FlowerDance: MeanFlow for Efficient and Refined 3D Dance Generation

Kaixing Yang[1*], Xulong Tang[4*], Ziqiao Peng[1*], Xiangyue Zhang[3], Chubin Chen[2], Xukun Zhou[1], Puwei Wang[1†], Hongyan Liu[2†], and Jun He[1†]

[1] Renmin University of China, Beijing, China
{yangkaixing,pengziqiao,xukun_zhou,wangpuwei,hejun}@ruc.edu.cn
[2] Tsinghua University, Beijing, China
liuhy@sem.tsinghua.edu.cn trubeeen@gmail.com
[3] Wuhan University, Wuhan, China
xiangyuezhang@whu.edu.cn
[4] Malou Tech Inc, Plano, Texas, USA
xulong.tang@maloutech.com

Abstract. Music-to-dance generation aims to translate auditory signals into expressive human motion. Despite promising progress, existing methods remain underexplored in achieving high-quality generation under a strict efficiency budget, due to (i) generative strategies that support high-fidelity few-step sampling, and (ii) per-step model architectures that are optimized for efficient yet refined long-horizon motion synthesis, thereby degrading downstream user experience in terms of responsiveness and visual fidelity. Thus, we propose FlowerDance, which not only generates refined motion with physical plausibility and artistic expressiveness, but also achieves significant generation efficiency on inference speed and memory utilization. Specifically, FlowerDance combines MeanFlow with Physical Consistency Constraints, which enables high-quality motion generation with only a few sampling steps. Moreover, FlowerDance leverages a simple but efficient model architecture with BiMamba-based backbone and Channel-Level Cross-Modal Fusion, which generates long-horizon dance with efficient non-autoregressive manner. Meanwhile, FlowerDance supports motion editing, enabling users to interactively refine dance sequences. Extensive experiments on AIST++ and FineDance show that FlowerDance achieves state-of-the-art results in both motion quality and generation efficiency. Code is available at https://sun-happy-ykx.github.io/FlowerDance/.

Keywords: Digital Human · 3D Motion Generation · AI for Art

1 Introduction

Music-to-dance generation is a crucial task that translates auditory input into dynamic motion, with significant applications in virtual reality, choreography,

[*] Equal contribution.
[†] Corresponding authors.

Fig. 1: FlowerDance not only generates refined motion with physical plausibility and artistic expressiveness, but also achieves generation efficiency that surpasses that of state-of-the-art (SOTA) methods in inference speed and memory utilization.

and digital entertainment [26,27]. By automating this process, it enables deeper exploration of the intrinsic relationship between music and movement [49], while expanding possibilities for creative content generation [3,4,15]. Due to its broad impact, music-to-dance (M2D) generation has attracted increasing attention [24,37,38]. A broadly applicable 3D M2D system should excel in two aspects: **(1) Dance Quality**: producing physically plausible and artistically expressive motion, which improves user immersion and reduces the need for manual post-editing; and **(2) Generation Efficiency**: achieving high inference speed with a low memory utilization, which preserves computational headroom for high-fidelity 3D rendering and real-time human-computor interaction.

A substantial body of Music-to-Dance methods has emerged, spanning GANs, autoregressive models, and diffusion approaches. GAN-based methods [16,40,48] enhance perceptual naturalness but often degenerate into overly simple, repetitive motion. Autoregressive models [23,36–38,45–47,52] incur substantial inference latency due to strictly sequential decoding. Diffusion-based methods [17,25, 26,41,50] achieve strong fidelity but typically require many denoising steps during inference time—attributable to their curved sampling trajectories—resulting in heavy computational overhead. However, existing methods remain under-explored in achieving high-quality generation under a strict efficiency budget, due to (i) generative strategies that support high-fidelity few-step sampling, and (ii) per-step model architectures that are optimized for efficient yet refined long-horizon motion synthesis, thereby degrading downstream user experience in terms of responsiveness and visual fidelity.

To this end, we present FlowerDance, a co-designed framework for high-quality long-horizon 3D dance generation under a strict inference budget. Our key observation is that end-to-end generation efficiency is jointly determined by (i) the number of sampling steps and (ii) the computational cost of each sampling step. Optimizing only one of these factors is insufficient in practice: few-step sampling yields limited wall-clock gains if each step is expensive, while a lightweight backbone alone remains slow when many iterative sampling steps are required. FlowerDance addresses these two factors jointly. At the generative-

strategy level, we adopt a MeanFlow-based few-step formulation and introduce a Physical Consistency Constraint to stabilize few-step updates over large integration interval and preserve physical plausibility in 3D motion generation. At the model-architecture level, we design a lightweight non-autoregressive BiMamba backbone with Channel-Level Cross-Modal Fusion to reduce per-step latency and memory cost while maintaining music-dance alignment. This trajectory-level and per-step efficiency co-design enables FlowerDance to simultaneously achieve strong dance quality and substantial efficiency gains, as shown in Fig. 1. To support real-time, human-in-the-loop motion authoring under few-step sampling, we further introduce a time-decayed soft masking strategy that enables coherent motion editing without additional training.

Trajectory-level efficiency via generative strategy. To reduce sampling steps without sacrificing motion quality, FlowerDance combines MeanFlow with a Physical Consistency Constraint. From a choreography perspective, dance creation involves progressively shaping coarse movements into coherent motion under musical guidance, a process that naturally resonates with Flow Matching [28,29]. However, naive flow-based formulations model only instantaneous velocity, while few-step inference must perform large-interval ODE updates, leading to a mismatch between the training target and the actual sampling process and thus degraded generation quality. We therefore adopt MeanFlow, which predicts interval-averaged velocity and is better aligned with few-step sampling, enabling efficient generation with only a small number of steps. Nevertheless, MeanFlow alone is insufficient for 3D dance generation, because 3D human motion lies on a tightly constrained kinematic manifold, and under large-step updates, even small prediction errors can accumulate over time and propagate along the kinematic chain, leading to temporal jitter, root drift, foot sliding, and implausible articulations. To address this issue, we introduce a Physical Consistency Constraint (PCC), which first recovers motion from the predicted mean velocity and then regularizes it with reconstruction, motion-velocity, and 3D joint-position losses, thereby stabilizing optimization and pulling generation toward the human motion manifold. In addition, FlowerDance supports motion editing for human-in-the-loop interaction. Specifically, we apply a time-decayed soft mask during sampling, which gradually relaxes the constraints over time and alleviates the boundary discontinuities caused by hard masks in the few-step setting.

