MOMO: HUMAN MOTION FROM MOVING SOUNDS

Anonymous authors
Paper under double-blind review

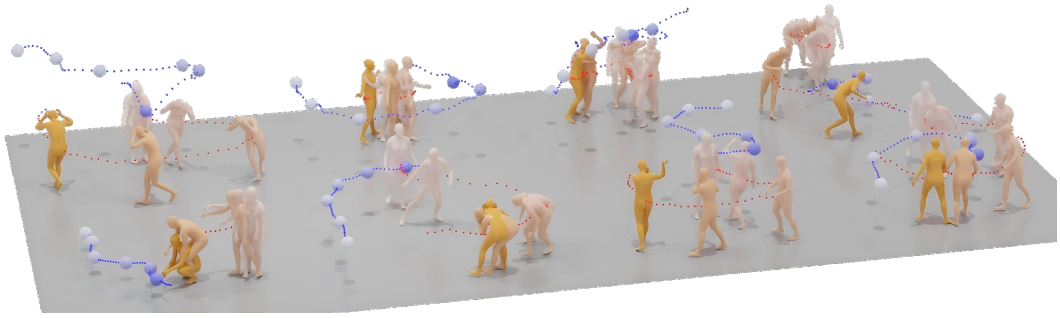


Figure 1: This paper proposes a novel task for synthesizing 3D human motions in response to *dynamic spatial audio*. To support this new task, we collect DySAM, the first dataset that records synchronized dynamic spatial sound source trajectories and high-quality 3D human motion. In this figure, the model generates diverse and realistic human reactions to various moving acoustic events.

ABSTRACT

Human responses to sound depend not only on its semantic content, but also on the evolving spatial relationship between source and listener, conveyed through spatial audio and its temporal dynamics. Modeling human reactions to such dynamic spatial audio is important for character animation, human behavior analysis, and robotics. As of yet, existing audio-driven motion generation methods primarily focus on non-spatial or static acoustic conditions, leaving this important problem largely underexplored. To bridge this gap, we formulate the novel task of dynamic spatial-audio-driven human motion generation and introduce two motion-captured datasets. We introduce the **Dy**namic **S**patial **A**udio and **M**otion dataset (DySAM) as the first dataset and benchmark for this task, providing synchronized binaural audio, 3D human motions, and moving sound source trajectories. Given the importance of dynamic spatial cues of sound sources, we further provide the **Dy**namic **S**patial **A**udio **L**ocalization dataset (DySAL), an auxiliary binaural localization dataset that pairs diverse acoustic events with head poses and moving source trajectories to support spatial audio perception. For benchmarking, we propose MOMO for human **Mo**tion generation from **Mo**ving Sounds, a diffusion-based framework guided by three coupled principles: localization, understanding, and generation. MOMO performs localization by estimating head-centric spatial direction cues of the moving source directly from raw binaural audio, avoiding reliance on ground-truth source coordinates. It performs understanding by extracting frame-wise audio-content representations, and performs generation by conditioning motion diffusion on both semantic features and dynamic spatial cues. Moreover, MOMO incorporates frame-wise motion correction during denoising, enabling temporally adaptive motion synthesis under evolving audio-spatial cues. Experiments show that MOMO outperforms existing baselines in motion fidelity, audio-motion alignment, and physical plausibility, demonstrating the potential of dynamic spatial audio for embodied motion generation.

1 INTRODUCTION

Human responses to sound arise from integrating semantic meaning with *spatial audio* cues and the *temporal dynamics of sound sources*. A barking dog approaching from behind may elicit avoidance, whereas a speeding car approaching may prompt reorientation. Crucially, the same sound can evoke

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

different actions depending on its source's location and motion relative to the listener. Modeling this perception–action coupling supports context-aware virtual humans and embodied agents in character animation, virtual reality, human–computer interaction, and embodied intelligence.



Figure 2: Diverse Reaction to Moving Sounds.

Existing human motion datasets and methods have mainly focused on conditions such as action labels (Guo et al., 2020), text descriptions (Guo et al., 2022), music (Zhuang et al., 2022; Li et al., 2021; Tseng et al., 2023; Li et al., 2023; 2024; Luo et al., 2024), and speech (Ao et al., 2022; 2023; Alexanderson et al., 2023). Recently, to the best of our knowledge, MOSPA (Xu et al., 2025) introduced the first spatial-audio-driven human motion dataset, SAM, along with a corresponding motion generation method, showing that spatial audio cues are crucial for generating realistic human motions. However, the SAM dataset mainly focuses on a static sound source, where the sound source position remains fixed within each sequence. Moreover, MOSPA requires the ground-truth absolute position of the sound source as a condition for motion generation, which is difficult to obtain in real-world scenarios and limits its applicability to embodied agents. Therefore, dynamic spatial-audio-driven human motion generation, which models human motion responses to time-varying spatial audio from moving sound sources, remains largely underexplored, especially in practical settings without access to ground-truth sound source positions.

To address this gap, we introduce DySAM, a dataset for dynamic spatial-audio-driven human motion generation. DySAM contains synchronized binaural audio, 3D human motion, and dynamic sound source trajectories captured in an indoor environment. Human motion and sound source trajectories are recorded using a Vicon motion capture system (Vicon, 2026), while binaural audio is captured by high-sensitivity microphones near the actors' ears, approximating the audio perceived by the actors. The main dataset contains 2,336 ten-second audio-motion sequences, totaling 23,360 seconds (approximately 6.5 hours), with motion at 30 FPS and audio at 48 kHz. It covers 42 audio types and 109 unique audio sources, including human sounds, animal sounds, vehicles, and environmental sounds, with 8 subjects and a binary genre label indicating sensitive or insensitive reactions. In addition, we provide DySAL, an auxiliary sound source localization dataset of 11,730 seconds, pairing binaural audio of diverse acoustic events with listener head 6DoF poses and frame-wise moving sound source annotations to support dynamic spatial audio perception.

Motivated by the fact that human reactions to spatial sound depend on both where the sound comes from and what the sound is, we formulate spatial-audio-driven motion generation around three coupled functional components: *localization*, *understanding*, and *generation*. Guided by this design principle, we propose MOMO, a diffusion-based framework for generating human motion conditioned on dynamic spatial audio. For localization, MOMO estimates frame-wise head-centric spatial cues of sound source from raw binaural audio using a sound source localization network, eliminating the reliance on privileged ground-truth source coordinates. For understanding, a pretrained CLAP audio encoder (Wu et al., 2023) encodes the binaural audio to obtain frame-wise audio-content features. For generation, MOMO employs a diffusion-based motion generator conditioned on the predicted spatial cues, CLAP audio features, and genre label. Since human motion needs to continuously adapt to the evolving spatial cues of the sound source, while self-attention only provides implicit interactions between motion tokens and condition tokens, we further introduce a *Frame-wise Motion Correction* module in the Transformer denoiser to explicitly refine motion tokens with frame-wise conditions, along with a *Correction Strength Predictor* that adaptively controls the correction magnitude for each frame.

By using head-centric DoA estimated directly from binaural audio, MOMO provides a more practical formulation for spatial-audio-driven motion generation in embodied robot settings. Compared with MOSPA (Xu et al., 2025), which relies on ground-truth global sound source coordinates, our egocentric formulation avoids privileged source-location annotations and better reflects how embodied agents infer dynamic sound direction from their own auditory observations. Extensive quantitative and qualitative experiments show that MOMO outperforms existing baselines, producing more realistic motions that respond to dynamic spatial-audio cues.

Our contributions are summarized as follows:

- We introduce two datasets: DySAM, to the best of our knowledge the first dataset for dynamic spatial-audio-driven human motion generation, and DySAL, an auxiliary sound source localization dataset collected in indoor reverberant environments.
- We propose MOMO, a practical framework for generating human motion from binaural audio without requiring ground-truth source coordinates. It integrates sound source localization, CLAP-based audio content encoding, and diffusion-based motion generation to model human responses to moving sound sources.
- Extensive quantitative and qualitative experiments show that MOMO generates natural, plausible, and spatially consistent human motion, better aligning with real motion and outperforming existing baselines.

2 Related Work

Conditional Human Motion Generation. Conditional human motion generation has been extensively studied under various conditions, such as action labels (Guo et al., 2020), textual descriptions (Guo et al., 2022; Zhang et al., 2023; Dabral et al., 2023), and motion language models that treat human motion as a foreign language (Jiang et al., 2023; Zhang et al., 2024). These methods generate diverse and natural motions from semantic conditions, but they generally do not consider audio, let alone its spatial attributes. In contrast, our work focuses on audio-driven conditions with explicit spatial information.

Audio-Driven Human Motion Generation. Audio-driven human motion generation has been studied mainly in two directions. The first is music-driven dance generation (Tang et al., 2018; Zhuang et al., 2022; Li et al., 2022; 2023; Siyao et al., 2022; Tseng et al., 2023; Li et al., 2024; Luo et al., 2024), which generates choreography that matches the rhythm and style of the music. The second is speech-driven gesture synthesis (Ginosar et al., 2019; Yoon et al., 2020; Ao et al., 2022; 2023; Alexanderson et al., 2023), which generates co-speech gestures from speech content and prosody. Most of these methods use monaural or stereo audio and ignore the spatial attributes of the sound source. MOSPA (Xu et al., 2025) is the first to study spatial-audio-driven human motion generation with binaural audio, but it assumes a static sound source position. Different from MOSPA, we consider dynamic spatial audio, where the source moves over time, and the DoA changes accordingly.

Spatial Sound Understanding. Estimating the direction of a sound source from binaural signals is a classic problem in acoustic scene analysis, for instance using binaural cues combined with deep learning (García-Barrios et al., 2022; Min et al., 2025). Recent works also model room acoustics and room impulse responses to support sound source localization in complex indoor scenes (Chen et al., 2023; 2022; Luo et al., 2022; Ratnarajah et al., 2022; Um et al., 2025). In this paper, we adopt a sound source localization network to predict the head-centric DoA from binaural audio, which serves as the spatial condition for motion generation. This makes our framework practical in real-world scenarios where ground-truth source coordinates are unavailable.

3 Datasets

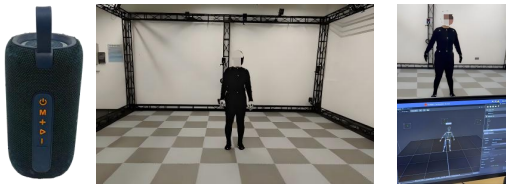
This section presents two newly collected motion-capture datasets for dynamic spatial-audio-driven human motion generation. The Dynamic Spatial Audio and Motion dataset (DySAM) serves as the main benchmark for this task, providing synchronized binaural audio, 3D human motions, and moving sound source trajectories. To support broader research on dynamic spatial audio perception, we introduce the Dynamic Spatial Audio Localization dataset (DySAL) that pairs diverse acoustic events with listener head poses and mocap-recorded moving source trajectories.

