# MotionRAG-Diff: A Retrieval-Augmented Diffusion Framework for Long-Term Music-to-Dance Generation

Mingyang Huang, Peng Zhang, Bang Zhang Tongyi Lab, Alibaba Group {hongcan.hmy, futian.zp, zhangbang.zb}@alibaba-inc.com

# Abstract

Generating long-term, coherent, and realistic music-conditioned dance sequences remains a challenging task in human motion synthesis. Existing approaches exhibit critical limitations: motion graph methods rely on fixed template libraries, restricting creative generation; diffusion models, while capable of producing novel motions, often lack temporal coherence and musical alignment. To address these challenges, we propose **MotionRAG-Diff**, a hybrid framework that integrates Retrieval-Augmented Generation (RAG) with diffusion-based refinement to enable high-quality, musically coherent dance generation for arbitrary long-term music inputs. Our method introduces three core innovations: (1) A cross-modal contrastive learning architecture that aligns heterogeneous music and dance representations in a shared latent space, establishing unsupervised semantic correspondence without paired data; (2) An optimized motion graph system for efficient retrieval and seamless concatenation of motion segments, ensuring realism and temporal coherence across long sequences; (3) A multi-condition diffusion model that jointly conditions on raw music signals and contrastive features to enhance motion quality and global synchronization. Extensive experiments demonstrate that MotionRAG-Diff achieves state-of-the-art performance in motion quality, diversity, and music-motion synchronization accuracy. This work establishes a new paradigm for music-driven dance generation by synergizing retrieval-based template fidelity with diffusion-based creative enhancement.

# 1 Introduction

Dance motion generation from music [25] [2] [9] [14] [19] [30] [32] [35] [16] [10] has emerged as a pivotal research area in human motion synthesis, with significant applications in entertainment, virtual reality, and human-computer interaction. Current approaches predominantly follow two paradigms: motion graph methods [21] [4] that rely on template-based action retrieval and pure generative models like diffusion-based frameworks [35] [17] [10]. However, these approaches exhibit inherent limitations that hinder the creation of high-quality, musically coherent dance sequences. Motion graph methods, while ensuring temporal coherence through pre-defined motion templates, suffer from a fundamental deficiency - their inability to generate novel dance patterns beyond the template library. Conversely, pure diffusion models demonstrate strong generative capabilities but often produce unnatural motion sequences that lag behind the quality of template-based actions. This dichotomy between template fidelity and creative generation remains a critical challenge in the field.

This paper presents a novel hybrid framework that synergistically combines the strengths of motion graphs and diffusion models while addressing their limitations through innovative architectural design. Our key contribution lies in developing a contrastive learning framework that effectively captures the complex correlations between musical features and corresponding dance movements. By

integrating this contrastive learning mechanism with an optimized motion graph structure, we achieve more accurate motion-node matching while maintaining temporal consistency across long music sequences. The proposed approach innovatively incorporates the principles of Retrieval-Augmented Generation (RAG), where the most semantically relevant motion segments are first retrieved from the motion graph, followed by diffusion-based refinement that enhances both motion quality and musical alignment.

The technical innovations of our framework include: 1) A contrastive learning architecture that learns discriminative representations for music-dance correspondence; 2) An enhanced motion graph system handles arbitrary long-term length music inputs through intelligent motion segment stitching; 3) The integration of DiT (Diffusion Transformer) [26] architecture to improve the quality of generated motion sequences. Extensive experiments demonstrate that our method not only preserves the naturalness of template-based motions but also enables the creation of novel dance patterns through diffusion-based enhancement. This dual capability of leveraging existing motion knowledge while enabling creative generation represents a significant advancement in musically driven dance motion synthesis. Our approach achieves state-of-the-art performance across multiple evaluation metrics on AIST++ [19] and FineDance [18] datasets, including both quantitative measures and qualitative assessments.

# 2 Related Works

Recent advances in music-conditioned dance generation have explored diverse paradigms, including contrastive learning, motion graphs, and diffusion models. Each addresses unique challenges in aligning audio and motion data. Below, we categorize existing approaches and highlight their distinctions from our proposed method.