Per-step efficiency via model architecture. To reduce the computational and memory cost of each sampling step, FlowerDance adopts a lightweight non-autoregressive architecture with a Bidirectional Mamba (BiMamba) backbone and Channel-Level Cross-Modal Fusion. BiMamba is well suited to music-to-dance generation because its sequential inductive bias better captures local temporal dependencies than attention-only backbones, while retaining linear-time complexity $\mathcal{O}(n)$ compared with the $\mathcal{O}(n^2)$ cost of Transformers [42]. Its bidirectional design further improves temporal coherence and music-motion alignment, and the non-autoregressive formulation avoids the latency of sequential decoding while mitigating exposure bias in autoregressive-based methods [37,38,47] and boundary artifacts in iterative inpainting-based methods [30, 41]. Moreover,

because music and dance are naturally aligned in time, we replace conventional cross-attention with a parameter-free Channel-Level Cross-Modal Fusion mechanism. This lightweight design reduces memory and latency while improving generalization in the non-autoregressive setting.

In summary, our contributions in the music-to-dance generation task are as follows: (1) We present FlowerDance, a co-designed framework for high-quality long-horizon 3D dance generation under a strict inference budget. By jointly optimizing trajectory-level sampling efficiency and per-step inference efficiency, FlowerDance achieves state-of-the-art (SOTA) performance in both dance quality and generation efficiency on AIST++ [27] and FineDance [26] datasets. (2) We propose a MeanFlow-based generative strategy for efficient 3D dance generation, including (i) a few-step formulation with a Physical Consistency Constraint to stabilize optimization and improve physical plausibility, and (ii) a time-decayed soft masking strategy during inference to support coherent motion editing under few-step sampling. (3) We design a lightweight non-autoregressive model architecture with a BiMamba backbone and Channel-Level Cross-Modal Fusion, which reduces per-step computational and memory cost while maintaining strong music-dance alignment and long-horizon motion coherence.

2 Related Work

2.1 Flow Matching Generative Models

Flow Matching (FM) [28] learns the conditional vector field that transports noise samples to data samples, and performs generation by solving the corresponding ordinary differential equation (ODE). Compared with diffusion-based models such as DDPM [14], which rely on stochastic score estimation and long sampling chains, FM offers a simpler and more stable training objective, while enabling deterministic sampling through ODE integration. This makes FM theoretically appealing for efficient generation. To further improve efficiency, several variants of FM have been proposed for few-step sampling. Rectified Flow (RF) [29] constrains the learned ODE to follow straight paths between noise and data, thereby reducing the transport cost. Consistency Flow Matching (CFM) [51] enforces step-wise consistency so that generation can be carried out in only a few steps. Gaussian Mixture Flow (GMFlow) [5] models the transport vector field as a mixture of Gaussian components, allowing more flexible trajectories. Despite these efforts, it remains difficult to achieve high-quality generation with very few steps on complex tasks. To address this, MeanFlow [10] predicts the interval-averaged velocity rather than instantaneous ones, aligning the training objective with the ODE-based inference procedure, thereby enabling high-fidelity dance generation with significantly fewer sampling steps.

Beyond theoretical advances, FM has also shown strong potential in various generative tasks. In image generation, Stable Diffusion3 [8] incorporates rectified flow into large-scale training, significantly improving efficiency and scalability. In video generation, FM has been applied with rectified formulations to accelerate frame synthesis [6, 19], and Pyramid Flow Matching [18] demonstrates coherent

temporal modeling with fewer sampling steps. In audio generation, Voicebox [21] and Audiobox [43] adopt FM to build large-scale speech and audio models, while Matcha-TTS [33] and VoiceFlow [13] leverage conditional and rectified FM for efficient text-to-speech. Interestingly, choreography can be interpreted as a flow-matching process shaped by the potential induced by music, suggesting that employing Flow Matching for music-to-dance generation is a promising but hitherto unexplored direction.

2.2 Music-to-Dance Generation

Music and dance are inherently interconnected, which has driven substantial progress in music-to-dance generation (M2D), with most research focusing on 3D dance generation. Prior studies extract musical features using toolkits such as Librosa [32], Jukebox [7], and MERT [45], and then predict human motion represented by SMPL parameters [27, 31] or 2D/3D body keypoints [37]. Existing approaches can be broadly grouped into three paradigms: GAN-based, Autoregressive, and Diffusion-based models. **1) GAN-based models.** Generators synthesize motion from music while discriminators provide adversarial feedback. Examples include CoheDancers [48], DeepDance [39], and Choreography cGAN [16]. These methods are simple but prone to mode collapse and lack explicit motion constraints, often producing monotonous and repetitive movements. **2) Autoregressive models.** These methods typically adopt a two-stage pipeline, curating choreographic units from motion databases followed by autoregressive modeling of music-conditioned distributions over these units. Since units are derived from real motion, results are biomechanically plausible. Works such as Bailando [37], Bailando++ [38], Duolando [36], and MEGADance [47] fall into this paradigm. However, reliance on local history hampers global music–motion alignment, and step-by-step inference introduces high latency. **3) Diffusion-based models.** These methods corrupt motion with noise and train denoising networks to iteratively recover sequences conditioned on music, enabling diverse and temporally coherent dances. Representative works include EDGE [41], FineNet [26], Lodge [25], Lodge++ [24], and GCDance [30]. While achieving high quality, they require many sampling steps due to curved denoising trajectories, resulting in high inference cost. Beyond dance-specific studies, speech-driven 3D gesture generation has also made notable progress in recent years [53–58]. However, existing methods remain underexplored in achieving high-quality generation under a strict efficiency budget, due to (i) generative strategies that support high-fidelity few-step sampling, and (ii) per-step model architectures that are optimized for efficient yet refined long-horizon motion synthesis, thereby degrading downstream user experience in terms of responsiveness and visual fidelity.

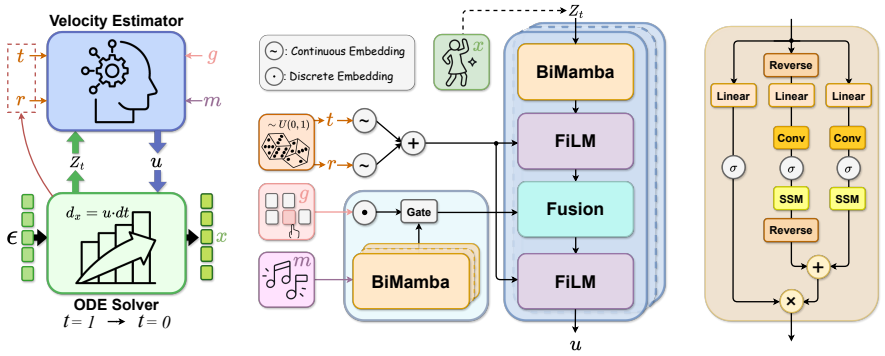

Fig. 2: Illustration of model architecture of FlowerDance at different levels.

