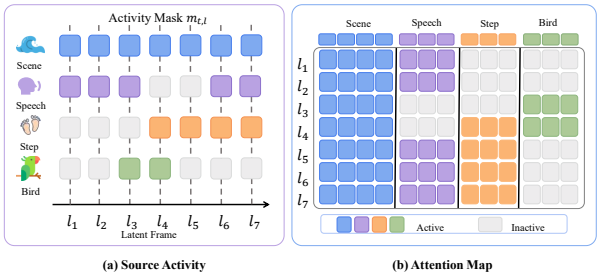

Figure 3: **Multi-source attention routing. (a) Source activity** shows the active temporal segments of individual sources. **(b) Attention map** visualizes how audio latent frames selectively attend to active text tokens.

Multi-Source Attention Routing. We organize the full context tokens as $\mathbf{C} = [\mathbf{H}_{\mathrm{scene}} \mid \mathbf{H}_1^{\mathrm{src}}, \ldots, \mathbf{H}_N^{\mathrm{src}}]$, where the global scene tokens are followed by per-source text tokens. Let $m_{i,\ell}$ be the activity mask derived from a_i, indicating whether source i is active at latent frame ℓ. In each joint block, context tokens attend only to context tokens, while the audio token at frame ℓ attends to scene tokens and active source tokens:

$$\mathcal{A}_\ell = \mathcal{C}_{\mathrm{scene}} \cup \{\mathcal{C}_i^{\mathrm{src}} \mid m_{i,\ell} = 1\}, \quad (17)$$

where $\mathcal{A}_\ell$ is the visible context set for the audio token, $\mathcal{C}_{\mathrm{scene}}$ is the scene-token set, and $\mathcal{C}_i^{\mathrm{src}}$ is the token set of source i. This routing reduces competition among simultaneous sources and preserves the correspondence among the source descriptions, active intervals, and spatial trajectories.

4 Experiments

4.1 Experimental Setup

Datasets. We use simulated and real-world datasets for training and evaluation. **Simulated data**: We construct a large-scale FOA dataset using Pyroomacoustics (Scheibler et al., 2018) with the image source method. Rooms are sampled from 20 acoustic presets spanning diverse environments with varying dimensions and absorption coefficients. Speech utterances and sound effects are convolved with the computed RIRs and rendered to a coincident four-channel FOA microphone array. For single-source scenes, one source is placed per room; for multi-source scenes, 2–3 speakers

engage in turn-taking conversation with optional sound effects, and sources are spatially separated. Both static and dynamic (trajectory-based) source placements are supported. Full simulation details are provided in Appendix A. **Real-world data**: We use the real-world STARSS23 datasets (Shimada et al., 2023) to build MUSBench-R. Full benchmark details are provided in Appendix B.

Baselines. We compare our method against two categories of baselines. (1) **VAE Comparison**: Stable Audio Open (44.1 kHz, Stereo) (Evans et al., 2025), DACVAE (48 kHz, Mono) (Polyak et al., 2024), Flow2GAN (44.1 kHz, Mono vocoder) (Yao et al., 2025), and OmniAudio (44.1 kHz, FOA) (Liu et al., 2025). (2) **Generation Comparison**: As there are no existing baselines capable of spatializing multi-source mono audio from text prompts, we adopt a pipeline approach. We first utilize SAM-Audio (Shi et al., 2025) to separate multi-source audio into single sources, and then apply mono spatialization tools Pyroomacoustics with the explicit trajectories to generate spatial audio.

Metrics. We evaluate our system on two tasks: FOA reconstruction and spatial audio generation. **FOA reconstruction**: We assess waveform fidelity using multi-resolution STFT (MRSTFT) and log-spectral distance (LSD). To evaluate spatial integrity, we finetune a pretrained PSELDNet (Hu et al., 2025) to extract sound events and directions (resampled to 24kHz). PSELDNet details are provided in Appendix D.1. Spatial accuracy is quantified via SELD metrics: error rate (ER), F-score (F), localization error (LE), localization recall (LR), and the comprehensive SELD score. **Spatial audio generation**: We combine objective and subjective evaluations. Objective audio quality is measured using LAION-CLAP (Wu* et al., 2023) similarity and Fréchet Audio Distance (FAD). Spatial accuracy is assessed using PSELDNet-derived metrics. Finally, subjective listening tests evaluate quality (MOS-Q) and spatial accuracy (MOS-P).

Implementation Details. The FOA-aware VAE is fine-tuned from the pretrained mono DACVAE on 8 GPUs (batch size 4/GPU) with generator lr 2×10^{-4} and discriminator lr 1×10^{-4}, for 2.4M steps (8K warmup, cosine annealing) at 48 kHz. The flow-matching generator is trained with batch size 24, lr 5×10^{-5}, and 500 warmup steps. It uses 18 Transformer blocks (6 joint + 12 fused, hidden dim 512, 16 heads). Classifier-free guidance drops scene, source, trajectory, and mono conditions with probabilities 0.1, 0.2, 0.2, 0.2. Inference uses 20 Euler ODE steps with guidance scale 3.0. Further details are in Appendix C.1 and C.2.