Data Capture Setup. We collect the data in an indoor environment with natural reverberation using a Vicon motion capture system (Vicon, 2026), which records both human motion and the trajectory of a moving sound source (as shown in Fig. 3). We use an omnidirectional speaker to play the dynamic audio. We choose binaural audio as the audio modality, following the SAM dataset (Xu et al., 2025), as it is a common form of spatial audio that aligns with human auditory perception and can be readily applied to virtual characters and robotic platforms. Binaural audio is recorded

Table 1: **Statistics of the DySAM dataset.** DySAM is the first dataset for dynamic spatial-audio-driven human motion generation, containing synchronized binaural audio, 3D human motion, and dynamic sound source trajectories (Dyn. SS). We also provide DySAL, an auxiliary sound source localization dataset containing dynamic binaural recordings of diverse acoustic events with frame-wise source annotations.

Dataset	**Dyn. SS**	SS	$3DJoint_{pos/rot}$	Model	Joints	Subjects	Hours
Dance with Melody (Tang et al., 2018)	×	×	✓/×	-	21	-	1.6
DanceNet (Zhuang et al., 2022)	×	×	✓/×	-	55	2	0.9
AIST++ (Li et al., 2021)	×	×	✓/✓	COCO/SMPL	17/24	30	5.2
PopDanceSet (Luo et al., 2024)	×	×	✓/✓	COCO/SMPL	17/24	132	3.5
FineDance (Li et al., 2023)	×	×	✓/✓	SMPL & hand joints	52	27	14.6
SAM (Xu et al., 2025)	×	✓	✓/✓	SMPL-X	55	12	9.5
DySAM (Ours)	✓	✓	✓/✓	SMPL-X	55	8	6.5
+ DySAL (Ours)	✓	✓	✓/✓	SMPL-X	55	1	3.3

using two microphones placed near the actor's ears, allowing the signals to approximate the auditory input the actor perceives during capture. Specifically, we use Sound Professionals MS-TFB-2 microphones (The Sound Professionals, 2026) and a Deity PR-2 recorder (Deity Microphones, 2026) for binaural audio recording. The audio recorder is synchronized with the mocap system using timecode, ensuring temporal alignment among binaural audio, human motion, and sound source trajectories.



(i) Speaker (ii) Mocap Environment (iii) Mocap Interface

Figure 3: Data Collection Setup.

3.1 DySAM Dataset

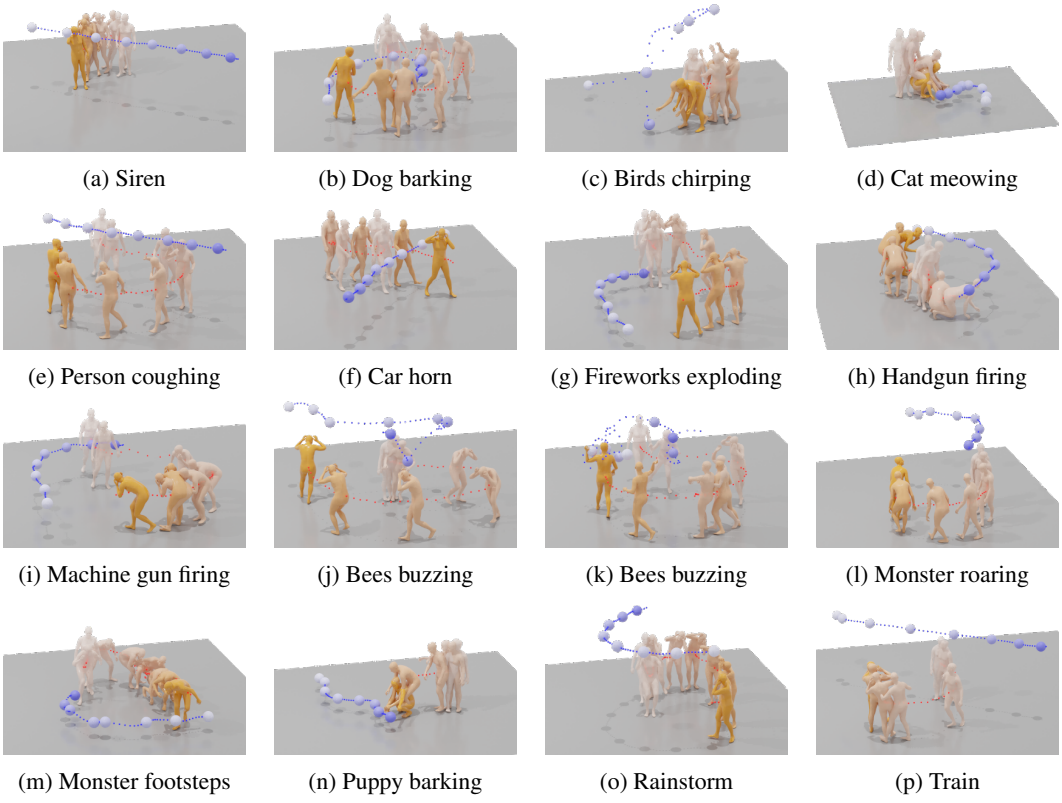
Our main dataset, DySAM, is collected for dynamic spatial-audio-driven human motion generation, with visualization snapshots provided in Fig. 4. Each sequence consists of a 10-second binaural audio clip, the corresponding 3D human motion stored in the SMPL-X format (Pavlakos et al., 2019), and the trajectory of a moving sound source in the world coordinate system. As summarized in Tab. 1, DySAM contains 2,336 synchronized audio-motion sequences, amounting to 23,360 seconds (approximately 6.5 hours) of paired data. The captured motions are represented as SMPL-X model (Pavlakos et al., 2019). The audio's sampling rate is 48,000 Hz, while the motion is represented at 30 FPS for network training. DySAM covers 42 audio types and 109 unique audio sources, spanning diverse daily-life acoustic events such as animal sounds, vehicles, alarms, music, and human speech. It was collected with 8 actors, including 5 female and 3 male. Each sequence is annotated with one of two reaction genres, sensitive or insensitive, with an approximately balanced 1:1 ratio.

3.2 DySAL Dataset

We additionally collect DySAL, an auxiliary sound source localization dataset, using the same motion capture setup. It provides abundant spatial supervision for training the SSL network, enabling robust estimation of dynamic head-centric DoA cues from binaural audio. It contains 11,730 seconds, approximately 3.3 hours, of binaural audio paired with frame-wise sound source positions and the listener's head 6DoF pose in the world coordinate system. These spatial cues can facilitate broader studies of spatial audio perception, such as absolute source localization and direction-of-arrival estimation. To ensure the diversity of acoustic events, we collect raw audio from AIST (Tsuchida et al., 2019), ESC-50 (Piczak, 2015), and UrbanSound8K (Salamon et al., 2014), covering a wide range of music, environmental, human, animal, mechanical, and urban sounds. Further details on the data collection and data processing pipeline for the two datasets are provided in Appendix A.

4 Method

We propose MOMO, a diffusion-based framework for dynamic-spatial-audio-driven human motion generation. In real world, human reactions to spatial sound depend on both spatial direction and



(a) Siren (b) Dog barking (c) Birds chirping (d) Cat meowing

(e) Person coughing (f) Car horn (g) Fireworks exploding (h) Handgun firing

(i) Machine gun firing (j) Bees buzzing (k) Bees buzzing (l) Monster roaring

(m) Monster footsteps (n) Puppy barking (o) Rainstorm (p) Train

Figure 4: Snapshots from the our DySAM dataset, spanning human sounds, animal sounds, and everyday environmental sounds. In each panel, the colors of the actor and the sound source transition from light to dark over time to indicate temporal evolution. Red dots mark the movement trajectory of the actor, while blue dots mark the trajectory of the moving sound source.

audio content. Accordingly, MOMO consists of three coupled functional components: *localization*, *understanding*, and *generation*, as illustrated in Fig. 5. For localization, a sound source localization (SSL) network estimates a frame-wise head-centric direction-of-arrival (DoA) sequence $\boldsymbol{D} = \{d_t \in \mathbb{R}^3\}_{t=1}^{T}$ from raw binaural audio, T is the max length of motion (Sec. 4.1). For understanding, a pretrained CLAP audio encoder extracts temporally aligned audio content representations $\boldsymbol{A} = \{a_t\}_{t=1}^{T}$ from the mono audio. For generation, a transformer-encoder-based conditional diffusion model synthesizes motion through frame-wise correction conditioned on $\boldsymbol{D}$, $\boldsymbol{A}$, the genre label g, and the diffusion timestep τ. Rather than relying solely on attention mechanism, it incorporates *Frame-wise Motion Correction* and *Correction Strength Predictor*, allowing the denoiser to adaptively determine the correction magnitude for each frame under dynamic spatial-audio conditions (Sec. 4.2). For network training, we adopt a 272-dimensional motion representation following (Xiao et al., 2025).

4.1 Sound Source Localization (SSL) Network

In real-world scenes, ground-truth absolute global coordinates of sound sources are often unavailable. To obtain a motion-relevant and coordinate-invariant spatial cue, we represent the sound source at each frame as a head-centric direction-of-arrival (DoA) unit vector, transformed from the source position in the world coordinate system to the head coordinate system using the 6DoF head pose. Specifically, the SSL network predicts a sequence of frame-wise DoA vectors $\hat{\boldsymbol{D}} = \{\hat{d}_t\}_{t=1}^{T}$, where each $\hat{d}_t \in \mathbb{R}^3$ is a unit vector. This head-centric representation avoids the need for absolute global source coordinates.