3 Methodology

3.1 Oveview

Given a music sequence $M = \{m_0, m_1, ..., m_T\}$ and a dance genre label g, our objective is to synthesize the corresponding dance sequence $D = \{d_0, d_1, ..., d_T\}$, where m_t and d_t denote the music and dance feature at time step t. We define each music feature m_t as a 35-dim vector [25] extracted by Librosa [32], including 20-dim MFCC, 12-dim Chroma, 1-dim Peak, 1-dim Beat and 1-dim Envelope. We encode genre label g as a one-hot vector. We represent each dance feature as a 147-dim vector (3-dim root translation and 144-dim rotation representation [59]. Furthermore, we synchronize the music sequence with the dance sequence at a temporal granularity of 30 FPS.

To meet this strict inference budget in practice, we co-design FlowerDance across both the generative strategy and model architecture, as summarized in Fig. 2. At the generative-strategy level, we adopt a MeanFlow-based few-step formulation and introduce a Physical Consistency Constraint to stabilize large-interval updates and preserve physical plausibility in 3D motion generation. At the model-architecture level, we design a lightweight non-autoregressive Bi-Mamba backbone with Channel-Level Cross-Modal Fusion to reduce per-step latency and memory cost while maintaining music–dance alignment. To support real-time, human-in-the-loop motion authoring under few-step sampling, we further introduce a time-decayed soft masking strategy that enables coherent motion editing without additional training.

3.2 Generative Strategy

MeanFlow. In choreographic design, the development of a dance sequence can be viewed as a smooth and steadily paced refinement from an initial unstructured motion toward a coherent target performance, guided consistently by musical

rhythm and style [2, 34]. This progression naturally aligns with Flow Matching (FM) [28]'s assumption that the process can be modeled by an ordinary differential equation (ODE) that transports a standard Gaussian distribution sample $\epsilon \sim \mathcal{N}(0, I)$ to a target distribution sample $x \sim p_{\text{data}}(x)$ by estimating the instantaneous velocity field $v(z_t, t) = \epsilon - x$ along the straight-line flow path $\text{FP}(x, t)$, where time t is randomly sampled from a uniform distribution $U(0, 1)$:

$$\text{FP}(x, t) = (1 - t)x + t\epsilon \tag{1}$$

Here, *straight-line* refers to the standard linear interpolation in latent space between ϵ and x. A vector estimator v_θ is trained to minimize the ground truth velocity $v(z_t, t)$:

$$\mathcal{L}_{\text{FM}} = \mathbb{E}\big[\|v_\theta(z_t, t) - v(z_t, t)\|^2\big]. \tag{2}$$

At inference, generation starts from an initial prior sample z_1 and iteratively integrates the learned velocity field $v_\theta(z_t, t)$ until reaching z_0 in the data distribution:

$$\frac{d}{dt} z_t = v(z_t, t). \tag{3}$$

Even when conditional flows are designed to be straight, latent trajectories in the high-dimensional space are often highly curved. When using large integration intervals, Flow Matching [28], which models the instantaneous velocity field, cannot accurately capture these curved trajectories, leading to degraded generation quality. Thus, **MeanFlow** [10] replaces instantaneous velocity $v(z_t, t)$ with the interval mean velocity $u(z_t, r, t)$, leading to a training objective aligned with the actual ODE inference procedure:

$$u(z_t, r, t) = \frac{1}{t - r} \int_r^t v(z_\tau, \tau)\, d\tau, \tag{4}$$

where (r, t) are two time randomly sampled from a uniform distribution $U(0, 1)$ with the constraint $t > r$. Differentiating Eq. 4 with respect to t yields the MeanFlow identity:

$$u(z_t, r, t) = v(z_t, t) - (t - r)\frac{d}{dt} u(z_t, r, t). \tag{5}$$

A velocity estimator u_θ is trained to satisfy MeanFlow identity:

$$\begin{aligned} \mathcal{L}_{\text{MF}} &= \mathbb{E}\big[\|u_\theta(z_t, r, t) - \text{sg}(u_{\text{tgt}})\|_2^2\big], \\ \text{where} \quad u_{\text{tgt}} &= v(z_t, t) - (t - r)(v(z_t, t)\partial_z u_\theta + \partial_t u_\theta) \end{aligned} \tag{6}$$

where $\text{sg}(\cdot)$ denotes stop-gradient. Notably, this reduces to the FM loss when $r = t$. During inference, MeanFlow enables direct mapping by replacing the time integral with the mean velocity. We solve the ODE using an Euler-based discretization scheme, which naturally supports multi-step sampling in a straightforward manner:

$$z_r = z_t - (t - r)u(z_t, r, t). \tag{7}$$

Physical Consistency Constraint. Although MeanFlow has shown promising results in image, video, and audio synthesis [8, 19, 33, 43], directly applying it to few-step 3D dance generation is non-trivial. A key reason is that these modalities often exhibit explicit local continuity in their native representations, such as spatially neighboring pixels, adjacent video frames, or temporally continuous waveforms, which provides a relatively direct structure for learning smooth generative fields. In contrast, SMPL-based 3D motion is represented as a sequence of pose parameters whose temporal continuity is explicit across frames, while the intra-frame dependencies among joints are only implicitly encoded by the human kinematic chain. Under large integration intervals, small errors in the predicted velocity field can therefore accumulate over time and disrupt these latent kinematic constraints, leading to temporal jitter, global drift, foot sliding, and implausible articulations. Moreover, the vanilla MeanFlow objective only constrains the averaged velocity field, without explicitly regularizing the recovered motion in kinematic or physical spaces. Empirically, we find that training with $\mathcal{L}_{\mathrm{MF}}$ alone remains persistently unstable and fails to converge in the SMPL-based 3D motion setting. More details can refer to Section 3 in Appendix.

To impose motion-level physical regularization within MeanFlow, we first recover an explicit motion sample from the predicted interval-averaged velocity. Directly applying conventional physical constraints (e.g., velocity, acceleration, or keypoint losses) is not feasible, because the network predicts interval-averaged velocities rather than motion itself. Specifically, in each training iteration, we additionally sample a time $t_1 \sim U(0,1)$ and let the velocity estimator predict the mean velocity $u_\theta(z_{t_1}, 0, t_1)$ over the interval $(t_1, 0)$. We then recover the corresponding motion sample via Eq. 7 as $\hat{z_0} = z_{t_1} - t_1 \cdot u_\theta(z_{t_1}, 0, t_1)$. Finally, we incorporate a composite loss function that combines reconstruction, velocity, and 3D joint position losses between the prediction $\hat{z_0}$ and the ground-truth motion z_0, ensuring physical consistency and generation stability:

$$\begin{aligned} \mathcal{L}_{\mathrm{rec}} &= \mathbb{E}\big[\|\hat{z_0} - z_0\|_2^2\big], \\ \mathcal{L}_{\mathrm{pos}} &= \mathbb{E}\big[\|FK(\hat{z_0}) - FK(z_0)\|_2^2\big], \\ \mathcal{L}_{\mathrm{vel}} &= \mathbb{E}\big[\|FK(\hat{z_0})' - FK(z_0)'\|_2^2\big] \end{aligned} \tag{8}$$

where $FK(\cdot)$ denotes the forward kinematic function that converts joint angles into joint positions. Our overall training loss is the weighted sum of the MeanFlow objective and the Physical Consistency Constraint, where the weights λ are chosen to balance the magnitudes of the losses at the start of training:

$$\mathcal{L} = \lambda_{\mathrm{mf}}\mathcal{L}_{\mathrm{MF}} + \lambda_{\mathrm{rec}}\mathcal{L}_{\mathrm{rec}} + \lambda_{\mathrm{pos}}\mathcal{L}_{\mathrm{pos}} + \lambda_{\mathrm{vel}}\mathcal{L}_{\mathrm{vel}}. \tag{9}$$