**Contrastive Learning**. Contrastive learning has been widely adopted to bridge heterogeneous modalities. In the domain of action generation, MotionClip [33] and CLIP [36] leverage vision-language pretraining to align text and motion/image representations, while TANGO [21] introduces a hierarchical audio-motion joint embedding space for speech-driven gesture synthesis. For audio-visual alignment, Wav2Clip [36] and Wav2Vec2 [3] demonstrate effective music-image and speech-embedding correspondence, respectively. Notably, MoMask [7] employs residual cascading with discrete motion encoding to model long-term dependencies. However, these works primarily focus on text-motion (MotionClip [33]), speech-motion (TANGO [21]), or music-image (Wav2Clip [36]) alignment. In contrast, our method explicitly addresses music-to-3D motion alignment by integrating MoMask's motion discretization with Wav2Clip's audio encoding strategy in a shared latent space. This enables unsupervised semantic correspondence without requiring paired data, a critical departure from prior methods.

**Motion Graph**. Motion graphs have been pivotal in ensuring temporal continuity in generated sequences. ChoreoMaster [4] constructs motion graphs using positional and velocity features, augmented with learned style embeddings and rhythm signatures to prevent style discontinuities. GVR [40] extends this to speech-driven gesture generation by incorporating SMPL [22] mesh IoU for node adjacency. TANGO [21] further refines cross-modal alignment through latent feature distance metrics and employs max-connected subgraph pruning to enable infinite-length generation. HMInterp [20] adapts TANGO's [21] framework to tag/description-to-dance tasks, prioritizing graph traversal cost minimization over contrastive matching. Our approach builds on TANGO's graph construction and pruning strategies but integrates contrastive learning-based node selection to ensure music-motion coherence. This hybridization of retrieval and generation principles allows seamless concatenation of motion segments while preserving rhythmic and semantic alignment.

**Diffusion Models**. Diffusion models [31] [8] have emerged as powerful tools for motion synthesis. Early works like MotionDiffuse [38] and ReMoDiffuse [39] establish text-to-motion generation pipelines, while MoRAG [11] partitions body segments (upper/lower body, torso) for retrieval-enhanced refinement. EDGE [35] and LODGE [17] introduce controllability via music editing and coarse-to-fine generation, with LODGE++ [16] optimizing for flexible primitives using VQ-VAE and GPT-based choreography networks. DiffDance [28] and Beat-It [10] further refine alignment by conditioning on contrastive audio embeddings or explicit beat loss. Our method diverges by combining multi-condition diffusion with a preprocessing network that fuses raw audio, contrastive embeddings, top-k retrieved motions, and beat annotations. This architecture enables high-fidelity generation



Figure 1: The overall framework of our work. It contains three core components: the contrastive learning model, the motion graph, and the diffusion model. This integrated architecture enables the processing of arbitrarily long-term music inputs for coherent and high-quality dance motion generation.

while maintaining global music-motion synchronization, surpassing the localized refinements of ReMoDiffuse [39] and MoRAG [11].

Other works explore reinforcement learning (Bailando [29], Bailando++ [30]) for rhythm alignment via VQ-VAE and hybrid training strategies. While these methods emphasize temporal precision, our framework prioritizes semantic consistency through contrastive learning and hierarchical motion graph design.

By synthesizing insights from these paradigms, our work establishes a novel hybrid framework that unifies retrieval-based template fidelity with diffusion-based creative enhancement, addresses the limitations of prior methods in motion quality, diversity, and music-motion alignment.

# 3 Methodology

As illustrated in Figure 1, our framework consists of three main components: the Contrastive Learning Model, the Motion Graph, and the Diffusion Model. The retrieval phase, following the principles of Retrieval-Augmented Generation (RAG), is conducted between the Contrastive Learning Model and the Motion Graph to select semantically relevant motion segments. The augmentation and generation phase of RAG is then applied within the Diffusion Model to refine and enhance the retrieved motions. Consequently, our approach can be characterized as a RAG-based framework for music-to-dance generation.

## 3.1 Contrastive Learning Model

To establish the correspondence between music and motion, we employ a contrastive learning model to learn the underlying correlations from our motion database.