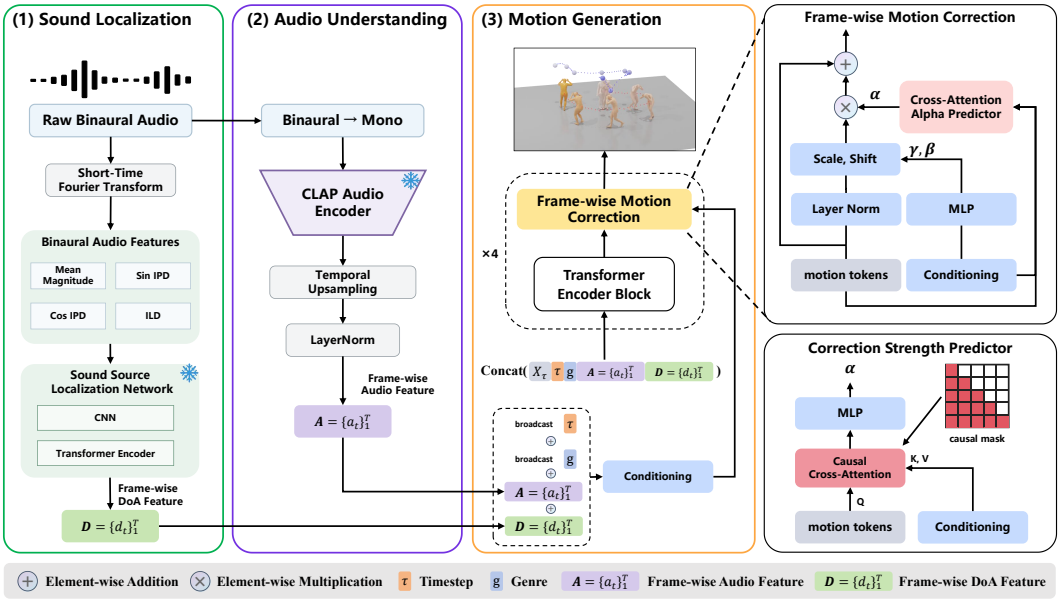


Figure 5: **Overview of MOMO framework.** MOMO comprises three coupled functional components: a sound source localization (SSL) network predicts a frame-wise head-centric DoA sequence $\hat{\boldsymbol{D}}$ from binaural audio (*localization*); a CLAP audio encoder extracts frame-wise audio content features $\boldsymbol{A}$ from mono audio (*understanding*); and a conditional diffusion model that iteratively denoise the Gaussian noise to generated motion, integrating Frame-wise Motion Correction to adaptively correct each motion according to frame-wise conditions (*generation*).

Our SSL network is inspired by the binaural sound localization framework of Garcia et al. (García-Barrios et al., 2022), which employs STFT-based binaural cues with a cascaded CNN-GRU architecture. We follow their use of STFT-based binaural spatial features and a CNN front-end for extracting local spectro-temporal patterns, but replace the GRU-based temporal modeling module with a Transformer encoder to better capture long-range temporal dependencies under reverberant environments. The SSL network is trained to predict the frame-wise head-centric DoA unit vector using a cosine-distance loss: $\mathcal{L}_{\text{ssl}} = \frac{1}{T}\sum_{t=1}^{T}\left(1 - \hat{d}_t \cdot d_t\right)$, where $\hat{d}_t$ and d_t denote the predicted and ground-truth DoA unit vectors, respectively, both normalized in the head coordinate system. Once trained, the SSL network is frozen and used to predict the spatial condition sequence $\hat{\boldsymbol{D}}$. Please refer to Appendix B for more details on the SSL architecture and training setup.

4.2 Conditional Diffusion Motion Generation

We formulate motion generation as conditional denoising diffusion (Ho et al., 2020). Given noisy motion $\mathbf{x}_\tau$ at diffusion step τ, the denoising network predicts the clean motion sequence $\mathbf{x}_0$ conditioned on audio content, DoA, genre, and timestep embeddings. For audio content, we average the binaural signal into a mono waveform and feed it into a pretrained CLAP encoder (Wu et al., 2023). The extracted hidden features are pooled over channel and frequency dimensions and temporally aligned with the motion sequence, producing frame-wise audio features $\boldsymbol{A} = \{a_t\}_{t=1}^{T}$.

Building on the Transformer encoder of Xu et al. (2025), we project $\mathbf{x}_\tau$ into motion tokens $\boldsymbol{m} = \{\mathbf{m}_t\}_{t=1}^{T}$ and concatenate them with condition tokens derived from audio features, DoA features, genre, and the diffusion-step embedding. The concatenated tokens are fed into the Transformer encoder for conditional denoising. Since the attention mechanism only provides implicit interactions between motion tokens and the frame-wise condition tokens, we further introduce a *Frame-wise Motion Correction* module to explicitly correct motion tokens based on frame-wise conditions, together with a *Correction Strength Predictor* that adaptively determines the correction strength.

Frame-wise Motion Correction. To explicitly leverage both global and frame-wise conditions, we apply a residual correction module based on Adaptive Layer Normalization (AdaLN) on after each Transformer encoder layer to refine the intermediate motion token, which is formulated as

$$\boldsymbol{m}' = \boldsymbol{m} + \boldsymbol{\alpha} \odot \boldsymbol{\delta}, \tag{1}$$

where $\boldsymbol{\delta} = \{\delta_i\}_{i=1}^{T}$ denotes the frame-wise correction, and $\boldsymbol{\alpha} = \{\alpha_i\}_{i=1}^{T}$ is an frame-wise correction strength. The modulation condition $\boldsymbol{c} = \{c_i\}_{i=1}^{T}$ consists of four components: the timestep embedding t, the genre embedding g, the frame-wise audio feature $\boldsymbol{A}$, and the frame-wise DoA feature $\boldsymbol{D}$. The unified condition sequence is obtained by summing these components at each frame, with the global timestep and genre embeddings broadcast to all T frames: $c_i = e_t + e_g + a_i + d_i$. This condition sequence is then used to modulate the motion tokens. An MLP then estimates scale and shift sequences $(\boldsymbol{\gamma}, \boldsymbol{\beta}) = \mathrm{MLP}(\mathbf{c}_i)$ for AdaLN (Peebles & Xie, 2023), yielding the correction:

$$\boldsymbol{\delta} = \mathrm{FFN}\left(\mathrm{LN}(\boldsymbol{m}) \odot (1 + \boldsymbol{\gamma}) + \boldsymbol{\beta}\right), \tag{2}$$

Correction Strength Predictor. The correction strength should not be a fixed parameter tied to a specific layer or frame position. Instead, the strength of motion correction should depend on the audio content, spatial DoA cues, genre, and the current denoising step. We therefore predict $\boldsymbol{\alpha}$ dynamically from the interaction between motion tokens and frame-wise conditions.

Specifically, we use motion tokens as queries $\boldsymbol{Q} = \boldsymbol{m}$ and modulation condition as keys and values $\boldsymbol{K} = \boldsymbol{V} = \boldsymbol{c}$. Considering that human responses to sounds may exhibit temporal delays, we apply a causal attention mask to prevent the token at frame t from attending to future condition frames. The correct strength is bounded by

$$\boldsymbol{\alpha} = \alpha_{\max} \cdot \mathrm{Sigmoid}(\mathrm{MLP}_{\alpha}(\mathrm{CrossAttn}(\boldsymbol{m}, \boldsymbol{c}_{\leq t}, \boldsymbol{c}_{\leq t}))). \tag{3}$$

This design decouples the estimation of the condition-dependent correction and its strength: the AdaLN modulation module predicts the correction term, while the cross-attention predictor dynamically determines how strongly this correction is applied.

4.3 Training Objectives

We train MOMO a reconstruction loss: $\mathcal{L}_{\mathrm{diff}} = \|\boldsymbol{x_0} - \hat{\boldsymbol{x_0}}(\mathbf{x}_\tau, \tau, \mathbf{c})\|_2^2$. To ensure temporal continuity and reduce foot-sliding artifacts, we further use a delta loss $\mathcal{L}_{\mathrm{del}} = \|\hat{\boldsymbol{x_0}} - \boldsymbol{x_0}\|_2^2$ and a foot contact loss $\mathcal{L}_{\mathrm{foot}}$ (Xu et al., 2025). The overall objective is:

$$\mathcal{L} = \mathcal{L}_{\mathrm{diff}} + \lambda_{\mathrm{del}}\mathcal{L}_{\mathrm{del}} + \lambda_{\mathrm{foot}}\mathcal{L}_{\mathrm{foot}}. \tag{4}$$

5 Experiment

Experimental Setup. We split the main DySAM dataset into train/validation/test sets with a ratio of 8:1:1. For training the sound source localization network, we transform the global sound source position into head-centric unit direction vector using the head 6DoF pose, and use it as the ground-truth DoA. Regarding metrics, we use FID, R-precision, Diversity (Guo et al., 2022), Average Pairwise Distance (APD) (Dou et al., 2023), and a physics-based skating ratio, as detailed in Appendix D. We introduce a multi-branch motion-audio-source evaluator for feature extraction, aiming to better measure the spatial alignment between the generated motion and the moving sound source; details are provided in Appendix C. We compare with diffusion-based and VQ-based baselines, including EDGE (Tseng et al., 2023), POPDG (Luo et al., 2024), MOSPA (Xu et al., 2025), and MoMask++ (Guo et al., 2025). All baselines are trained on the same DySAM split, with their original input conditions replaced by traditional acoustic features for fair comparison.

5.1 Comparisons

Table 2 reports the quantitative comparisons on the DySAM test set. MOMO achieves best overall performance across all metrics. Among the baselines, MoMask++ (Guo et al., 2025) obtains a competitive FID but suffers from lower R-precision and a higher skating ratio, indicating weaker audio-motion alignment and physical plausibility. MOSPA (Xu et al., 2025) achieves the second-best R-precision by using ground-truth source positions, while MOMO estimates head-centric DoA directly from binaural audio and obtains better overall results. Qualitative comparisons are shown in Fig. 6, with more results in the supplementary videos.

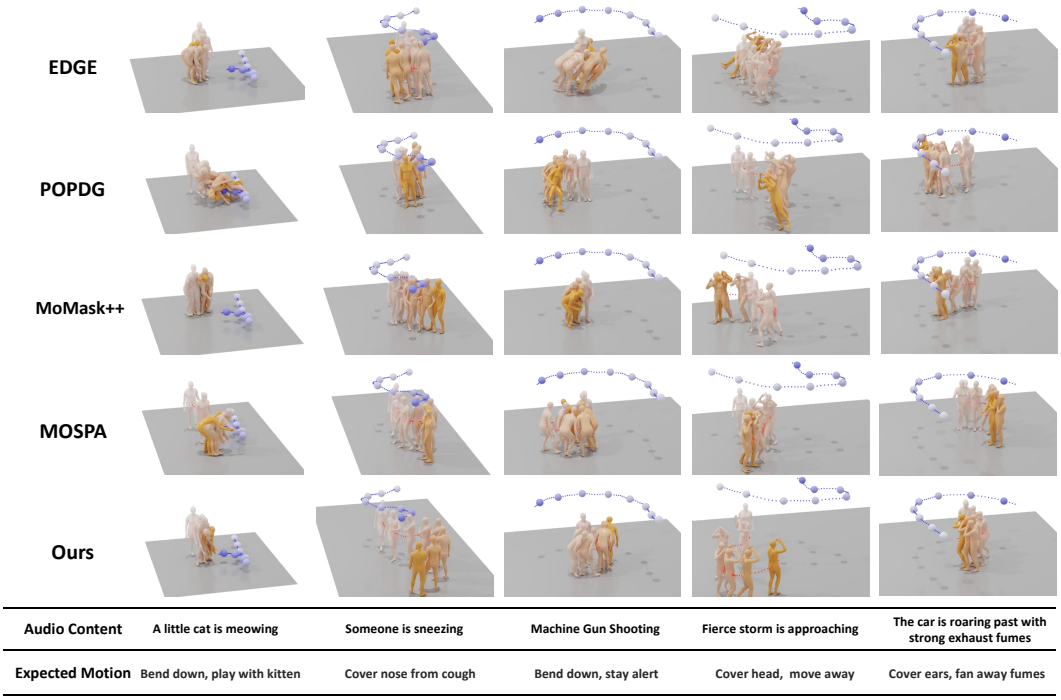


Figure 6: Qualitative comparison of state-of-the-art methods for the dynamic spatial audio-to-motion task. We visualize motion results from five cases.