Motion Editing. Extending the idea of inpainting in diffusion-based motion generation [30, 41], FlowerDance also provides a flexible motion editing capability during the sampling stage without additional training cost. This feature enables users to impose a wide range of constraints, such as generating smooth in-between movements in the temporal domain or modifying specific joint trajectories in the spatial domain, as shown in Fig. 4. For example, a user may

wish to interpolate missing frames between two dance clips or adjust the position of the hands while keeping other joints intact. Since FlowerDance operates with few sampling steps and lacks iterative correction, using a fixed hard mask may cause dynamic mismatches. To address this, we introduce a time-decayed soft mask, where the influence of the known regions gradually diminishes over time, allowing smoother coupling between constrained and unconstrained regions for more coherent inpainting results. Given a joint-wise or temporal constraint $x' \in \mathbb{R}^{N\times 147}$ with positions indicated by a binary mask $M \in \{0,1\}^{N\times 147}$ at time t, where N denotes the sequence length, FlowerDance performs constraint-aware denoising as follows:

$$z_t := (t \cdot M) \odot \mathrm{FP}(x', t) + (1 - (t \cdot M)) \odot z_t, \tag{10}$$

where $\odot$ denotes the Hadamard product, and known regions follow the constraint flow.

3.3 Model Architecture

Overview. Building on MeanFlow, our model architecture receives as input the time information (start t and end r), conditional information (music feature m and genre label g), and the current flow-path state z_t, and produces the interval-averaged velocity u, as illustrated in Fig. 2. For the conditional input, the music feature m is first processed by a multi-layer BiMamba, while the genre label g is encoded via discrete embedding; the resulting genre feature and music feature are then fused using a Gating Mechanism following [1], dynamically balancing the contributions of music and genre features. For the time information, the start time t and end time r are first represented as sinusoidal embeddings and then fused via addition. Finally, the velocity generation part consists of multi-layer blocks, and each block employs BiMamba for temporal modeling, Feature-wise Linear Modulation (FiLM [35]) for integrating time information, and Channel-Level Cross-Modal Fusion (Fusion) for integrating conditional information. For long-term generation, FlowerDance produces the entire sequence in a single non-autoregressive pass. Although simple in design, the synergistic interaction of these components enables FlowerDance to generate high-quality dances with high efficiency.

Bidirectional Mamba. Our choice of Mamba is motivated by the following considerations: (1) Music and dance demand strong local continuity between movements. Transformer is inherently position-invariant and captures sequence order only through positional encodings [42], which limits its deep understanding of local dependencies. Owing to its inherent sequential inductive bias, Mamba [11] has demonstrated strong performance in modeling fine-grained local dependencies [9, 44]. (2) Mamba has only $\mathcal{O}(n)$ computational complexity compared to the $\mathcal{O}(n^2)$ complexity of attention-based Transformers [42]. (3) Mamba supports non-autoregressive generation, which not only further improves efficiency but also mitigates the exposure-bias problem in autoregressive methods [37, 38, 47] and alleviates transition artifacts at segment boundaries in inpainting-based approaches [30, 41]. Furthermore, temporal dependencies in both music and dance

are inherently bidirectional, but standard Mamba models sequences only in a single direction. To address this limitation, we introduce Bidirectional Mamba (BiMamba), which captures information from both temporal directions. As shown in Fig. 2, the sequence is processed by forward and backward Mamba branches; their outputs are fused by addition and refined through a multiplicative skip connection, enhancing gradient flow and feature preservation. This bidirectional design improves motion coherence and strengthens music–dance alignment.

Specifically, Selective State Space model (Mamba) incorporates a Selection mechanism and a Scan module (S6) [11] to dynamically select salient input segments for efficient sequence modeling. Unlike the S4 model [12] with fixed parameters A, B, C, and scalar Δ, Mamba adaptively learns these parameters via fully-connected layers, enhancing generalization capabilities. Mamba employs structured state-space matrices, imposing constraints that improve computational efficiency. For each time step t, the input x_t, hidden state h_t, and output y_t follow:

$$\begin{aligned} h_t &= \bar{A}_t h_{t-1} + \bar{B}_t x_t, \\ y_t &= C_t h_t, \end{aligned} \tag{11}$$

where $\bar{A}_t, \bar{B}_t, C_t$ are dynamically updated parameters. Through discretization with sampling interval Δ, the state transitions become:

$$\begin{aligned} \bar{A} &= \exp(\Delta A), \\ \bar{B} &= (\Delta A)^{-1}(\exp(\Delta A) - I) \cdot \Delta B, \\ h_t &= \bar{A} h_{t-1} + \bar{B} x_t, \end{aligned} \tag{12}$$

where $(\Delta A)^{-1}$ denotes the inverse of ΔA and I is the identity matrix; the scan module captures temporal dependencies using shared trainable parameters over time.

Channel-Level Cross-Modal Fusion. Different from conventional *cross-attention*, we adopt *element-wise addition* for channel-level fusion for several reasons: (1) Music and dance are naturally aligned frame-by-frame, and temporal dependencies have already been fully modeled by BiMamba; (2) 3D dance datasets are typically small-scale, and the parameter-free addition operation offers stronger generalization in a non-autoregressive generation setting; (3) The lightweight approach reduces the parameters and improves efficiency.

4 Experiment

4.1 Dataset

FineDance. *FineDance* [26] is the largest public dataset for 3D music-to-dance generation, featuring professionally performed dances captured via optical motion capture. It provides 7.7 hours of motion data at 30 fps across 16 distinct dance genres. Following [25], we evaluate on test-set music clips, generating 1024-frame (34.13s) dance sequences.

Table 1: Quantitative comparison on dance quality on the FineDance dataset.

	FID_k↓	FID_g↓	FSR↓	DIV_k↑	DIV_g↑	BAS↑
Ground Truth	–	–	0.216	9.94	7.54	0.201
FACT [27] [ICCV'21]	113.38	97.05	0.284	3.36	6.37	0.183
MNET [20] [CVPR'22]	104.71	90.31	0.394	3.12	6.14	0.186
Bailando [37] [CVPR'22]	82.81	28.17	0.188	7.74	6.25	0.202
EDGE [41] [CVPR'23]	94.34	50.38	0.200	8.13	6.45	0.212
FineNet [26] [ICCV'23]	65.15	23.81	0.192	5.84	5.19	0.219
Lodge [25] [CVPR'24]	50.00	35.52	**0.028**	5.67	4.96	0.226
MEGA [47] [NeurIPS'25]	50.00	**13.02**	0.243	6.23	6.27	0.226
FlowerDance (**Ours**)	**29.73**	<u>19.59</u>	<u>0.147</u>	**8.42**	**7.18**	**0.232**

Table 2: Quantitative comparison on dance quality on the AIST++ dataset.