4.2 FOA VAE Reconstruction Quality

We evaluate the reconstruction quality of our FOA VAE against baselines to verify its ability to preserve both acoustic fidelity and spatial structure. Because existing models are designed for fewer channels (e.g., mono or stereo), they must process the four FOA channels independently. This channel-wise decoupling severs cross-channel correlations, causing a failure to maintain phase and energy consistency even if single-channel spectra appear reasonable. As shown in Table 1, MUSAudio demonstrates high efficiency through joint sound-field modeling. While channel-wise baselines inflate the latent space to 4×128, MUSAudio utilizes a compact 128-dimensional representation. Despite this significantly lower information bottleneck, it achieves superior spectral reconstruction over OmniAudio and exhibits robust generalization on real data. Baselines like Flow2GAN overfit single-channel features, showing adequate spectral scores on simulation but poor spatial metrics. Conversely, MUSAudio maintains stable spectral recovery and accurate spatial event structures under real-world conditions. Ablation results further validate our design: removing the virtual-direction and active-intensity reconstruction losses (*w/o FOA loss*) leads to a clear degradation in spatial metrics (e.g., SELD and F-score), proving that explicit spatial constraints are essential beyond naive per-channel spectral matching.

4.3 Main Results

As shown in Table 2, MUSAudio yields nearly identical SELD scores and localization errors compared to its ground-truth counterpart (MUSAudio (GT)), firmly validating the accuracy of our MLLM-based spatial planning agent. Furthermore, MUSAudio significantly outperforms the separation-based geometric baseline (SAMPyroom) in multi-source positioning, reducing the localization error from 42.9° to 27.1°. Although SAMPyroom retains marginal advantages in FAD and CLAP due to explicit signal-level separation, it lacks inter-channel consistency in overlapping scenes, whereas our generative approach synthesizes a more coherent sound field. Subjective evaluations mirror these trends: MUSAudio maintains stable audio quality

Table 1: **FOA reconstruction quality and spatial accuracy on MUSBench-S and MUSBench-R.** MRSTFT and LSD evaluate spectral reconstruction; PSELDNet metrics evaluate spatial reconstruction.

Model	Dim	Rate	Reconstruction		Spatial (PSELDNet)				
			MRSTFT↓	LSD↓	ER↓	F↑	LE↓	LR↑	SELD↓
MUSBench-R (real-world recordings)									
Flow2GAN	4×128	86.1	2.6186	12.9451	0.8412	0.0590	90.8421	0.5854	0.6873
Stable Audio Open	2×64	21.5	2.8534	13.9682	0.7353	0.2507	30.4464	0.7182	0.4838
DACVAE	4×128	25.0	3.2851	15.8729	0.3762	0.7047	11.9511	0.7691	0.2422
OmniAudio	64	21.5	3.0279	14.5616	0.8267	0.1199	84.6422	0.6930	0.6210
MUSAudio (w/o FOA loss)	128	25.0	2.1331	10.6868	0.6030	0.4450	31.1290	0.7544	0.3941
MUSAudio (FOA)	128	25.0	2.1471	10.6429	0.5383	0.5038	25.3114	0.7024	0.3767
MUSBench-S (Simulated recordings)									
Flow2GAN	4×128	86.1	2.0820	8.3587	0.8456	0.1318	72.5759	0.7597	0.5893
Stable Audio Open	2×64	21.5	2.2124	10.3675	0.6636	0.3667	24.6420	0.8222	0.4029
DACVAE	4×128	25.0	3.0189	14.6501	0.2006	0.8622	7.7180	0.8900	0.1227
OmniAudio	64	21.5	2.5834	11.9679	0.8702	0.1556	66.0619	0.7399	0.5854
MUSAudio (w/o FOA loss)	128	25.0	2.1932	10.4360	0.3837	0.6736	15.2671	0.8150	0.2449
MUSAudio (FOA)	128	25.0	2.2260	10.3583	0.3639	0.7005	14.988	0.8234	0.2308

Table 2: **Multi-source spatial audio generation comparison on real (MUSBench-R) data.**

Method	Quality		Spatial (PSELDNet)					Subjective	
	FAD↓	CLAP↑	ER↓	F↑	LE↓	LR↑	SELD↓	MOS-Q↑	MOS-P↑
SAMPyroom	0.3261	0.7300	0.776	0.250	42.9	0.424	0.585	3.69 ± 0.18	3.73 ± 0.22
MUSAudio (GT)	0.4077	0.6975	0.661	0.388	25.6	0.618	0.450	3.75 ± 0.21	3.81 ± 0.23
MUSAudio	0.4076	0.6986	0.662	0.373	27.1	0.625	0.454	3.71 ± 0.19	3.78 ± 0.17

(MOS-Q) while markedly improving perceived spatial accuracy (MOS-P) over the heuristic baseline.