Table 2: **Quantitative Comparisons on the DySAM Dataset.** ± indicates a 95% confidence interval. **Bold face** indicates the best result. → means the closer to Real Motion the better.

Method	R-precision ↑			FID ↓	Diversity →	APD →	Skating Ratio (%) ↓
	Top1	Top2	Top3				
Real Motion	$0.720^{\pm0.000}$	$0.886^{\pm0.000}$	$0.976^{\pm0.000}$	0.001	$29.66^{\pm0.512}$	45.53	1.26
EDGE (Tseng et al., 2023)	$0.457^{\pm0.006}$	$0.730^{\pm0.010}$	$0.819^{\pm0.009}$	$7.498^{\pm0.485}$	$27.49^{\pm0.119}$	35.44	9.23
POPDG (Luo et al., 2024)	$0.462^{\pm0.006}$	$0.681^{\pm0.005}$	$0.782^{\pm0.004}$	$5.727^{\pm0.485}$	$28.15^{\pm0.141}$	34.99	2.77
MOSPA (Xu et al., 2025)	$0.585^{\pm0.013}$	$0.788^{\pm0.009}$	$0.885^{\pm0.006}$	$5.160^{\pm0.875}$	$33.50^{\pm0.583}$	50.29	3.69
MoMask++ (Guo et al., 2025)	$0.415^{\pm0.006}$	$0.656^{\pm0.008}$	$0.775^{\pm0.009}$	$3.909^{\pm0.331}$	$28.39^{\pm0.787}$	38.24	14.11
MOMO (Ours)	$\mathbf{0.623}^{\pm0.004}$	$\mathbf{0.824}^{\pm0.003}$	$\mathbf{0.920}^{\pm0.004}$	$\mathbf{0.857}^{\pm0.082}$	$\mathbf{28.97}^{\pm0.636}$	**42.60**	**1.66**

5.2 Ablation Study

We conduct several ablation experiments to evaluate each component's contribution, including the type of sound source, audio feature and network module. Each variant is trained with the totally same training schedule, and the results are summarized in Tab. 3.

Sound Source. This group evaluates the effect of sound source. Replacing the predicted DoA with source positions (②) leads to a clear performance drop. Removing the sound source condition entirely (①) leads to a drastic decrease in R-precision, highlighting the importance of spatial cues for audio-motion alignment. As expected, using the ground-truth DoA (③) achieves the best performance, confirming the importance of accurate head-centric spatial cues for generating spatially and semantically aligned motions.

Audio Feature. This group studies the effect of different audio representations by replacing the frame-wise CLAP embedding in the full model with either hand-crafted acoustic features (④) or a global CLAP feature pooled over the entire audio segment (⑤). Both alternatives lead to obvious degraded performance. This suggests that frame-wise CLAP features are important for providing better representation for audio content, which are crucial for generating high-quality motions.

Network Module. This group investigates our proposed conditioning modules. Removing the *Correction Strength Predictor* leads to degradation in all metrics, indicating that adaptively estimating the correction strength is important for both motion quality and audio-motion alignment (⑦).

Table 3: **Ablation on the DySAM Dataset,** including the types of sound source and audio features, Frame-wise Correction (Frame Corr.) , and Correction Strength Predictor (Strength Pred.).

Item	Ablated Component	Method	Sound Source	Audio Feature	Frame Corr.	Strength Pred.	FID ↓	R-precision ↑			Skating ↓ Ratio (%)
								Top1	Top2	Top3	
		Real Motion						$0.720^{\pm 0.000}$	$0.886^{\pm 0.000}$	$0.976^{\pm 0.000}$	1.26
①	Sound Source	w/o Sound Source	∅	Frame-wise CLAP	✓	✓	$1.334^{\pm 0.236}$	$0.359^{\pm 0.010}$	$0.593^{\pm 0.021}$	$0.722^{\pm 0.011}$	2.25
②		World Position	Position	Frame-wise CLAP	✓	✓	$2.222^{\pm 0.318}$	$0.549^{\pm 0.012}$	$0.790^{\pm 0.008}$	$0.895^{\pm 0.005}$	2.77
③		GT DoA	GT DoA	Frame-wise CLAP	✓	✓	$\mathbf{0.726}^{\pm 0.056}$	$\mathbf{0.673}^{\pm 0.006}$	$0.839^{\pm 0.003}$	$\mathbf{0.936}^{\pm 0.001}$	2.32
④	Audio Feature	Trad. Feat.	Pred DoA	Trad. Feat.	✓	✓	$2.343^{\pm 0.160}$	$0.547^{\pm 0.016}$	$0.768^{\pm 0.011}$	$0.888^{\pm 0.011}$	4.68
⑤		Global CLAP	Pred DoA	Global CLAP	✓	✓	$6.693^{\pm 0.726}$	$0.569^{\pm 0.012}$	$0.767^{\pm 0.007}$	$0.860^{\pm 0.011}$	3.28
⑥	Network Module	w/o Corr.	Pred DoA	Frame-wise CLAP			$4.458^{\pm 0.328}$	$0.556^{\pm 0.011}$	$0.772^{\pm 0.010}$	$0.896^{\pm 0.006}$	3.71
⑦		w/o Strength Pred.	Pred DoA	Frame-wise CLAP	✓		$1.530^{\pm 0.185}$	$0.586^{\pm 0.009}$	$0.813^{\pm 0.004}$	$0.917^{\pm 0.002}$	2.41
⑧		Trans. Dec.	Pred DoA	Frame-wise CLAP	✓	✓	$0.906^{\pm 0.138}$	$0.627^{\pm 0.008}$	$\mathbf{0.853}^{\pm 0.005}$	$0.915^{\pm 0.004}$	1.81
⑨		**Ours (full model)**	Pred DoA	Frame-wise CLAP	✓	✓	$0.857^{\pm 0.082}$	$0.623^{\pm 0.005}$	$0.824^{\pm 0.003}$	$0.920^{\pm 0.004}$	**1.66**

Table 5: **Ablation of the sound source localization network.** We evaluate the DoA representation and temporal architecture on the DySAM validation set. All variants are trained using the same training data. Reported numbers are the full-frame angular errors (°) between the predicted and ground-truth source directions.

	Method	DoA Representation	Temporal Module	DySAM Val. Angular Error (°) ↓
①	Using Spherical	Spherical (azimuth, elevation)	Bi-GRU	18.52
②	Using GRU	Unit direction vector	Bi-GRU	16.76
③	Using Transformer	Unit direction vector	Transformer encoder	**12.87**

When the *Frame-wise Residual Correction* is further removed (⑥), the performance drops more significantly, suggesting that simply concatenating condition tokens and relying on self-attention cannot explicitly align motion with frame-wise conditions. We further ablate the main backbone by replacing it with a Transformer decoder (⑧), which leads to a clear performance drop.

Ablation Study on Sound Source Localization Network. We evaluate the architectural design of the SSL network, focusing on the DoA representation and temporal modeling module. All variants follow the same training schedule and are evaluated by frame-wise angular error on the DySAM validation set. As shown in Table 5, unit direction vectors outperform spherical coordinates, while the Transformer encoder further improves localization accuracy over Bi-GRU, supporting its effectiveness in temporal spatial-audio modeling. In addition, Table 4 shows that incorporating the DySAL training split reduces the angular error on the DySAL validation set, indicating that its frame-wise annotations provide useful spatial supervision for the diverse acoustic events covered by the dataset. The SSL network used in MOMO adopts the unit-direction-vector representation, the Transformer encoder, and joint training on DySAM and DySAL.

Table 4: **Effect of the auxiliary DySAL training data.** We report the angular error on the DySAL validation set.

SSL Training Data	DySAL Val. Angular Error (°) ↓
DySAM	48.92
DySAM + DySAL	**36.62**

6 Conclusion

This paper introduces the task of dynamic spatial-audio-driven human motion generation. We present DySAM, the first dataset and benchmark for this task containing synchronized binaural audio, 3D human motion, and moving sound source trajectories, together with DySAL, an auxiliary dataset that provides diverse acoustic events and frame-wise spatial annotations for dynamic spatial audio perception. We further propose MOMO, a diffusion-based framework that explicitly leverages frame-wise audio-spatial conditions to generate high-quality human motions in response to dynamic spatial audio. Extensive experiments demonstrate that MOMO produces natural and physically plausible motions with improved alignment to dynamic spatial audio. *Limitations and Future Work:* Our current setting considers a single prominent moving sound source. Extending the framework to multiple simultaneous sources would require jointly reasoning about source separation, localization, and their potentially different behavioral relevance. We leave this challenging setting for future research and evaluation.

AI USE STATEMENT

In this work, we used generative AI tools for assisting with code implementation, editing, completion, and debugging. We have not used generative AI tools for generating synthetic datasets, developing theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for or assisting in writing proofs, proposing or refining research hypotheses, designing or providing feedback on research methodology or experiments, assisting with translation, cleaning or reformatting datasets, supporting qualitative or thematic data analysis, or interpreting experimental results, and the remaining required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for drafting and editing parts of the manuscript, improving grammar, clarity, organization, and readability, and assisting with general software code creation or editing. We have reviewed all AI-assisted work. All AI-assisted manuscript text was carefully reviewed and revised by the authors to ensure factual and technical accuracy and to preserve the authors' intended meaning. All AI-assisted code was manually inspected, executed, tested, and validated by the authors before being used in the experiments. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

To support reproducibility, we provide detailed descriptions of the dataset construction, model architecture, implementation details, training procedure, and evaluation protocols in the main paper and appendix. Implementation details are included in the supplementary material. Upon acceptance, we will publicly release the datasets, source code, pretrained models, and data-processing scripts to facilitate reproduction and future research.

REFERENCES

Simon Alexanderson, Rajmund Nagy, Jonas Beskow, and Gustav Eje Henter. Listen, denoise, action! audio-driven motion synthesis with diffusion models. *ACM Transactions on Graphics (TOG)*, 42 (4):1–20, 2023.

Tenglong Ao, Qingzhe Gao, Yuke Lou, Baoquan Chen, and Libin Liu. Rhythmic gesticulator: Rhythm-aware co-speech gesture synthesis with hierarchical neural embeddings. *ACM Transactions on Graphics (TOG)*, 41(6):1–19, 2022.

Tenglong Ao, Zeyi Zhang, and Libin Liu. Gesturediffuclip: Gesture diffusion model with clip latents. *ACM Transactions on Graphics (TOG)*, 42(4):1–18, 2023.