As illustrated in Figure 2, the contrastive learning framework consists of a motion encoder and a music encoder. The overall architecture follows a standard design similar to that in [28]. In our implementation, both the motion and music encoders are retrained to improve the alignment between audio and motion representations in the shared latent space.



Figure 2: The contrastive learning pipeline. It contains a motion encoder, a music encoder, and an adaptive layer that follows the music encoder. All parameters in the pipeline are re-trained through the training process.



Figure 3: The music-motion similarity matrix after the contrastive learning model. The left image depicts the similarity computed directly from the raw features of motion and music, whereas the right image shows the result obtained from their embeddings after being processed by the contrastive learning model.

**Motion Encoding**. We leverage the motion encoding capability provided by MoMask [7] and follow the architectural settings proposed in the original work. First, we pre-train the model on the AIST++ [19] and FineDance [18] datasets individually. Subsequently, we fine-tune the entire network within the contrastive learning framework to further enhance the alignment between music and motion representations.

**Music Encoding**. We follow the architectural settings proposed in [36] and incorporate an adaptive layer inspired by [28]. While [36] focuses on learning the correlation between audio and images, our task aims to model the relationship between audio and 3D motion. In our framework, we fine-tune the pre-trained music encoder and train all parameters of the adaptive layer from scratch to better align the audio and motion representations in the shared latent space.

**Music-Motion Contrastive Learning**. We learn the correlation between motion and music features using the InfoNCE loss [1]. The contrastive learning objective for the <music, motion> pairs is formulated as follows:

$$\mathcal{L}_i^{m \to d} = -\log \frac{exp[s(m_i, d_i)/\tau]}{\sum_{j=1}^N exp[s(m_i, d_i)/\tau]},\tag{1}$$

where  $m_i$  stands for the *i*-th music clip,  $d_i$  stands for the *i*-th dance sequence,  $\tau$  stands for a learnable temperature parameter. Figure 2 illustrates the overall pipeline of the training process of the contrastive learning model.

As shown in Figure 3, the correlation between motion and music becomes significantly stronger after applying contrastive learning. The left image depicts the similarity computed directly from the raw features of motion and music, whereas the right image shows the result obtained from their embeddings after being processed by the contrastive learning model. The prominent diagonal pattern in the right image indicates that the learned representations bring the motion and music features much closer in the latent space, demonstrating the effectiveness of the contrastive learning process.

## 3.2 Motion Graph

Following a similar approach to TANGO [21], we construct a motion graph to establish connections among different motion segments from the motion database. The construction process comprises two main stages: graph building and graph pruning. In the first stage, all motion clips are integrated into a unified graph structure based on their compatibility in terms of position and velocity. The second stage involves pruning the graph to identify the largest connected subgraph, which ensures temporal coherence and enables the generation of arbitrarily long motion sequences.

**Graph Building**. The motion graph consists of nodes and edges. Each node represents a 3D motion clip, containing both positional and velocity information. Edges are constructed based on the compatibility between the position and velocity of adjacent nodes. The detailed edge-building procedure is outlined in Algorithm 1.



Figure 4: The comparison of motion graph pruning results. The first and second graphs are constructed based on the AIST++ [19] dataset, while the third and fourth are built using the FineDance [18] dataset. red points represent isolated nodes that are removed during the pruning process, whereas blue points indicate connected nodes. The green arrows illustrate the directional relationships between adjacent, connected nodes in the pruned graph.

### Algorithm 1 Motion Graph Edge Building Process

## **Require:**

- 1: N: number of frames for mean calculation
- 2: T: body joint count threshold
- 3: current node: sequence of joint positions/velocities
- 4: next node: sequence of joint positions/velocities
- Ensure: Edge between nodes if motion continuity is satisfied
- 5:  $M_p \leftarrow$  mean of last N frames' positions
- 6:  $\dot{M_v} \leftarrow$  mean of last N frames' velocities
- 7:  $T_p \leftarrow$  current node's position differences from  $M_p$
- 8:  $T_v \leftarrow$  current node's velocity differences from  $M_v$
- 9:  $S_p \leftarrow$  position difference between last frame of current node and first frame of next node 10:  $S_v \leftarrow$  velocity difference between last frame of current node and first frame of next node
- 11: if each node with  $Count_{S_p < T_p} \ge T$  and  $Count_{S_v < T_v} \ge T$  then 12: Add an edge between the current node and the next node
- 13: end if

**Graph Pruning**. Like TANGO [21], we eliminate dead-end nodes by merging strongly connected components (SCCs). After this graph pruning process, the resulting motion graph becomes fully connected, enabling the generation of arbitrary long motion sequences.