4.4 Ablation Studies

Agent Planning Quality We evaluate the planning module against ground-truth source annotations on our constructed benchmark, comparing our multi-stage agentic workflow with a direct one-shot baseline across Gemini2.5-Pro (Comanici et al., 2025) and Qwen3.5-Omni (Team, 2026). As shown in Table 3, the code-based agentic workflow substantially improves angular localization. Evaluation metrics include Source Accuracy (SrcAcc), Distance Mean Absolute Error (Dist MAE), Angular Error (Ang. Err), and the frame ratio with an angular error below 20° (DOA@20°). For Gemini, the agentic verification loop sharply reduces Ang. Err to 7.31° and boosts DOA@20° to 88.33%. A similar performance leap is observed with Qwen-3.5-Omni, where the agent substantially improves SrcAcc (from 24.38% to 84.38%) and lowers the spatial deviation. These results demonstrate that iterative planning and explicit verification are vital for producing precise spatial trajectories.

Table 3: **Agent spatial planning quality.**

Backend	Method	SrcAcc↑	Dist MAE↓	Ang. Err↓	DOA@20°↑
Gemini2.5-Pro	Direct	**98.75**	**0.067**	34.76	57.08
	Agent	97.50	**0.067**	**7.31**	**88.33**
Qwen3.5-Omni	Direct	24.38	7.453	138.46	20.00
	Agent	**84.38**	**1.497**	**30.72**	**76.98**

5 Conclusion

We presented MUSAudio, an agentic framework for automatic multi-source spatial audio generation from mono audio. MUSAudio decomposes the underdetermined mono-to-FOA problem into MLLM-based spatial planning and flow-matching generation. With a multi-source attention routing mechanism and an FOA-aware VAE, our model better aligns source-wise conditions with temporal latent regions while preserving cross-channel spatial coherence. Experiments on simulated and real-world spatial audio data demonstrate that MUSAudio improves both spatial accuracy and perceptual quality over existing baselines. These results suggest that combining structured audio-scene reasoning with spatially aware generative modeling is a promising direction for controllable spatial audio creation.

Limitations

Our work has several limitations. First, mono-to-FOA generation is inherently underdetermined, since a mono mixture cannot uniquely recover the original spatial layout. MUSAudio should therefore be viewed as producing a plausible and controllable spatialization rather than reconstructing the true recording geometry. Second, the system depends on the planning stage. In highly overlapping, noisy, or ambiguous scenes, the MLLM-based planner may make errors in source detection, temporal grounding, or spatial reasoning, which can propagate to the generator. The verification module reduces invalid plans but cannot fully correct semantic mistakes. Third, our training data relies heavily on simulated acoustics, which may not cover the full diversity of real-world rooms, microphone responses, and complex source motions. This may limit generalization to in-the-wild recordings. Finally, the current interface is limited to mono audio and optional text prompts, which may be insufficient for complex spatial control or visually grounded motion. The inference pipeline is also not yet optimized for real-time deployment.

Ethical considerations

MUSAudio aims to support immersive audio creation and controllable spatial audio rendering. However, generated spatial audio should not be misrepresented as an authentic recording of real acoustic events, especially in speech, journalism, legal, or evidential contexts where spatial cues may affect interpretation. Users should ensure proper rights and consent when processing real speech, music, or environmental recordings. The MLLM-based planner may also inherit biases from its training data and may produce culturally or contextually inappropriate spatial assumptions in underspecified cases. We encourage responsible use, clear disclosure of generated content, and documentation of training data, intended use cases, and known failure modes.

References

James Broderick, Jim Duggan, and Sam Redfern. 2018. The importance of spatial audio in modern games and virtual environments. In *2018 IEEE games, entertainment, media conference (GEM)*, pages 1–9. IEEE.

Changan Chen, Unnat Jain, Carl Schissler, Sebastia Vicenc Amengual Gari, Ziad Al-Halah, Vamsi Krishna Ithapu, Philip Robinson, and Kristen Grauman. 2020. Soundspaces: Audio-visual navigation in 3d environments. pages 17–36.

Changan Chen, Carl Schissler, Sanchit Garg, Philip Kobernik, Alexander Clegg, Paul Calamia, Dhruv Batra, Philip Robinson, and Kristen Grauman. 2022. Soundspaces 2.0: A simulation platform for visual-acoustic learning. *Advances in Neural Information Processing Systems*, 35:8896–8911.

Ho Kei Cheng, Masato Ishii, Akio Hayakawa, Takashi Shibuya, Alexander Schwing, and Yuki Mitsufuji. 2025. Mmaudio: Taming multimodal joint training for high-quality video-to-audio synthesis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 28901–28911.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

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

Ruohan Gao and Kristen Grauman. 2019. 2.5 d visual sound. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 324–333.

Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. 2017. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 776–780. IEEE.

Haorui He, Zengqiang Shang, Chaoren Wang, Xuyuan Li, Yicheng Gu, Hua Hua, Liwei Liu, Chen Yang, Jiaqi Li, Peiyang Shi, and 1 others. 2024. Emilia: An extensive, multilingual, and diverse speech dataset for large-scale speech generation. In *2024 IEEE Spoken Language Technology Workshop (SLT)*, pages 885–890. IEEE.

Jinbo Hu, Yin Cao, Ming Wu, Fang Kang, Feiran Yang, Wenwu Wang, Mark D Plumbley, and Jun Yang. 2025. Pseldnets: Pre-trained neural networks on a large-scale synthetic dataset for sound event localization and detection. *IEEE Transactions on Audio, Speech and Language Processing*.