Joao Pedro Araujo, Yanjie Ze, Pei Xu, Jiajun Wu, and C Karen Liu. Retargeting matters: General motion retargeting for humanoid motion tracking. *arXiv preprint arXiv:2510.02252*, 2025.

Sebastian Böck and Gerhard Widmer. Maximum filter vibrato suppression for onset detection. In *Proc. of the 16th Int. Conf. on Digital Audio Effects (DAFx). Maynooth, Ireland (Sept 2013)*, volume 7, pp. 4. Citeseer, 2013.

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

Ziyang Chen, Shengyi Qian, and Andrew Owens. Sound localization from motion: Jointly learning sound direction and camera rotation. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7863–7874. IEEE, 2023.

Rishabh Dabral, Muhammad Hamza Mughal, Vladislav Golyanik, and Christian Theobalt. Mofusion: A framework for denoising-diffusion-based motion synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9760–9770, 2023.

Steven Davis and Paul Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE transactions on acoustics, speech, and signal processing*, 28(4):357–366, 1980.

Deity Microphones. Pr-2. https://deitymic.com/products/pr-2/, 2026. URL https://deitymic.com/products/pr-2/. Accessed: 2026-09-18.

Zhiyang Dou, Xuelin Chen, Qingnan Fan, Taku Komura, and Wenping Wang. C· ase: Learning conditional adversarial skill embeddings for physics-based characters. In *SIGGRAPH Asia 2023 Conference Papers*, pp. 1–11, 2023.

Daniel PW Ellis. Beat tracking by dynamic programming. *Journal of New Music Research*, 36(1): 51–60, 2007.

Guillermo García-Barrios, Daniel Aleksander Krause, Archontis Politis, Annamaria Mesaros, Juana M Gutiérrez-Arriola, and Rubén Fraile. Binaural source localization using deep learning and head rotation information. In *2022 30th European Signal Processing Conference (EUSIPCO)*, pp. 36–40. IEEE, 2022.

Shiry Ginosar, Amir Bar, Gefen Kohavi, Caroline Chan, Andrew Owens, and Jitendra Malik. Learning individual styles of conversational gesture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3497–3506, 2019.

Peter Grosche, Meinard Müller, and Frank Kurth. Cyclic tempogram—a mid-level tempo representation for musicsignals. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5522–5525. IEEE, 2010.

Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2motion: Conditioned generation of 3d human motions. In *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 2021–2029, 2020.

Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5152–5161, 2022.

Chuan Guo, Inwoo Hwang, Jian Wang, and Bing Zhou. Snapmogen: Human motion generation from expressive texts. In *Advances in Neural Information Processing Systems*, 2025.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. Motiongpt: Human motion as a foreign language. *Advances in Neural Information Processing Systems*, 36:20067–20079, 2023.

Korrawe Karunratanakul, Konpat Preechakul, Supasorn Suwajanakorn, and Siyu Tang. Guided motion diffusion for controllable human motion synthesis. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2151–2162. IEEE, 2023.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Buyu Li, Yongchi Zhao, Shi Zhelun, and Lu Sheng. Danceformer: Music conditioned 3d dance generation with parametric motion transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 1272–1279, 2022.

Ronghui Li, Junfan Zhao, Yachao Zhang, Mingyang Su, Zeping Ren, Han Zhang, Yansong Tang, and Xiu Li. Finedance: A fine-grained choreography dataset for 3d full body dance generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10234–10243, 2023.

Ronghui Li, YuXiang Zhang, Yachao Zhang, Hongwen Zhang, Jie Guo, Yan Zhang, Yebin Liu, and Xiu Li. Lodge: A coarse to fine diffusion network for long dance generation guided by the characteristic dance primitives. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1524–1534, 2024.

Ruilong Li, Shan Yang, David A Ross, and Angjoo Kanazawa. Ai choreographer: Music conditioned 3d dance generation with aist++. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13401–13412, 2021.

Andrew Luo, Yilun Du, Michael Tarr, Josh Tenenbaum, Antonio Torralba, and Chuang Gan. Learning neural acoustic fields. *Advances in Neural Information Processing Systems*, 35:3165–3177, 2022.

Zhengyi Luo, Ye Yuan, Tingwu Wang, Chenran Li, Fernando Castañeda, Sirui Chen, Zi-Ang Cao, Jiefeng Li, David Minor, Qingwei Ben, et al. Sonic: Supersizing motion tracking for natural humanoid whole-body control. *Science Robotics*, 11(117):eaed4592, 2026.

Zhenye Luo, Min Ren, Xuecai Hu, Yongzhen Huang, and Li Yao. Popdg: Popular 3d dance generation with popdanceset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26984–26993, 2024.

Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, and Michael J Black. Amass: Archive of motion capture as surface shapes. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5442–5451, 2019.

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

Anna Min, Ziyang Chen, Hang Zhao, and Andrew Owens. Supervising sound localization by in-the-wild egomotion. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23936–23946. IEEE, 2025.

Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3D hands, face, and body from a single image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 10975–10985, 2019.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4172–4182. IEEE, 2023.

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

Anton Ratnarajah, Shi-Xiong Zhang, Meng Yu, Zhenyu Tang, Dinesh Manocha, and Dong Yu. Fast-rir: Fast neural diffuse room impulse response generator. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 571–575. IEEE, 2022.

Justin Salamon, Christopher Jacoby, and Juan Pablo Bello. A dataset and taxonomy for urban sound research. In *Proceedings of the 22nd ACM International Conference on Multimedia*, pp. 1041–1044, 2014.

Christian Schörkhuber and Anssi Klapuri. Constant-q transform toolbox for music processing. In *7th sound and music computing conference, Barcelona, Spain*, pp. 3–64. SMC, 2010.

Li Siyao, Weijiang Yu, Tianpei Gu, Chunze Lin, Quan Wang, Chen Qian, Chen Change Loy, and Ziwei Liu. Bailando: 3d dance generation by actor-critic gpt with choreographic memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11050–11059, 2022.

Taoran Tang, Jia Jia, and Hanyang Mao. Dance with melody: An lstm-autoencoder approach to music-oriented dance synthesis. In *Proceedings of the 26th ACM international conference on Multimedia*, pp. 1598–1606, 2018.

The Sound Professionals. MS-TFB-2 Master Series Ultra-Low-Noise In-Ear Binaural Microphones. `https://soundprofessionals.com/product/master-series-ultra-low-noise-in-ear-binaural-microphones-made-in-usa/`, 2026. URL `https://soundprofessionals.com/product/master-series-ultra-low-noise-in-ear-binaural-microphones-made-in-usa/`. Accessed: 2026-09-18.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pp. 5026–5033. IEEE, 2012.

Jonathan Tseng, Rodrigo Castellon, and Karen Liu. Edge: Editable dance generation from music. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 448–458, 2023.

Shuhei Tsuchida, Satoru Fukayama, Masahiro Hamasaki, and Masataka Goto. Aist dance video database: Multi-genre, multi-dancer, and multi-camera database for dance information processing. In *Proceedings of the 20th International Society for Music Information Retrieval Conference, ISMIR 2019*, pp. 501–510, Delft, Netherlands, November 2019.

Sung Jin Um, Dongjin Kim, Sangmin Lee, and Jung Uk Kim. Object-aware sound source localization via audio-visual scene understanding. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8342–8351. IEEE, 2025.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Vicon. Vicon motion capture systems, 2026. URL `https://www.vicon.com/`. Accessed: 2026-09-17.

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

Lixing Xiao, Shunlin Lu, Huaijin Pi, Ke Fan, Liang Pan, Yueer Zhou, Ziyong Feng, Xiaowei Zhou, Sida Peng, and Jingbo Wang. Motionstreamer: Streaming motion generation via diffusion-based autoregressive model in causal latent space. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10086–10096. IEEE, 2025.

Shuyang Xu, Zhiyang Dou, Mingyi Shi, Liang Pan, Leo Ho, Jingbo Wang, Yuan Liu, Cheng Lin, Yuexin Ma, Wenping Wang, and Taku Komura. Mospa: Human motion generation driven by spatial audio. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Youngwoo Yoon, Bok Cha, Joo-Haeng Lee, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. Speech gesture generation from the trimodal context of text, audio, and speaker identity. *ACM Transactions on Graphics (TOG)*, 39(6):1–16, 2020.

Yanjie Ze, Zixuan Chen, João Pedro Araújo, Zi ang Cao, Xue Bin Peng, Jiajun Wu, and C. Karen Liu. Twist: Teleoperated whole-body imitation system. *arXiv preprint arXiv:2505.02833*, 2025.

Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Yong Zhang, Hongwei Zhao, Hongtao Lu, Xi Shen, and Ying Shan. Generating human motion from textual descriptions with discrete representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14730–14740, 2023.

Yaqi Zhang, Di Huang, Bin Liu, Shixiang Tang, Yan Lu, Lu Chen, Lei Bai, Qi Chu, Nenghai Yu, and Wanli Ouyang. Motiongpt: Finetuned llms are general-purpose motion generators. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 7368–7376, 2024.

Wenlin Zhuang, Congyi Wang, Jinxiang Chai, Yangang Wang, Ming Shao, and Siyu Xia. Music2dance: Dancenet for music-driven dance generation. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 18(2):1–21, 2022.

A Dataset Details

A.1 Motion Capture Design

At the beginning of each clip, the dynamic sound source, i.e., a speaker, starts from one of four azimuthal regions relative to the actor: front-left, rear-left, rear-right, and front-right. This design enables us to capture diverse human reactions to sounds arriving from different directions. The trajectory of the sound source is designed to be either straight or curved. Moreover, the sound source either moves toward the actor and reaches the actor's position, or passes by without contact. These two movement patterns elicit fundamentally different responses. For example, an actor may react differently when a dog or a car rushes toward them than when it merely passes by.

We further define two reaction styles: insensitive and sensitive, to represent different levels of responsiveness to the same acoustic event. For instance, when hearing an explosion, an insensitive actor may remain largely unreactive, whereas a sensitive actor may immediately move away from the sound source. The two reaction styles are approximately balanced in the dataset.

A.2 Motion Capture and Audio Recording Setup

The data are collected in a typical indoor environment with natural room reverberation. The motion capture system consists of 20 Vicon cameras installed around the capture space, recording both the human body motion and the sound source trajectory at 120 Hz.

For the moving sound source, we use a cylindrical omni-directional speaker to play the audio, ensuring stable sound radiation in different directions as the speaker moves. For binaural audio recording, a high-sensitivity microphone is placed near each ear of the actor to capture the audio perceived by the actor as faithfully as possible. The two microphones are connected to a Deity PR-2 recorder (Deity Microphones, 2026), which is synchronized with the Vicon system in advance using timecode. This setup ensures accurate temporal alignment between the recorded binaural audio and the corresponding motion capture data.