	FID_k↓	FID_g↓	Div_k↑	Div_g↑	BAS↑
Ground Truth	-	-	8.19	7.45	0.237
DanceNet [60] [TOMM'22]	69.18	25.49	2.86	2.85	0.143
FACT [27] [ICCV'21]	35.35	22.11	5.94	6.18	0.221
Bailando [37] [CVPR'22]	28.16	**9.62**	**7.83**	6.34	0.233
EDGE [41] [CVPR'23]	42.16	22.12	3.96	4.61	0.233
Lodge [25] [CVPR'24]	37.09	18.79	5.58	4.85	**0.242**
FlowerDance (**Ours**)	**20.50**	<u>15.75</u>	<u>6.95</u>	**6.52**	0.227

	DQ↑	DS↑	DC↑
FineNet [26]	3.72	3.53	3.51
Lodge [25]	4.09	3.89	4.22
MEGA [47]	4.12	4.23	4.27
FlowerDance	**4.18**	**4.41**	**4.33**

Table 3: Comparison of User study on the FineDance dataset.

	Param↓	FPS@10^3↑	FPS@10^4↑
Bailando [37]	152M	461	673
EDGE [41]	**50M**	372	589
FineNet [26]	94M	656	1137
Lodge [25]	235M	434	884
MEGA [47]	120M	288	340
FlowerDance	<u>63M</u>	**2008**	**4145**

Table 4: Comparison on generation efficiency.

AIST++. *AIST++* [27] is a widely used benchmark comprising 5.2 hours of 60 fps street dance motion, covering 10 dance genres. Following [27], we use test-set music clips to generate 1200-frame (20.00s) sequences.

4.2 Quantitative Evaluation

Dance Quality. We evaluate FlowerDance against state-of-the-art (SOTA) baselines on both the FineDance [26] and AIST++ [27] datasets using a comprehensive suite of metrics. For each generated sequence, we report Fréchet Inception Distance (FID) and Diversity (DIV) across two feature spaces [27, 37]: (1) kinetic (k), capturing motion dynamics, and (2) geometric (g), encoding spatial joint relations. To assess physical plausibility, we compute the Foot Slide Ratio (FSR) [25, 41]. To measure music–motion synchronization, we utilize the Beat Alignment Score (BAS) [25, 27]. On the FineDance dataset (Tab. 1), FlowerDance achieves the best results in FID_k (29.73), DIV_k (8.42), DIV_g (7.18), and BAS (0.232). It also attains competitive performance in FSR (0.147) and FID_g (19.59). On the AIST++ dataset (Tab. 2), FlowerDance obtains the lowest FID_k (20.50) and the highest DIV_g (6.52). It further shows strong performance in FID_g (15.75), DIV_k (6.95), and BAS (0.227). Taken together, these results establish FlowerDance as a state-of-the-art (SOTA) approach, delivering balanced and competitive performance across metrics and datasets.

Generation Efficiency. We evaluate generation efficiency from two perspectives: memory footprint (model parameters, Param) and inference speed (frames per second, FPS). We report FPS for both moderately long (1024 frames, FPS@10^3) and ultra-long (10240 frames, FPS@10^4) sequences to assess scalability across generation horizons. All experiments are conducted on a single NVIDIA RTX

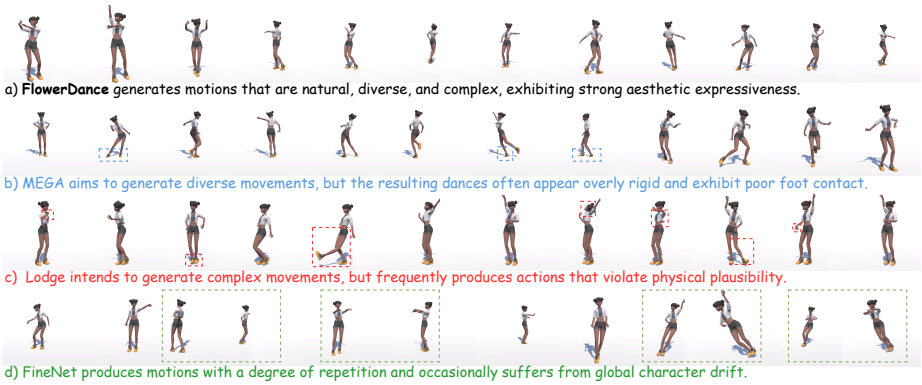

Fig. 3: Qualitative Analysis on a typical Eastern Folk music clip.

3090 GPU with an Intel Xeon Gold 5218 CPU (2.30 GHz). Benefiting from the BiMamba-based backbone and the MeanFlow-driven generative strategy, FlowerDance achieves the highest inference speed by a substantial margin while maintaining a competitive parameter size, only slightly larger than EDGE [41]. Importantly, the efficiency advantage becomes more pronounced as the sequence length increases, reaching up to FPS@10^4 = 4145. This favorable scaling behavior makes FlowerDance particularly suitable for real-time and long-horizon interactive applications.

User Study. Dance's inherent subjectivity makes user feedback essential for evaluating generated movements [22] in the music-to-dance generation task. Following [47], we select 40 in-the-wild music segments (34.13s) and generate dance sequences using above models. These sequences are evaluated through a double-blind questionnaire, by 40 participants with backgrounds in dance practice. The questionnaires are based on a 5-point scale (Great, Good, Fair, Bad, Terrible) and assess three aspects: Dance Synchronization (DS, alignment with rhythm and style), Dance Quality (DQ, biomechanical plausibility and aesthetics), and Dance Creativity (DC, originality and range). As shown in Tab. 3, FlowerDance significantly outperforms all baselines across user-rated metrics (i.e., $DQ = 4.18$, $DS = 4.41$, $DC = 4.33$). Its high scores in various aspects underscore its superiority in generating movements in terms of human preferences.

4.3 Qualitative Evaluation

Comparison. To assess the visual quality of the generated dances, we conduct a qualitative comparison between FlowerDance and representative baselines, as shown in Fig. 3. MEGA [47] seeks to promote diversity, yet its outputs often appear rigid and suffer from poor foot–ground contact. FineNet [26] produces motions with a noticeable degree of repetition and sometimes causes the entire character to drift globally. Lodge [25] attempts to generate complex movements, but its results frequently contain physically implausible actions, such as unnat-

ural bending of the torso and unrealistic limb articulations. In contrast, FlowerDance produces motions that are natural, diverse, and physically coherent, while also demonstrating strong aesthetic expressiveness. This superior performance stems from two main factors: (1) the powerful generative capability of MeanFlow-based model with the Physical Consistency Constraint, enabling the synthesis of complex and diverse movements while ensuring generation stability and physical plausibility; (2) the strong temporal understanding of the BiMamba-based architecture with Channel-Level Cross-Modal Fusion, enabling coherent long-range choreography while maintaining alignment with musical cues.