Jiawei Huang, Yi Ren, Rongjie Huang, Dongchao Yang, Zhenhui Ye, Chen Zhang, Jinglin Liu, Xiang Yin, Zejun Ma, and Zhou Zhao. 2023. Make-an-audio 2: Temporal-enhanced text-to-audio generation. *arXiv preprint arXiv:2305.18474*.

Jaeyeon Kim, Heeseung Yun, and Gunhee Kim. 2025. Visage: Video-to-spatial audio generation. *arXiv preprint arXiv:2506.12199*.

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

Yichong Leng, Zehua Chen, Junliang Guo, Haohe Liu, Jiawei Chen, Xu Tan, Danilo Mandic, Lei He, Xiangyang Li, Tao Qin, and 1 others. 2022. Binauralgrad: A two-stage conditional diffusion probabilistic model for binaural audio synthesis. *Advances in Neural Information Processing Systems*, 35:23689–23700.

Zhaojian Li, Bin Zhao, and Yuan Yuan. 2024. Tas: Personalized text-guided audio spatialization. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 9029–9037.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. 2022. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*.

Haohe Liu, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong, Qiao Tian, Yuping Wang, Wenwu Wang, Yuxuan Wang, and Mark D Plumbley. 2024. Audioldm 2: Learning holistic audio generation with self-supervised pretraining. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:2871–2883.

Huadai Liu, Tianyi Luo, Kaicheng Luo, Qikai Jiang, Peiwen Sun, Jialei Wang, Rongjie Huang, Qian Chen, Wen Wang, Xiangtai Li, and 1 others. 2025. Omniaudio: Generating spatial audio from 360-degree video. *arXiv preprint arXiv:2504.14906*.

Ziyu Luo, Lin Chen, Qiang Qu, Xiaoming Chen, and Yiran Shen. 2026. Dynfoa: Generating first-order ambisonics with conditional diffusion for dynamic and acoustically complex 360-degree videos. *arXiv preprint arXiv:2602.06846*.

David Murphy and Flaithrí Neff. 2011. Spatial sound for computer games and virtual reality. In *Game sound technology and player interaction: Concepts and developments*, pages 287–312. IGI Global Scientific Publishing.

Tianrui Pan, Jie Liu, Zewen Huang, Jie Tang, and Gangshan Wu. 2025. In-the-wild audio spatialization with flexible text-guided localization. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1989–2001.

Sandra Poeschl, Konstantin Wall, and Nicola Doering. 2013. Integration of spatial sound in immersive virtual environments an experimental study on effects of spatial sound on presence. In *2013 IEEE virtual reality (VR)*, pages 129–130. IEEE.

Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, and 1 others. 2024. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*.

Alexander Richard, Dejan Markovic, Israel D Gebru, Steven Krenn, Gladstone Alexander Butler, Fernando Torre, and Yaser Sheikh. 2021. Neural synthesis of binaural speech from mono audio. In *International Conference on Learning Representations*.

Robin Scheibler, Eric Bezzam, and Ivan Dokmanić. 2018. Pyroomacoustics: A python package for audio room simulation and array processing algorithms. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 351–355. IEEE.

Bowen Shi, Andros Tjandra, John Hoffman, Helin Wang, Yi-Chiao Wu, Luya Gao, Julius Richter, Matt Le, Apoorv Vyas, Sanyuan Chen, Christoph Feichtenhofer, Piotr Dollár, Wei-Ning Hsu, and Ann Lee. 2025. Sam audio: Segment anything in audio.

Kazuki Shimada, Archontis Politis, Parthasaarathy Sudarsanam, Daniel A Krause, Kengo Uchida, Sharath Adavanne, Aapo Hakala, Yuichiro Koyama, Naoya Takahashi, Shusuke Takahashi, and 1 others. 2023. Starss23: An audio-visual dataset of spatial recordings of real scenes with spatiotemporal annotations of sound events. *Advances in neural information processing systems*, 36:72931–72957.

Qwen Team. 2026. Qwen3. 5-omni technical report. *arXiv preprint arXiv:2604.15804*.

Zixuan Wang, Chi-Keung Tang, and Yu-Wing Tai. 2024. Audio-agent: Leveraging llms for audio generation, editing and composition. *arXiv preprint arXiv:2410.03335*.

Yusong Wu*, Ke Chen*, Tianyu Zhang*, Yuchen Hui*, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. 2023. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*.

Siyi Xie, Hanxin Zhu, Xinyi Chen, Tianyu He, Xin Li, and Zhibo Chen. 2026. Sonic4d: Spatial audio generation for immersive 4d scene exploration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 11087–11095.

Zengwei Yao, Wei Kang, Han Zhu, Liyong Guo, Lingxuan Ye, Fangjun Kuang, Weiji Zhuang, Zhaoqing Li, Zhifeng Han, Long Lin, and 1 others. 2025. Flow2gan: Hybrid flow matching and gan with multi-resolution network for few-step high-fidelity audio generation. *arXiv preprint arXiv:2512.23278*.