A.3 Spatial Audio Scenarios in DySAM

The DySAM dataset covers 42 distinct audio types, corresponding to 109 unique audio sources, as listed in Tab. A1. Each audio type comprises one to six audio sources, and for each audio source 8 to 32 ten-second motion sequences are collected, yielding 2,336 (binaural audio, motion) pairs in total. These audio types span daily-life sounds including animal vocalizations, vehicles, environmental noises, alarms, musical instruments, and human speech.

Audio Type			
alarm	bark	bicycle_bell	bird
car_bidirect	car_fast	car_slow	cat
cough	drill	drill_laugh	engine_slow
firework	gun_hand	gun_machine	helicopter
horn_car	horn_train	horn_truck	insect_bees
insect_flying	insect_msqt	monster_roar	monster_step
motor_fast	motor_slow	plane	puppy
rainstorm	rockroll	sneezing	train
violin	voice_danger_female	voice_danger_male	voice_follow_female
voice_follow_male	voice_greet_female	voice_greet_male	voice_help_female
voice_help_male	witch		

Table A1: The 42 distinct audio types covered in DySAM, corresponding to 109 unique audio sources. Each audio source is accompanied by 8 to 32 ten-second motion sequences, and the dataset contains 2,336 (binaural audio, motion) pairs in total.

A.4 Audio Sources for DySAL

To ensure broad audio diversity, we collect raw audio materials from AIST (Tsuchida et al., 2019), ESC-50 (Piczak, 2015), and UrbanSound8K (Salamon et al., 2014). AIST mainly contains music

and dance-related audio with diverse rhythms, styles, and structures. ESC-50 provides environmental sounds from natural scenes, human activities, animal sounds, indoor and outdoor daily scenes, and mechanical devices, such as rain, sea waves, bird calls, coughing, footsteps, keyboard typing, bells, and vacuum cleaners. UrbanSound8K contains typical urban sounds, including air conditioners, car horns, children playing, dog barks, drilling, engine idling, gunshots, jackhammers, sirens, and street music.

By combining these sources, DySAL covers music, human activities, natural environments, animals, indoor and outdoor daily activities, transportation, mechanical construction, alarms, and urban public spaces. This diversity broadens the range of acoustic events available for training and evaluating sound source localization models under dynamic spatial-audio conditions.

A.5 Data Processing

For intuitively understand our data processing pipeline, we provide four visualization examples of the DySAM dataset in Fig. A1.

Motion processing. During capture, the actor wears a motion capture suit with multiple reflective markers attached to the body, represented by the green points in Fig. A1 The Vicon system exports the captured data as C3D files, where each file stores the world coordinates of all body markers for every frame. We use Mosh++ (Mahmood et al., 2019) to fit each C3D sequence into the SMPL-X parameters. The fitted SMPL-X motion is then canonicalized by rotating the human body in the initial frame to face the positive y-axis and move the center of the body to the origin. For training the motion generation model, we further convert the SMPL-X motion into the 272-dimensional motion representation following (Xiao et al., 2025).

Sound source location processing. To track the position of the sound source, i.e., the speaker, we attach eight reflective markers around the speaker (orange points in Fig. A1), which approximately correspond to the eight vertices of a cuboid. For each C3D sequence, we use the speaker markers in the initial frame as reference markers. We then estimate the speaker translation in each subsequent frame by solving the rigid alignment between the reference markers and the current-frame markers using singular value decomposition (SVD). The estimated translation is used as the global sound source location in our dataset, represented by the blue sphere in Fig. A1. In practice, some speaker markers may be missing in certain frames due to occlusion or tracking failure, making SVD-based estimation unreliable or infeasible. For these frames, we use Kalman filtering to smoothly interpolate and complete the missing sound source locations.

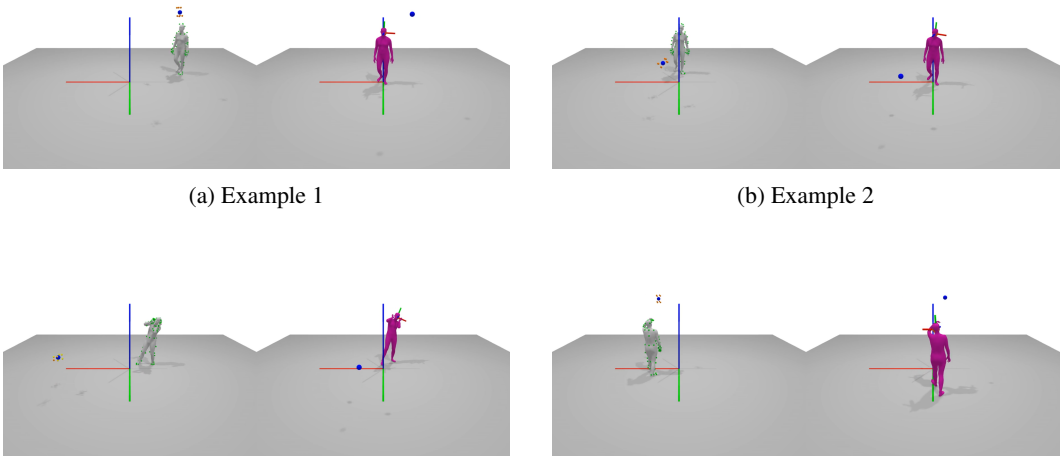
Head-centric DoA processing. To train the sound source localization network, we further compute the head-centric sound source location. Specifically, for each frame, we perform forward kinematics on the fitted SMPL-X body model to obtain the translation and rotation of the head joint, i.e., joint 15. We then transform the global sound source location into the head coordinate system using the estimated head pose, yielding the head-centric sound source location as the direction-related supervision signal.

B Implementation Details of the SSL Network

We provide additional implementation details of the sound source localization network used in Sec. 4.1. The SSL network estimates a head-centric direction of arrival from binaural audio. For every frame, it predicts a unit vector $\mathbf{d}_t \in \mathbb{R}^3$, which points from the listener's head toward the sound source and is expressed in the listener's head coordinate system.

Binaural feature extraction. We compute the short-time Fourier transform (STFT) of the left and right binaural signals using a $40\,\mathrm{ms}$ Hamming window, zero-padded to an FFT size of 2048, and a hop size of $33.3\,\mathrm{ms}$, yielding a frame rate of $30\,\mathrm{Hz}$ aligned with the motion sequence. We keep the lowest 256 frequency bins, corresponding to frequencies up to approximately $6\,\mathrm{kHz}$.

Let $L_{t,f}$ and $R_{t,f}$ denote the complex STFT coefficients of the left and right channels at time frame t and frequency bin f. The SSL input contains four channels: (1) the average magnitude spectrum, (2-3) the sine and cosine of the interaural phase difference (IPD), and (4) the interaural level difference (ILD). The first channel encodes the overall spectral magnitude, the second and third channels



(a) Example 1 (b) Example 2

(c) Example 3 (d) Example 4

Figure A1: Four examples of DySAM dataset. In each image, the left column shows the original C3D body markers and speaker markers, SMPL-X motion (gray body), and calculated speaker location; the right column shows the canonicalized SMPL-X motion (purple body) and the head coordinate system. The red, green, and blue axes denote the x, y, and z axes, respectively.

encode the IPD without phase-wrapping discontinuities, and the fourth channel provides the relative loudness between the two ears. The resulting binaural feature is a tensor $\mathbf{X} \in \mathbb{R}^{4\times300\times256}$, where every sequence is represented by 300 frames at 30 Hz to match the motion sequences. The extraction procedure is summarized in the Python pseudo-code below.

```python
def extract_binaural_features(audio):           # audio: (2, N), 10 s @ 48 kHz

    # per-ear STFT: 40 ms Hamming window, hop of 33.3 ms (30 frames/s)
    S_L = librosa.stft(audio[0])                # (F, T), F = 1025 bins
    S_R = librosa.stft(audio[1])                # (F, T)

    # per-ear magnitude and phase
    A_L, phi_L = np.abs(S_L), np.angle(S_L)     # (F, T)
    A_R, phi_R = np.abs(S_R), np.angle(S_R)     # (F, T)

    # four binaural channels, each of shape (F, T)
    X1 = (A_L + A_R) / 2                        # (1) average magnitude
    X2 = np.sin(phi_L - phi_R)                  # (2) IPD sine
    X3 = np.cos(phi_L - phi_R)                  # (3) IPD cosine
    eps = 1e-10                                 # avoid division by zero
    X4 = 20 * np.log10((A_L + eps) / (A_R + eps)) # (4) ILD

    # stack the four channels, keep the lowest 256 bins, align with motion
    X = np.stack([X1, X2, X3, X4])              # (4, F, T)
    X = X[:, :256, :]                           # (4, 256, T)
    return X
```

Network architecture. The SSL network consists of a convolutional spectral encoder followed by a Transformer-based temporal encoder. The convolutional front-end contains three convolutional blocks. Each block applies a 3×3 convolution followed by a ReLU activation and max pooling along the frequency axis with stride 4. This progressively reduces the frequency dimension from 256 to 64, 16, and 4, while preserving the temporal resolution.

After the convolutional blocks, the feature map is flattened at each time frame and projected into a C-dimensional latent space, where $C = 128$. We add a learnable absolute positional encoding to the frame-wise tokens before feeding them into the temporal encoder.

Different from (García-Barrios et al., 2022), which adopts a GRU for temporal modeling, we use a Transformer encoder. The encoder has $L = 4$ layers, 8 attention heads, and a feed-forward hidden

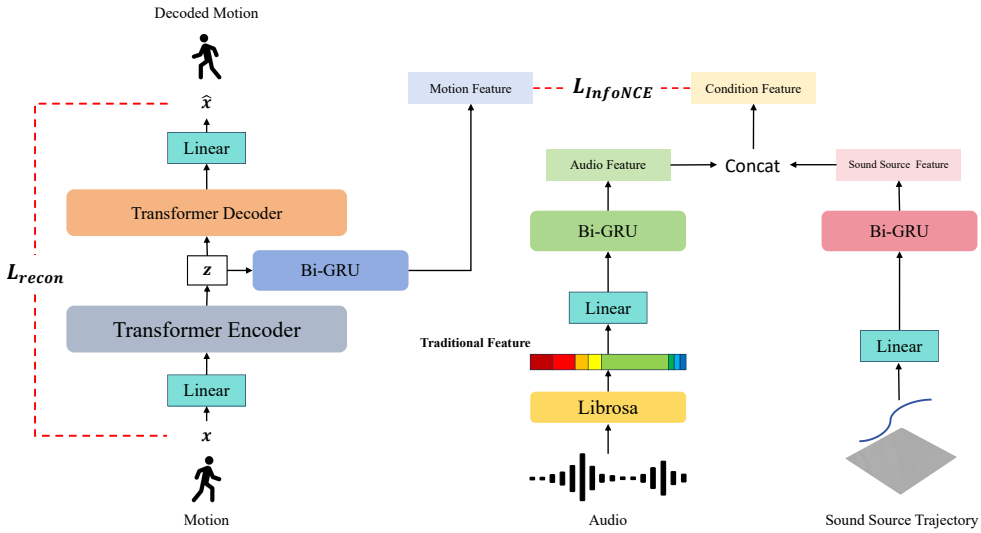


Figure C2: The evaluator framework consists of a motion autoencoder co-trained with Bi-GRU feature extractors. A reconstruction loss $\mathcal{L}_{\text{recon}}$ is applied between the ground-truth motion $\mathbf{x}$ and the decoded motion $\hat{\mathbf{x}}$ produced by the motion autoencoder. Two Bi-GRU encoders extract the audio feature and the sound source feature from the binaural audio and the sound source trajectory, respectively, and the concatenation of the two features forms the condition latent. Additionally, an InfoNCE loss $\mathcal{L}_{\text{InfoNCE}}$ operates on the extracted condition features and motion features.

dimension of 256. We use pre-normalization Transformer blocks for stable training. The self-attention mechanism enables each frame to aggregate binaural evidence from the entire sequence, which helps resolve ambiguous localization cues caused by reverberation or transient source motion.