Motion Editing Motion editing is an important step in dance generation, as it allows users to refine results through interactive control. As shown in Fig. 4, FlowerDance with soft mask generates smooth and high-quality in-between motions given start and end clips. Without the proposed time-decayed soft constraint, FlowerDance (hard mask) can still produce reasonable motions within the inpainted segment, but abrupt discontinuities often occur at the transition boundaries between fixed and inpainted regions, severely compromising the fluency of the generated dance. Details are provided in the supplementary video.

4.4 Ablation Study

Tab. 4-5 summarize the quantitative statistics, while qualitative visualizations are provided in supplementary video. **1) Generative Model.** We compare MeanFlow against Rectified Flow (RectFlow) [29] and Diffusion [41] under different ODE sampling steps, using the same architecture for fairness. We omit BAS because its unstable jitters can coincidentally align with beats and yield misleadingly high scores. MeanFlow attains state-of-the-art (SOTA) performance with 20 sampling steps, remains near-SOTA at 10 steps, and still produces reasonable results with only 5 steps; performance collapses at a single step, indicating one-step generation is an open but promising direction. From visual inspection, 20-step generations are fully stable, 10-step outputs sometimes show minor jitter, and 5-step outputs occasionally exhibit noticeable jitter. However, RectFlow and Diffusion require 50 steps to reach the quality that MeanFlow achieves with 10 steps. The key reason is that MeanFlow predicts interval-averaged velocity rather than instantaneous velocity, aligning the training objective with the ODE updates used at inference and removing the training–inference mismatch present in RectFlow. Diffusion models rely on progressive denoising to correct errors over many small steps; in few-step regimes, each update must span a large time interval and, without explicit large-step velocity modeling, denoising effectively becomes long-range interpolation, producing trajectory mismatches and loss of fine motion details. Overall, integrating MeanFlow enables FlowerDance to synthesize high-fidelity dances with far fewer sampling steps, thereby significantly improving generation efficiency.

2) Model Backbone. To evaluate the effect of the trans-temporal backbone, we compare BiMamba against Mamba, Transformer (II) and Transformer (NAR). Both II and NAR are trained on 8-second clips; at inference, II iteratively inpaints to 34.13s, whereas NAR generates the full 34.13s in a single pass with

Fig. 4: Given start (yellow) and end (red) motions, FlowerDance generates smooth in-between motions with a soft mask (green), whereas a hard mask (blue) causes teleportation artifacts.

	S	$FID_k\downarrow$	$FID_g\downarrow$	FSR ↓	$Div_k\uparrow$	$Div_g\uparrow$	FPS ↑
Ground Truth	-	–	–	0.216	9.94	7.54	-
MeanFlow	5	29.05	55.96	1.206	7.62	6.73	3531
	10	26.17	25.00	0.201	8.20	7.10	2725
	20	29.73	19.59	0.147	8.42	7.18	2008
RectFlow	20	49.59	39.65	0.247	8.18	7.21	2275
	50	33.47	38.81	0.251	7.01	6.60	1066
Diffusion	20	66.46	48.25	0.274	6.78	6.38	1462
	50	32.06	28.44	0.207	8.75	7.04	890

Table 5: Effect of the Generative Model under different ODE sampling steps (S) during inference on the FineDance dataset.

non-autoregression. This setting reflects the practical need for temporal extrapolation, as real-world dance production often requires 3-4 minute sequences, far beyond the maximum continuous duration (1 minute) covered by existing training data. BiMamba substantially outperforms both alternatives, as it effectively captures bidirectional dependencies between music and dance, achieving superior generation quality. Mamba is more efficient but underperforms on metrics and lacks aesthetic appeal and motion diversity. Transformer (NAR) fails to generalize to long sequences—metrics collapse and outputs degenerate into repetitive in-place jittering, owing to self-attention's scale-dependent positional extrapolation. Transformer (II) attains decent metrics (slightly below BiMamba) but incurs excessive latency; visually, it occasionally exhibits "teleportation," an inherent boundary limitation of inpainting-based approaches. Overall, BiMamba serves as an effective and efficient backbone for FlowerDance.

3) Cross Modal Fusion. We further investigate the effect of channel-level cross-modal fusion by replacing the element-wise addition (ADD) with the widely used cross-attention (CA). Overall, ADD shows some improvements over CA. In practice, motions generated by CA are also reasonable, but those produced by ADD tend to align more accurately with musical beats. While the gains are modest, ADD reduces computational cost and inference time, thereby improving FlowerDance's generation efficiency.

4) Physical Consistency Constraint. We analyze the Physical Consistency Constraint (PCC) via ablation. PCC imposes inductive biases toward physically plausible kinematics, stabilizing optimization without introducing inference-time overhead. Without PCC, training fails to converge: metrics become NaN and motions collapse into invalid sequences with severe temporal jitter and global drift. Even when retaining only the reconstruction term (w/o. $\mathcal{L}_{\text{pos}} + \mathcal{L}_{\text{vel}}$) or combining it with the forward-kinematic constraint ((w/o. $\mathcal{L}_{\text{vel}}$)), the model maintains competitive dance quality, underscoring PCC's essential role in stabilizing MeanFlow optimization.

5) MeanFlow Loss. Removing the MeanFlow Loss $\mathcal{L}_{\text{mf}}$ effectively reduces FlowerDance to a specific Rectified Flow Matching–based generative strategy. Across all step settings (20/50), its removal results in substantial performance

Table 6: More ablation study on the FineDance dataset.

	$FID_k\downarrow$	$FID_g\downarrow$	FSR↓	$Div_k\uparrow$	$Div_g\uparrow$	BAS↑	FPS ↑
Ground Truth	–	–	0.216	9.94	7.54	0.201	–
BiMamba → Mamba	39.40	38.93	0.197	8.69	7.07	0.219	**2387**
BiMamba → Transformer(NAR)	NaN	NaN	NaN	NaN	NaN	NaN	1829
BiMamba → Transformer(II)	**23.29**	21.02	0.191	8.04	7.01	0.226	218
Addition → Cross Attention	32.40	24.45	0.194	**9.12**	6.90	0.229	1463
w/o. PCC	NaN	NaN	NaN	NaN	NaN	NaN	2008
w/o. $\mathcal{L}\text{pos} + \mathcal{L}_{\text{vel}}$	37.45	22.56	**0.124**	8.03	6.32	0.224	2008
w/o. $\mathcal{L}_{\text{vel}}$	25.86	27.41	0.201	8.26	7.44	0.229	2008
w/o. MeanFlow Loss (20-step)	49.59	39.65	0.247	8.18	7.21	0.223	2275
w/o. MeanFlow Loss (50-step)	33.47	38.81	0.251	7.01	6.60	**0.232**	1066
FlowerDance (**Full Model**)	29.73	**19.59**	**0.147**	8.42	**7.18**	**0.232**	2008

degradation, highlighting its critical role in stabilizing optimization and improving generation quality in few step generation setting.