Biao Zhang, Fedor Moiseev, Joshua Ainslie, Paul Suganthan, Min Ma, Surya Bhupatiraju, Fede Lebron, Orhan Firat, Armand Joulin, and Zhe Dong. 2025.

Encoder-decoder gemma: Improving the quality-efficiency trade-off via adaptation. *arXiv preprint arXiv:2504.06225*.

Wen Zhang, Parasanga N Samarasinghe, Hanchi Chen, and Thushara D Abhayapala. 2017. Surround by sound: A review of spatial audio recording and reproduction. *Applied Sciences*, 7(5):532.

Zhiyuan Zhu, Yu Zhang, Wenxiang Guo, Changhao Pan, and Zhou Zhao. 2025. Asaudio: A survey of advanced spatial audio research. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 417–442.

Appendix Contents

The Appendix is structured as follows:

- Section A: Details of simulation dataset's construction and the statistics of dataset.
- Section B: Details of MUSBench and the evaluation protocol.
- Section C: Details of the implementation of MUSAudio.
- Section D: Details of the setting of the experiment.

A Data Simulation Details

This appendix provides the full specification of the FOA data simulation pipeline used to construct training data.

Room Acoustics. We define 20 room presets covering diverse acoustic environments, ranging from small treated spaces (recording booths, studios) through domestic rooms (bedrooms, living rooms, kitchens, bathrooms) and workspaces (offices, conference rooms, classrooms) to large public venues (lobbies, auditoriums, concert halls, churches) and special geometries (corridors, stairwells). For each scene, a room type is sampled and its dimensions and broadband absorption coefficient are drawn uniformly from preset-specific ranges. Rooms are simulated as shoebox models using Pyroomacoustics with the image source method (ISM); the maximum reflection order is set between 5 and 15 depending on room size. Table 4 lists representative presets.

Table 4: **Representative room presets** (subset of 20 total).

Room type	Dimensions (m)	Absorption
Recording booth	2–3.5	0.60–0.95
Bedroom	3–5.5	0.30–0.60
Living room	4–7.5	0.20–0.50
Conference room	5–10	0.15–0.35
Lecture hall	10–20	0.05–0.22
Concert hall	18–35	0.03–0.15
Church	10–28	0.02–0.10

Microphone Configuration. The receiver is a coincident first-order Ambisonics (FOA) array, and the rendered signals are encoded in the AmbiX format, i.e., ACN channel ordering with SN3D normalization. Specifically, the four FOA channels are ordered as $[W, Y, Z, X]$, where W denotes the omnidirectional component, and Y, Z, and X denote the left–right, up–down, and front–back directional components, respectively. The microphone position is sampled from 3–4 preset-specific locations per room type (e.g., desk height, standing height, conference table center) at heights between 0.5 and 1.5 m.

Source Placement and Audio. Speech utterances are drawn from the Emilia dataset (He et al., 2024) (English and Mandarin), and sound effects from AudioSet (Gemmeke et al., 2017). All audio is processed at 48 kHz. Source positions are sampled uniformly within the room volume subject to minimum distances of 0.3–0.5 m from walls and 0.5 m from the microphone. Per-source gain is sampled uniformly in $[-3, +3]$ dB.

Static and Dynamic Rendering. For static sources, a single room impulse response (RIR) is computed and convolved with the source audio. For dynamic sources, a quadratic Bézier trajectory is generated between random start and end positions, and the audio is rendered via overlap-add (OLA) processing: the signal is segmented into short windows (0.5 s for single-source, 0.08 s for multi-source), each segment is convolved with the RIR at the interpolated trajectory position, and segments are recombined with Hanning windows. Trajectory waypoints are recorded at 0.25 s intervals for use as training supervision.

Single-Source Scenes. Each scene contains one source (speech or sound effect). Approximately 40% of sources follow dynamic trajectories; the rest are static.

Multi-Source Scenes. Each scene contains 2–3 speakers sampled from the same recording session to ensure speaker consistency, plus 0–2 optional sound effects. Speaker turns follow a round-robin schedule with 15% overlap probability (overlap duration 0.1–0.4 s) and inter-turn gaps of 0.2–1.0 s. Approximately 20% of speech sources and 10% of effect sources are dynamic. Sources are required to maintain a minimum mutual distance of 0.5 m.

Output Format. Each scene produces a four-channel FOA waveform (B-format, 48 kHz) and a JSONL metadata record containing room parameters, per-source positions (or trajectory waypoints), active time segments, textual descriptions, and a natural-language scene caption.

Data Statistics. Currently, the source count distribution within the simulated data comprises

1,928,587 single-source, 159,657 two-source, and 319,982 three-source clips. Among these, static sources account for 68.60% of the data, with an average duration of 12.960 seconds per clip.

B Benchmark Details

B.1 Details of Agent-Driven Spatial Planning

This appendix provides the complete description of the planning agent used in MUSAudio. The main paper only summarizes the planning objective and its interface to the generator due to space constraints.