The Transformer outputs one token per frame. Each token is passed through two fully connected layers to regress a three-dimensional vector, which is then L2-normalized to obtain the predicted DoA: $\mathbf{d}_t = \frac{\mathbf{o}_t}{\|\mathbf{o}_t\|_2}$, where $\mathbf{o}_t \in \mathbb{R}^3$ is the raw output vector before normalization.

Training objective. The SSL network is trained using head-relative source direction labels from DySAM and DySAL. The ground-truth direction vector at frame t is normalized to unit length and denoted as $\bar{\mathbf{d}}_t$. We optimize the cosine-distance loss: $\mathcal{L}_{\text{ssl}} = \frac{1}{T}\sum_{t=1}^{T}\left(1 - \mathbf{d}_t \cdot \bar{\mathbf{d}}_t\right)$. This objective is monotonically related to the angular error between the predicted and ground-truth DoA vectors and directly supervises the directional information used by the downstream motion generator.

During motion generation, the SSL network parameters are frozen. Given only binaural audio, the network produces the frame-wise DoA sequence $\mathbf{D} = \{\mathbf{d}_1, \ldots, \mathbf{d}_T\}$, which is used as the spatial condition of the diffusion model.

C Evaluator Implementation Details

Inspired by the evaluator design from (Xu et al., 2025), our evaluator is a motion–condition matching network , which measures how well a motion sequence corresponds to a given spatial audio condition. It maps both modalities into a shared latent space, in which the R-precision metrics are computed by nearest-neighbor retrieval and the FID is computed between the feature distributions of the ground-truth and the generated motions. Different from (Xu et al., 2025), whose condition only contains the acoustic content, our condition additionally involves the trajectory of the sound source, so that the evaluator is sensitive to the spatial correspondence between the moving sound source and the human reaction.

Network architecture. As illustrated in Fig. C2, the evaluator consists of a motion autoencoder and three bidirectional GRUs (Bi-GRUs), each with 4 layers and a hidden size of 512. The feature of a Bi-GRU is obtained by concatenating its last forward and backward hidden states, which yields a 1024-dimensional feature vector.

The motion autoencoder follows a transformer encoder–decoder structure. The input motion sequence $\mathbf{x} \in \mathbb{R}^{T\times 272}$ is first projected to a 512-dimensional latent space and added with a positional encoding, and is then encoded by a 4-layer transformer encoder with 4 attention heads and a 1024-dimensional feed-forward layer. The encoder output is averaged over time into a single latent vector and replicated along the temporal axis, which serves as the query of a transformer decoder of the same configuration attending to the encoder output. A final linear layer maps the decoder output back to the 272-dimensional motion space to produce the reconstructed motion $\hat{\mathbf{x}}$. The output of the motion encoder is further encoded by a Bi-GRU to obtain the motion feature.

The condition branch consists of an audio encoder and a sound source encoder. The audio encoder takes the traditional features described in Appendix E. The sound source encoder takes the absolute position of the sound source in the world coordinate system $\mathbf{s} \in \mathbb{R}^{T\times 3}$, projects it to 64 dimensions and encodes it with a Bi-GRU. The resulting audio and sound source features are concatenated and projected by a linear layer into the condition feature, whose dimension matches that of the motion feature.

Training objective and details. The evaluator is optimized with two objectives. The motion autoencoder is supervised by a reconstruction loss $\mathcal{L}_{\mathrm{recon}}$, which encourages the latent space to preserve the motion content. The condition and motion representations are aligned through contrastive learning using the InfoNCE loss $\mathcal{L}_{\mathrm{InfoNCE}}$ (van den Oord et al., 2018). During training, we randomly zero the sound source feature of an entire sequence with a probability of 0.3, producing audio-only conditions. This enables the evaluator to also apply to settings where the sound source trajectory is unavailable, while the audio condition is always retained. We train the evaluator for up to 1,000 epochs using the Adam optimizer (Kingma & Ba, 2014) with a learning rate of 5×10^{-5} and a batch size of 32. The evaluator is trained on all sequences of DySAM.

D Evaluation Metrics

Fréchet Inception Distance (FID). The FID is computed between the motion features of the ground-truth motions in the test set and those of the generated motions, where both sets of features are extracted by the frozen motion encoder of the evaluator. Gaussian distributions are fitted to the two feature sets, and the Fréchet distance between them serves as a metric for assessing the quality and fidelity of the synthesized motions.

R-precision. We compute the R-precision within batches of 32 sequences of the test set, following the protocol of (Guo et al., 2022). For each batch, the conditions and the motion features are L2-normalized, and the cosine similarities between all condition–motion pairs are ranked, where the pool of each condition consists of its matched motion together with the other 31 in-batch motions as distractors. The R-precision is then obtained by determining the probability that the matched motion appears among the top-1, top-2 and top-3 ranked positions, and is averaged over all the batches. The metric is computed in both the condition-to-motion and the motion-to-condition retrieval directions. The query condition used in our experiments consists of the ground-truth binaural audio and the ground-truth sound source trajectory, independently of the condition adopted by each generator, so that all compared methods are evaluated against the same query set. The Real Motion row in the main tables corresponds to using the ground-truth motions as the retrieval target.

Skating Ratio. We compute the skating ratio from recovered world-space foot joints, using the left and right foot joints indexed as 10 and 11 in the 22-joint SMPL protocol. For each foot, we compute its horizontal velocity on the ground plane at 30 Hz and smooth it with a 5-frame window. A foot is considered *planted* if it remains within 5 cm above the floor in adjacent frames and its smoothed horizontal speed is below 0.30 m/s. We count a valid contact frame as skating only when neither foot is planted. Airborne frames, where both feet are above 5 cm, and penetration frames, where any foot is below the floor plane, are excluded from the skating ratio and reported separately as float and penetration ratios. The procedure is summarized as follows.

```python
def skating_ratio(world_joints):                # (T, 22, 3) Z-up joints

    feet = world_joints[:, [10, 11], :]         # (T, 2, 3) left / right foot
    height = feet[:, :, 2]                      # (T, 2) foot height
    vel = calculate_feet_vel(feet)              # (T-1, 2)
    vel = smooth(vel, window=5)                 # (T-1, 2)

    # a foot is planted when it stays near the floor and barely moves
    h = np.maximum(height[:-1], height[1:])     # (T-1, 2) max of two frames
    planted_feet = (h < 0.05) & (vel < 0.30)          # (T-1, 2)
    on_floor_feet = (h >= 0) & (h < 0.05)             # (T-1, 2) valid contact

    # a contact frame skates when neither foot is planted
    planted_frame = planted_feet.any(axis=1)            # (T-1,) any foot planted
    on_floor_frame = on_floor_feet.any(axis=1)          # (T-1,) any foot on floor
    skating_frame = ~planted_frame & on_floor_frame     # (T-1,) skating frame
    return skating_frame.sum() / on_floor_frame.sum()         # ratio over contact frames
```

Our skating ratio formulation is modified from GMD (Karunratanakul et al., 2023), which marks a frame as skating when either foot is close to the ground and moves faster than $0.50\,\text{m/s}$. This can falsely flag frames where one foot is stably planted while the other foot briefly moves or drags near the floor, since the original metric evaluates each foot independently and ignores whether the body has a valid support foot. In contrast, our metric detects the absence of any stable support foot. This support-foot criterion avoids counting normal swing-foot motion as skating when the other foot stably supports the body.

To verify the metric behavior, we evaluate both definitions on the ground-truth motions of DySAM. As shown in Tab. D2, the original formulation reports a skating ratio of 13.9% on ground-truth data, which is inconsistent with the fact that these motions are captured from real humans and should contain little foot skating. In contrast, our formulation gives a much lower ratio of 1.26%, indicating that the revised metric better reflects actual foot skating artifacts rather than false positives caused by normal foot motion or representation noise.

Metric Definition	Skating Ratio on GT ↓
GMD formulation (Karunratanakul et al., 2023)	13.90%
Ours	1.26%

Table D2: Skating ratio computed on the ground-truth motions of DySAM using different metric definitions.

E Traditional Audio Feature Set

Following MOSPA (Xu et al., 2025), our acoustic feature set primarily includes *Mel-frequency cepstral coefficients (MFCCs), MFCC delta, constant-Q chromagram, short-time Fourier transform (STFT) chromagram, onset strength, tempogram, and beats* (Davis & Mermelstein, 1980; Schörkhuber & Klapuri, 2010; Grosche et al., 2010; McFee et al., 2015; Böck & Widmer, 2013; Ellis, 2007). In addition, we incorporate the root mean square (RMS) energy E_{rms} of the audio (McFee et al., 2015), and define the active frames as $F_{active} = E_{rms} > 0.01$ to capture distance-related audio information.

The full list of audio features and their corresponding dimensions are listed in Tab. E3. The audio is sampled at 48,000 Hz. When computing frame-level audio features with Librosa, we set the FFT window length to 2048, the hop length to 1600, and use a Hann window, yielding audio features with a frame rate of 30 Hz aligned with the motion sequences. The final shape of the audio features for a single ear is $\mathbb{R}^{T\times 335}$. We concatenate the features from the two ears along the feature dimension, obtaining the audio condition $\mathbf{a} \in \mathbb{R}^{T\times 670}$ for the evaluator, where $T = 300$ denotes the number of frames in each motion sequence.

Feature Name	Dimension
MFCC	20
MFCC delta	20
constant-Q chromagram	12
STFT chromagram	12
onset strength	1
tempogram	267
one-hot beats	1
RMS energy	1
active frames	1

Table E3: The extracted audio features for a single ear comprise a 335-dimensional vector. The total dimension of the binaural audio feature vector $\mathbf{a}$ is 670.