5 Conclusion

In this work, we introduce FlowerDance, an efficient and refined framework for music-to-dance generation, achieving state-of-the-art performance in both dance quality and generation efficiency. By incorporating MeanFlow and Physical Consistency Constraints, FlowerDance ensures faithful motion trajectories while requiring only few sampling steps. Moreover, FlowerDance leverages the BiMamba backbone and Channel-Level Cross-Modal Fusion to construct its model architecture. Meanwhile, FlowerDance also supports motion editing, enabling users to interactively refine dance sequences. In future work, we plan to extend FlowerDance with text conditioning to enable more flexible dance generation.

6 Acknowledge

This work was supported in part by the National Natural Science Foundation of China under Grants 62436010, 72572090, 62572474 and 62172421.

References

1. Arevalo, J., Solorio, T., Montes-y Gómez, M., González, F.A.: Gated multimodal units for information fusion. arXiv preprint arXiv:1702.01992 (2017)
2. Butterworth*, J.: Teaching choreography in higher education: A process continuum model. Research in dance education **5**(1), 45–67 (2004)
3. Chen, C., Hu, S., Zhu, J., Wu, M., Chen, J., Li, Y., Huang, N., Fang, C., Wu, J., Chu, X., et al.: Taming preference mode collapse via directional decoupling alignment in diffusion reinforcement learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12775–12786 (2026)

4. Chen, C., Zhu, J., Feng, X., Huang, N., Wu, M., Mao, F., Wu, J., Chu, X., Li, X.: S^2-Guidance: Stochastic self guidance for training-free enhancement of diffusion models. arXiv preprint arXiv:2508.12880 (2025)
5. Chen, H., Zhang, K., Tan, H., Xu, Z., Luan, F., Guibas, L., Wetzstein, G., Bi, S.: Gaussian mixture flow matching models. arXiv preprint arXiv:2504.05304 (2025)
6. Davtyan, A., Sameni, S., Favaro, P.: Efficient video prediction via sparsely conditioned flow matching. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 23263–23274 (2023)
7. Dhariwal, P., Jun, H., Payne, C., Kim, J.W., Radford, A., Sutskever, I.: Jukebox: A generative model for music. arXiv preprint arXiv:2005.00341 (2020)
8. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al.: Scaling rectified flow transformers for high-resolution image synthesis. In: Forty-first international conference on machine learning (2024)
9. Fu, C., Wang, Y., Zhang, J., Jiang, Z., Mao, X., Wu, J., Cao, W., Wang, C., Ge, Y., Liu, Y.: Mambagesture: Enhancing co-speech gesture generation with mamba and disentangled multi-modality fusion. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 10794–10803 (2024)
10. Geng, Z., Deng, M., Bai, X., Kolter, J.Z., He, K.: Mean flows for one-step generative modeling. arXiv preprint arXiv:2505.13447 (2025)
11. Gu, A., Dao, T.: Mamba: Linear-time sequence modeling with selective state spaces. arXiv preprint arXiv:2312.00752 (2023)
12. Gu, A., Goel, K., Ré, C.: Efficiently modeling long sequences with structured state spaces. arXiv preprint arXiv:2111.00396 (2021)
13. Guo, Y., Du, C., Ma, Z., Chen, X., Yu, K.: Voiceflow: Efficient text-to-speech with rectified flow matching. In: ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 11121–11125. IEEE (2024)
14. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
15. Hu, S., Chen, C., Zhu, J., Wu, J., Chu, X., Li, X.: Embedding-perturbed exploration preference optimization for flow models. arXiv preprint arXiv:2605.15803 (2026)
16. Huang, Y.F., Liu, W.D.: Choreography cgan: generating dances with music beats using conditional generative adversarial networks. Neural Computing and Applications **33**(16), 9817–9833 (2021)
17. Jia, T., Yang, K., Yang, X., Tang, X., Qiu, K., Wei, S., Zhao, Y.: Bitdiff: Fine-grained 3d conducting motion generation via bimamba-transformer diffusion (2026), `https://arxiv.org/abs/2604.04395`
18. Jin, Y., Sun, Z., Li, N., Xu, K., Jiang, H., Zhuang, N., Huang, Q., Song, Y., Mu, Y., Lin, Z.: Pyramidal flow matching for efficient video generative modeling. arXiv preprint arXiv:2410.05954 (2024)
19. Ki, T., Min, D., Chae, G.: Float: Generative motion latent flow matching for audio-driven talking portrait. arXiv preprint arXiv:2412.01064 (2024)
20. Kim, J., Oh, H., Kim, S., Tong, H., Lee, S.: A brand new dance partner: Music-conditioned pluralistic dancing controlled by multiple dance genres. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3490–3500 (2022)
21. Le, M., Vyas, A., Shi, B., Karrer, B., Sari, L., Moritz, R., Williamson, M., Manohar, V., Adi, Y., Mahadeokar, J., et al.: Voicebox: Text-guided multilingual universal speech generation at scale. Advances in neural information processing systems **36**, 14005–14034 (2023)