Input and Output. The planner receives a mono audio clip and, optionally, a user text instruction. The output is a structured spatial plan rather than free-form text. The plan contains a scene-level description and a list of source-level entries. Each source entry specifies the source identity, a natural-language description, active time intervals, and either a static position or a time-varying trajectory. In the experiments of this paper, we mainly use static positions, while the same schema supports dynamic trajectories.

```
{
  "scene": "indoor conversation with
      background music",
  "sources": [
    {
      "source_id": 0,
      "type": "speech",
      "description": "female speaker",
      "active_segments": [[0.3, 4.8],
          [8.1, 12.5]],
      "position": {
        "azimuth_deg": -35.0,
        "elevation_deg": 5.0,
        "distance_m": 1.8
      }
    },
    {
      "source_id": 1,
      "type": "music",
      "description": "background music",
      "active_segments": [[0.0, 12.5]],
      "position": {
        "azimuth_deg": 140.0,
        "elevation_deg": 0.0,
        "distance_m": 3.0
      }
    }
  ]
}
```

Planning Stages. The agent decomposes planning into four stages. First, it performs scene and source parsing, where an audio-capable MLLM or audio tagging model produces source categories, semantic descriptions, and a coarse scene caption. Second, it performs temporal grounding by estimating the active intervals of each source. Third, it performs spatial reasoning. If the user instruction contains explicit spatial constraints, such as "place the speaker on the left", these constraints are used directly. Otherwise, the agent follows perceptual priors: concurrent speech sources should be angularly separated, background ambience or music can be placed behind or around the listener, and semantically salient foreground events should be placed closer than diffuse background events. Fourth, the agent assembles the final plan and passes it to a verifier.

Plan Verification. The verifier is intentionally lightweight and deterministic. It does not replace the MLLM; instead, it prevents invalid or self-contradictory plans from reaching the generator. Specifically, it first checks temporal validity by clipping active segments to the audio duration, removing empty intervals, and merging overlapping intervals from the same source. It then enforces coordinate validity by constraining azimuth, elevation, and distance to valid ranges. To ensure source consistency, the verifier caps the number of source entries by the model's maximum supported source count and fills missing descriptions with generic source labels. Finally, it applies a separation prior for simultaneously active foreground sources by encouraging a minimum angular separation; when two sources are too close, the lower-confidence source is reassigned to a nearby valid direction.

Agent Skill and Prompt Template. As shown in Figure 4, we implement the planner as a reusable mono-to-FOA agent skill. The skill specifies the full procedure used by the agent: audio-scene perception with an audio-capable MLLM, source localization, structured spatial trajectory planning, MLLM-based review, and final invocation of the trained spatial generator. The concrete MLLM backend can be replaced by Gemini, Qwen-Omni, or another audio-text model that supports structured output.

B.2 Agent Benchmark Construction and Evaluation

We build a dedicated benchmark to measure whether the agent can infer accurate source-level spatial conditions from mono audio and a natural user prompt. The benchmark contains 160 clips in total: 80 simulated clips (MUSBench-S) and 80 real STARSS23 clips (MUSBench-R). Each item

contains a mono waveform, a standardized trajectory annotation, the raw metadata, and a single user-style prompt that combines scene description, source activity, and coarse spatial hints.

Data Sampling. The simulated subset is sampled from our synthesized FOA metadata. For real-world recordings, we utilize the STARSS23 FOA evaluation split, segmenting long recordings into clips under 30 seconds. To ensure cross-dataset consistency, we preprocess the simulated data to match the official STARSS23 metadata schema, and conversely, convert the official frame-level trajectories into our simulation format, thereby unifying the data pipeline for subsequent evaluation.

User Prompt Generation. To mimic realistic deployment, user prompts in our benchmark are not synthesized via rigid templates. Instead, we leverage Gemini 2.5 Pro to perceive the audio content and rewrite the raw metadata into natural language user instructions. Each prompt describes the acoustic scene, approximate source types, active temporal intervals, and coarse spatial hints (e.g., left/right, front/back, elevation, and distance). This design aligns with the intended user interface of MUSAudio while avoiding explicit exposure of dense frame-level trajectories. Following a strict pipeline of manual verification and quality filtering, we curate a final benchmark consisting of 80 distinct evaluation samples each for both the simulated and real-world subsets.

Data Statistics. Ultimately, the compiled MUSBench-R dataset consists of 15 single-source, 37 two-source, and 28 three-source audio clips, with an average duration of 23.690 seconds. For MUSBench-S, the dataset comprises 40 two-source and 40 three-source clips, featuring an average duration of 19.471 seconds.

Evaluation Protocol. Given a prediction, we first sanitize its schema and match predicted sources to ground-truth sources by minimizing the cost

$$(1 - \mathrm{tIoU}) + \frac{\mathrm{AngErr}}{180} + \mathrm{ClassMismatch}. \quad (18)$$

The class mismatch term is 0 for exact or substring matches, 0.1 for speech-to-speech matches, 0.2 if either side is unknown, and 0.5 otherwise. Since each clip contains at most three sources, exhaustive assignment is sufficient.

For each matched source, we compute temporal IoU over active segments. Spatial errors are computed by sampling the union of active intervals at 0.1 second resolution, interpolating both trajectories, and measuring spherical DOA angular distance. Distance use absolute error, and DOA@τ checks whether the mean angular error is below τ degrees. If a clip has no matched source, we assign default penalties: angular error $180°$, distance MAE 10 meters, and DOA accuracy 0.