F Visualization of Reaction Genre Control

To evaluate the controllability of MOMO with respect to the reaction genre, we compare generated motions under two genre labels, where label 1 denotes a sensitive reaction and label 0 denotes an insensitive reaction. As shown in Fig. F3, for three representative audio examples, we keep the binaural audio input unchanged and only switch the genre label during generation.

MOMO produces clearly different responses for the two labels. Motions generated with label 1 show larger amplitudes and stronger avoidance reactions, whereas those with label 0 are more restrained and subtle. These results demonstrate that MOMO can effectively control the reaction intensity and expressiveness through the genre condition.

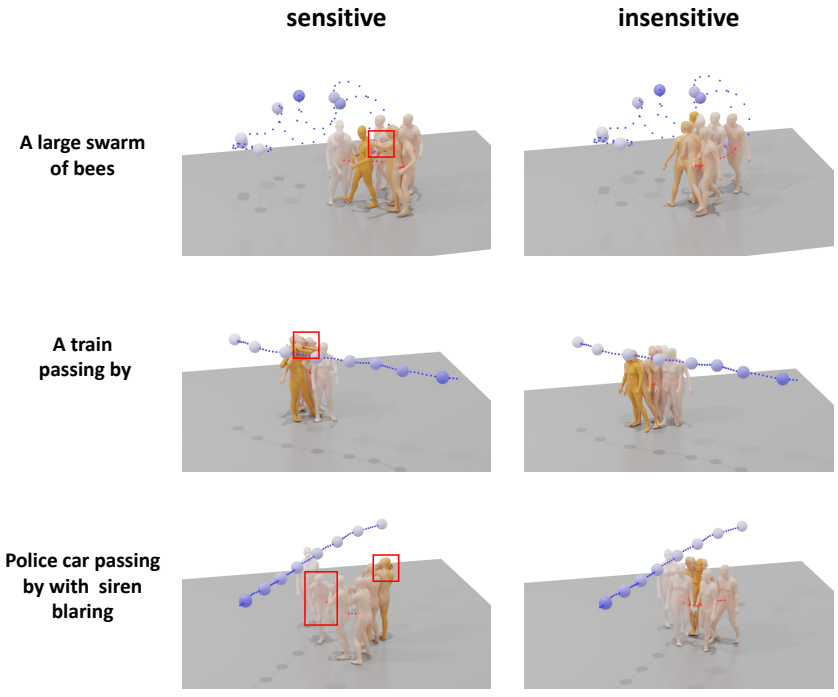


Figure F3: Qualitative comparison of reaction genre control on three audio examples. Given the same binaural audio, label 1, i.e., sensitive, produces stronger and larger-amplitude reactions than label 0, i.e., insensitive, demonstrating effective genre-based motion control.

G Robot Demonstrations

To investigate the applicability of dynamic spatial-audio-driven motion generation to embodied agents, we provide both simulated and real-world robot demonstrations. These demonstrations serve as preliminary evidence that the generated motions can be transferred to robotic platforms, suggesting the potential of our task formulation for future embodied deployment.

For the simulated robot demonstration, we retarget the generated SMPL-X motion parameters to a humanoid robot using GMR (Araujo et al., 2025). A pretrained TWIST controller (Ze et al., 2025) is then used to track the retargeted motion and drive the robot in the MuJoCo environment (Todorov et al., 2012), as shown in Fig. G4.

For the real-world robot demonstration, we use SONIC (Luo et al., 2026) as a general-purpose motion controller. The generated motion is fed into SONIC as the reference motion, and the real robot performs imitation based on this reference. The corresponding real-world demonstration is provided in the supplementary video. These results indicate that dynamic spatial audio-to-motion generation may serve as a useful interface for future robots to produce responsive behaviors toward moving sound sources. Please refer to our robot demonstration videos in the supplementary material.

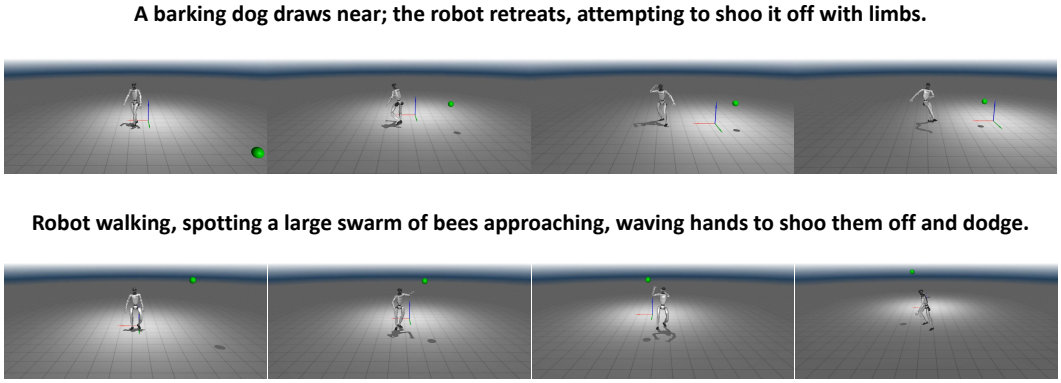


Figure G4: Simulated robot demonstration in MuJoCo. The generated SMPL-X motion is retargeted to a humanoid robot using GMR and tracked by a pretrained TWIST controller.

H User Study

We conduct a user study to evaluate the perceptual quality of generated motions. The study focuses on two aspects: (1) **motion realism and fluency**, which measures whether the generated motion looks natural, realistic, and temporally smooth; and (2) **alignment with dynamic spatial audio**, which measures whether the motion response is consistent with the audio content and the movement of the sound source.

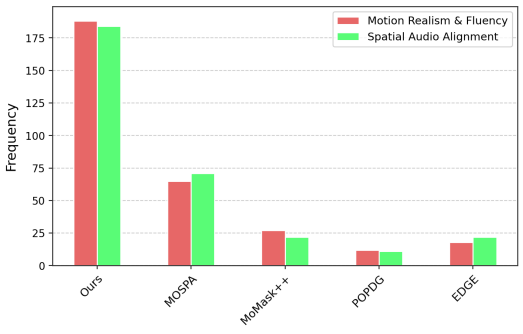


Figure H5: User study results.

A total of 30 participants take part in the study. We provide 10 audio cases for evaluation. For each case, participants are shown six videos: one ground-truth motion video as the reference, and five generated motion videos from different methods, including MOMO(Ours), MOSPA (Xu et al., 2025), MoMask++ (**?**), POPDG (Luo et al., 2024), and EDGE (Tseng et al., 2023). The order of the generated videos is randomized to reduce potential bias. The user study interface is shown in Fig. H6.

For each audio case, participants watch the videos with the corresponding dynamic spatial audio and select the best generated result under each evaluation aspect. Fig. H5 shows that MOMO is selected most frequently in both motion realism and fluency, and alignment with dynamic spatial

audio, outperforming all baseline methods. This indicates that MOMO generates motions that are not only more natural and fluent, but also better aligned with dynamic spatial-audio cues.

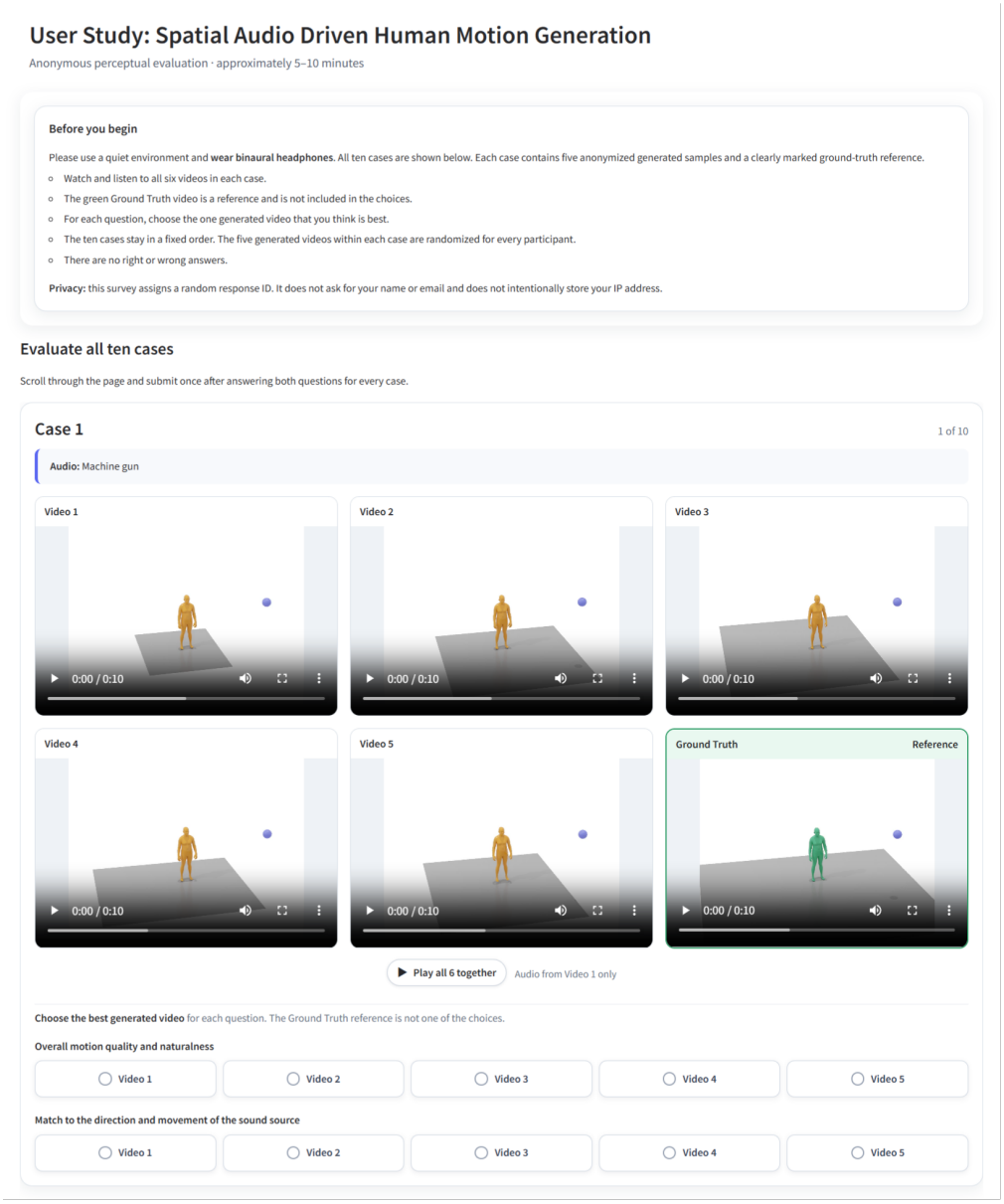


Figure H6: Screenshot of the user study interface.