22. Legrand, D., Ravn, S.: Perceiving subjectivity in bodily movement: The case of dancers. Phenomenology and the Cognitive Sciences **8**, 389–408 (2009)
23. Li, R., Hu, Z., Siyao, L., Zhang, Y., Xie, H., Zhang, M., Guo, J., Li, X., Liu, Z.: Infinitedance: Scalable 3d dance generation towards in-the-wild generalization (2026), https://arxiv.org/abs/2603.13375
24. Li, R., Zhang, H., Zhang, Y., Zhang, Y., Zhang, Y., Guo, J., Zhang, Y., Li, X., Liu, Y.: Lodge++: High-quality and long dance generation with vivid choreography patterns. arXiv preprint arXiv:2410.20389 (2024)
25. Li, R., Zhang, Y., Zhang, Y., Zhang, H., Guo, J., Zhang, Y., Liu, Y., Li, X.: 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)
26. Li, R., Zhao, J., Zhang, Y., Su, M., Ren, Z., Zhang, H., Tang, Y., Li, X.: 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)
27. Li, R., Yang, S., Ross, D.A., Kanazawa, A.: Ai choreographer: Music conditioned 3d dance generation with aist++. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13401–13412 (2021)
28. Lipman, Y., Chen, R.T., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. arXiv preprint arXiv:2210.02747 (2022)
29. Liu, X., Gong, C., Liu, Q.: Flow straight and fast: Learning to generate and transfer data with rectified flow. arXiv preprint arXiv:2209.03003 (2022)
30. Liu, X., Dong, X., Kanojia, D., Wang, W., Feng, Z.: Gcdance: Genre-controlled 3d full body dance generation driven by music. arXiv preprint arXiv:2502.18309 (2025)
31. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: Smpl: A skinned multi-person linear model. In: Seminal Graphics Papers: Pushing the Boundaries, Volume 2, pp. 851–866 (2023)
32. McFee, B., Raffel, C., Liang, D., Ellis, D.P., McVicar, M., Battenberg, E., Nieto, O.: librosa: Audio and music signal analysis in python. In: SciPy. pp. 18–24 (2015)
33. Mehta, S., Tu, R., Beskow, J., Székely, É., Henter, G.E.: Matcha-tts: A fast tts architecture with conditional flow matching. In: ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 11341–11345. IEEE (2024)
34. Morris, G.: Dance studies/cultural studies. Dance Research Journal **41**(1), 82–100 (2009)
35. Perez, E., Strub, F., De Vries, H., Dumoulin, V., Courville, A.: Film: Visual reasoning with a general conditioning layer. In: Proceedings of the AAAI conference on artificial intelligence. vol. 32 (2018)
36. Siyao, L., Gu, T., Yang, Z., Lin, Z., Liu, Z., Ding, H., Yang, L., Loy, C.C.: Duolando: Follower gpt with off-policy reinforcement learning for dance accompaniment. arXiv preprint arXiv:2403.18811 (2024)
37. Siyao, L., Yu, W., Gu, T., Lin, C., Wang, Q., Qian, C., Loy, C.C., Liu, Z.: 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)
38. Siyao, L., Yu, W., Gu, T., Lin, C., Wang, Q., Qian, C., Loy, C.C., Liu, Z.: Bailando++: 3d dance gpt with choreographic memory. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023)

39. Sun, G., Wong, Y., Cheng, Z., Kankanhalli, M.S., Geng, W., Li, X.: Deepdance: music-to-dance motion choreography with adversarial learning. IEEE Transactions on Multimedia **23**, 497–509 (2020)
40. Sun, K., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5693–5703 (2019)
41. Tseng, J., Castellon, R., Liu, K.: Edge: Editable dance generation from music. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 448–458 (2023)
42. Vaswani, A.: Attention is all you need. Advances in Neural Information Processing Systems (2017)
43. Vyas, A., Shi, B., Le, M., Tjandra, A., Wu, Y.C., Guo, B., Zhang, J., Zhang, X., Adkins, R., Ngan, W., et al.: Audiobox: Unified audio generation with natural language prompts. arXiv preprint arXiv:2312.15821 (2023)
44. Xu, Z., Lin, Y., Han, H., Yang, S., Li, R., Zhang, Y., Li, X.: Mambatalk: Efficient holistic gesture synthesis with selective state space models. In: The Thirty-eighth Annual Conference on Neural Information Processing Systems (2024)
45. Yang, K., Tang, X., Diao, R., Liu, H., He, J., Fan, Z.: Codancers: Music-driven coherent group dance generation with choreographic unit. In: Proceedings of the 2024 International Conference on Multimedia Retrieval. pp. 675–683 (2024)
46. Yang, K., Tang, X., Hu, Y., Yang, J., Liu, H., Zhang, Q., He, J., Fan, Z.: Matchdance: Collaborative mamba-transformer architecture matching for high-quality 3d dance synthesis. arXiv preprint arXiv:2505.14222 (2025)
47. Yang, K., Tang, X., Peng, Z., Hu, Y., He, J., Liu, H.: Megadance: Mixture-of-experts architecture for genre-aware 3d dance generation. arXiv preprint arXiv:2505.17543 (2025)
48. Yang, K., Tang, X., Wu, H., Xue, Q., Qin, B., Liu, H., Fan, Z.: Cohedancers: Enhancing interactive group dance generation through music-driven coherence decomposition. arXiv preprint arXiv:2412.19123 (2024)
49. Yang, K., Zhou, X., Tang, X., Diao, R., Liu, H., He, J., Fan, Z.: Beatdance: A beat-based model-agnostic contrastive learning framework for music-dance retrieval. In: Proceedings of the 2024 International Conference on Multimedia Retrieval. pp. 11–19 (2024)
50. Yang, K., Zhu, J., Tang, X., Peng, Z., Zhang, X., Wang, P., Wu, J., Chu, X., Liu, H., He, J.: Mace-dance: Motion-appearance cascaded experts for music-driven dance video generation (2026), https://arxiv.org/abs/2512.18181
51. Yang, L., Zhang, Z., Zhang, Z., Liu, X., Xu, M., Zhang, W., Meng, C., Ermon, S., Cui, B.: Consistency flow matching: Defining straight flows with velocity consistency. arXiv preprint arXiv:2407.02398 (2024)
52. Yang, Z., Yang, K., Tang, X.: Tokendance: Token-to-token music-to-dance generation with bidirectional mamba. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. pp. 5520–5529 (June 2026)
53. Zhang, X., Cai, Y., Li, K., Yang, K., Zhou, Y., Li, Z., Chu, X., Zhang, J., Liu, H.: Personagesture: Single-reference co-speech gesture personalization for unseen speakers. arXiv preprint arXiv:2605.06064 (2026)
54. Zhang, X., Jia, Y., Zhang, J., Yang, Y., Tu, Z.: Robust 2d skeleton action recognition via decoupling and distilling 3d latent features. IEEE Transactions on Circuits and Systems for Video Technology (2025)

55. Zhang, X., Li, J., Ren, J., Zhang, J.: Mitigating error accumulation in co-speech motion generation via global rotation diffusion and multi-level constraints. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 40, pp. 12834–12842 (2026)
56. Zhang, X., Li, J., Zhang, J., Dang, Z., Ren, J., Bo, L., Tu, Z.: Semtalk: Holistic co-speech motion generation with frame-level semantic emphasis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13761–13771 (2025)
57. Zhang, X., Li, J., Zhang, J., Ren, J., Bo, L., Tu, Z.: Echomask: Speech-queried attention-based mask modeling for holistic co-speech motion generation. In: Proceedings of the 33rd ACM International Conference on Multimedia. pp. 10827–10836 (2025)
58. Zhou, P., Zhang, X., Shen, X., Hu, Y.: Not all frames are equal: Complexity-aware masked motion generation via motion spectral descriptors. arXiv preprint arXiv:2603.29655 (2026)
59. Zhou, Y., Barnes, C., Lu, J., Yang, J., Li, H.: On the continuity of rotation representations in neural networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5745–5753 (2019)
60. Zhuang, W., Wang, C., Chai, J., Wang, Y., Shao, M., Xia, S.: Music2dance: Dancenet for music-driven dance generation. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM) **18**(2), 1–21 (2022)