C Model Details

C.1 FOA-aware VAE Architecture Details

This appendix supplements Section 3.4 with full architecture specifications and training configurations of the FOA-aware VAE.

Encoder-Decoder Architecture. The FOA-aware VAE is built on the AudioVAE backbone, a causal encoder-decoder architecture. The encoder applies a sequence of strided causal convolutions with downsampling rates $[8, 5, 4, 3]$, yielding a total temporal downsampling factor of 480. A diagonal Gaussian bottleneck produces a latent representation of dimension 64 per channel. The decoder mirrors the encoder with transposed convolutions at matching upsampling rates. Both encoder and decoder are augmented with causal Transformer post-modules to capture long-range temporal dependencies. Table 5 summarizes the key architectural hyperparameters.

Table 5: **FOA-aware VAE architecture hyperparameters.**

Hyperparameter	Value
Sample rate	48 kHz
Input / output channels	4 (FOA)
Segment length	192 000 samples (≈4 s)
Total downsampling factor	1920
Latent dimension (per channel)	128

Mono-to-FOA Weight Initialization. The VAE is initialized from a pretrained mono AudioVAE checkpoint. For the first encoder convolution and the final decoder convolution, whose channel dimensions change from 1 to 4, we adopt a *repeat* strategy: the mono convolution kernels are replicated across four channels and rescaled by a factor of $1/4$. This preserves the activation magnitude of the pretrained model at the start of fine-tuning while allowing subsequent training to learn cross-channel spatial dependencies from FOA data.

Discriminator Details. Five waveform and spectral discriminators are adapted to accept the full four-channel FOA tensor: the Multi-Period Discriminator (MPD; periods $[2, 3, 5, 7, 11]$), Multi-Scale Discriminator (MSD), Multi-Resolution Discriminator (MRD) and Multi-Scale STFT Discriminator (MSSTFTD; FFT sizes $[206, 334, 542, 876, 1418, 2296]$). Each discriminator's first convolution layer is expanded from mono to four-channel input using the same repeat initialization strategy. The discriminator learning rate is 1×10^{-4} with Adam (β_1=0.5, β_2=0.9).

C.2 Flow Matching Model Architecture

This appendix supplements Section 3.3 and Section 3.5 with full model architecture and inference configurations.

Transformer Architecture. The flow-matching generator is a Transformer-based velocity-field predictor operating on FOA VAE latent sequences. It consists of 18 blocks organized into two stages: 6 *joint* blocks where context tokens (scene and source text) and audio tokens attend to each other via multi-source attention routing, followed by 12 *fused* blocks where audio tokens undergo self-attention with causal convolution and rotary positional embeddings. All blocks share a hidden dimension of 512 and 16 attention heads with an MLP expansion ratio of 4. Table 6 summarizes the key hyperparameters.

Table 6: **Flow matching model hyperparameters.**

Hyperparameter	Value
Hidden dimension	512
Total depth	18 blocks
Joint blocks (context + audio)	6
Fused blocks (audio only)	12
Attention heads	16
MLP ratio	4.0
Chunk size	16
FOA latent dimension	128
Mono latent dimension	128
Latent frame rate	25 Hz
Max latent sequence length	1024 frames (≈40 s)
Text encoder	T5 (1 024-dim)
Max scene text tokens	256
Max source text tokens	128
Spatial Fourier frequencies	8
Max sources per scene	3

Classifier-Free Guidance. During training, conditions are independently dropped with the following null probabilities to enable classifier-free guidance: scene text 0.1, per-source text 0.2, trajectory 0.2, and mono audio 0.2. At inference, generation uses 20 Euler ODE integration steps from t=1 (noise) to t=0 (data) with a classifier-free guidance scale of 3.0.

D Experiments

D.1 PSELDNet

To evaluate whether the generated FOA waveform contains physically plausible and semantically consistent spatial cues, we use PSELDNet (Hu et al., 2025) as an external spatial-audio evaluator. PSELDNet performs sound event localization and detection (SELD) directly from multi-channel audio and predicts both the active sound-event classes and their directions of arrival.

We start from the public PSELDNet checkpoint based on the mACCDOA-HTSAT architecture and fine-tune it on the real STARSS23 development recordings. Fine-tuning follows the official one-GPU setting with a batch size of 32, using the pretrained PSELDNet checkpoint for the SELD model and the pretrained HTSAT checkpoint for the audio backbone. We use the best checkpoint after two epochs in our evaluation.

The two-epoch checkpoint obtains a pseudo-SELD score of 0.2923 under macro averaging and 0.3340 under micro averaging on 765 files and 13 event classes. The corresponding macro metrics are ER $= 0.5050$, F $= 0.6189$, LE $= 17.47°$, and LR $= 0.8140$; the micro metrics are ER $= 0.5050$, F $= 0.5556$, LE $= 18.22°$, and LR $= 0.7147$. These results indicate that the fine-tuned PSELDNet can reliably detect and localize sound events on real FOA recordings, and we therefore use it as a fixed evaluator for both reference and generated FOA audio. During evaluation, both generated and reference FOA signals are resampled to the PSELDNet input sampling rate and passed through the same fine-tuned model.

D.2 Metric

Objective metrics. For FOA reconstruction, we report spectral fidelity and spatial consistency metrics. Multi-resolution STFT (MRSTFT) measures waveform reconstruction error in the time–frequency domain by averaging spectral convergence and magnitude differences over several STFT resolutions. Log-spectral distance (LSD) measures the frame-level discrepancy between the log spectra of the reference and reconstructed signals. Both metrics are computed channel-wise on

the FOA waveform and averaged over the four channels; lower values indicate better reconstruction.

For spatial evaluation, we use the fine-tuned PSELDNet described in Appendix D.1. We adopt standard SELD metrics: error rate (ER), F-score (F), localization error (LE), localization recall (LR), and the combined SELD score. ER and F measure event-detection accuracy, LE measures the angular error in degrees between matched predicted and reference directions, and LR measures the fraction of reference events that are correctly localized. The SELD score combines detection and localization errors into a single value, where lower is better.

For spatial audio generation, we additionally evaluate semantic and distributional audio quality. LAION-CLAP similarity is computed between generated audio and the corresponding text description to measure audio–text semantic alignment; higher similarity indicates better prompt consistency. Fréchet Audio Distance (FAD) compares the embedding distribution of generated audio with that of reference audio; lower FAD indicates that the generated samples are closer to real recordings in the embedding space.

Subjective metrics. We conduct subjective evaluations of the generation tasks using Mean Opinion Score (MOS). For each task, we randomly sample 40 utterances from the test set. Each utterance is paired with the corresponding source-position prompt, and the generated FOA audio is rendered into binaural audio using HRIRs (Head-Related Impulse Response) for headphone-based listening. Each sample is evaluated by at least five expert listeners. Listeners rate each sample on a five-point Likert scale ranging from 1 (bad) to 5 (excellent). All participants are fairly compensated for their time at a rate of $15 per hour.

For audio quality, we use MOS-Q. Listeners wear headphones and assess the clarity, naturalness, and absence of artifacts in the generated audio. For spatial perception, we use MOS-P. Listeners evaluate the realism of the spatial cues and judge whether the perceived direction and distance of the sound source match the prompt description. Higher MOS-Q and MOS-P indicate better perceived audio quality and spatial plausibility, respectively.

E Licenses and Availability

We respect the original licenses and terms of use of all external artifacts used in this work. This work uses publicly available datasets, including STARSS23, AudioSet, and Emilia. We do not redistribute the original third-party audio data. Users who wish to reproduce our dataset or experiments should obtain the original datasets from their respective providers and comply with their licenses and terms of use.

We will release our code, preprocessing scripts, benchmark questions, and derived annotations to support reproducibility. The artifacts created by us, including derived annotations and benchmark files, will be released under the CC BY 4.0 license where applicable.

F Use of AI Assistants

We used AI-based writing assistants during manuscript preparation solely for language polishing, including grammar checking, spelling correction, and improving readability. The scientific ideas, experimental design, analysis, and conclusions were developed and verified by the authors.

During code implementation, we used AI coding assistants to support programming tasks, such as generating boilerplate code, debugging, and implementation suggestions. All generated or assisted code was reviewed, tested, and validated by the authors before use in the experiments.

MUSAudio Mono-to-FOA Agent Skill

Name: mono-to-foa-agent
Description: End-to-end MUSAudio agent workflow for converting mono audio and a user prompt into multi-channel FOA spatial audio...

Inputs & Output Contract

- **Inputs:** Mono audio path, user instruction, target dataset mode (`sim`/`starss`), output paths, and MLLM backend selection.
- **Output Schema:** Every planned item must match the following JSONL structure:

```
{
  "scene_id": "...", "dataset": "...", "status": "...",
  "prediction": {
    "scene": "...",
    "sources": [
      {
        "id": 1, "description": "...", "class_label": "...",
        "active_segments": [{"start": 0.0, "end": 1.0}],
        "trajectory": [{"time": 0.0, "azimuth": 0.0,
                "elevation": 0.0, "distance": 1.5}],
        "is_dynamic": false
      }
    ]
  },
  "review": {"passed": true, "issues": []}, ...
}
```

- **Coordinate Convention:** Time (s), Azimuth ([-180, 180]), Elevation ([-90, 90]), Distance (m). Max three active sources by default.

Agent Workflow Stages

1. **Basic attribute perception:** Call MLLM to extract global scene type, source count, and category descriptions.
2. **Temporal reasoning:** Call MLLM/Sound Source Localization Model to extract the coarse activity intervals of the sound sources, and motion cues.
3. **Spatial planning:** Estimate trajectories via (1) user constraints, (2) audio cues, and (3) spatial-audio perceptual priors.
4. **Verification:** A second MLLM call verifies schema validity, coordinates, constraints, and source separation. Triggers up to N repair loops if `passed=false`.
5. **FOA spatialization:** Invoke the trained spatialization pipeline via: `bash generation.sh`

Figure 4: MUSAudio mono-to-FOA agent skill.