As illustrated in Figure 4, the first and third graphs depict the original motion graphs constructed on the AIST++ [19] and FineDance [18] datasets, respectively. The second and fourth graphs display their corresponding pruned versions. From the results, we observe that the number of nodes is reduced by 17.0% and 47.5%, respectively, following the pruning process. This reduction ensures that all remaining nodes form a single connected component, thereby enabling seamless traversal from any node to any other node within the graph.

Motion Generation. Each node in the motion graph has a high out-degree, indicating that it can transition to multiple different nodes. Directly concatenating the current node with the next one often results in a noticeable discrepancy between the last frame of the current motion segment and the first frame of the subsequent one, leading to visually jarring transitions. To address this issue, we apply smoothing techniques to ensure a more natural and continuous motion flow. Following a similar approach to [40], we smoothed the joint angles across adjacent connected nodes to reduce discontinuities and enhance temporal coherence.

After this process, if the input is a long-term music clip, we can generate a motion sequence of the same duration, which we refer to as  $motion_{max}$ 

#### 3.3 Diffusion Model

After the motion graph process, the  $motion_{mq}$  can be used directly, but it is limited to the number of clips of the motion database, and the total motion performance is limited. Therefore, we need



Figure 5: The diffusion model. We propose the multi-condition pairwise fusion network to fuse the input conditions. The fused condition is then fed into the diffusion model through a cross-attention layer, enabling effective guidance of the generation process.

to augment the performance of the total motion. As the strong generation ability of the diffusion model [8], we use it to augment our music-to-dance motion.

An overview of our diffusion process is presented in Figure 5. It consists of a multi-condition fusion stage, implemented through the multi-condition pairwise fusion network, followed by the diffusion generation process, which builds upon the framework proposed in EDGE [35].

**Diffusion Formulation**. Diffusion models [8] define a consistent Markovian forward process that incrementally adds noise to clean sample data  $x_0^{1:L} \in q(x_0)$ , along with a corresponding reverse process that gradually removes noise from corrupted samples. For brevity, we denote the entire sequence at time step  $x_t$ . In the forward process, a predefined noise variance schedule  $\beta_t$  is used to control the amount of noise added at each step. The forward process can be formulated as follows:

$$q(x_{1:T}|x_0) = \prod_{t=1}^T q(x_t|x_{t-1}), q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1 - \beta_t} x_{t-1}, \beta_t I).$$
(2)

After T steps, the input data will be transformed to the noise distribution  $q(x_T)$ , which is usually a standard Gaussian distribution  $\mathcal{N}(0, I)$ . In the reverse process, the noise will be removed from the noisy sample  $x_T$ , and finally, the clean sample  $x_0$  will be obtained. In our method, we need to inject other conditions to modify the generation process. Thus our object is to model the distribution  $p(x_0|C)$  with a set of conditions C. Following [8], we directly predict the clean sample  $x_0$  from the noise distribution  $q(x_T)$  with the following objective:

$$\mathcal{L}_{simple} = \mathbb{E}_{x_0 \sim q(x|C), t \sim [1,T]} [\|x_0 - G(x_t, t, C)\|_2^2].$$
(3)

**Condition Model**. Similar to Beat-It [10], we also directly inject the conditions' feature into the diffusion model by the Cross-Attention module. Different from Beat-It [10], it proposes a hierarchical multi-condition fusion network to fuse the input conditions, while our approach propose a multi-condition pairwise fusion network to fuse the input conditions.

Our method incorporates four key conditioning signals: the input music, the extracted music beat, the top-k motion candidates, and the learned music embeddings. The input music is encoded using Jukebox [5]. The music beat is extracted following the approach proposed in Beat-It [10] [24]. The top-k motion candidates are retrieved through our previously described contrastive learning model and motion graph pipeline. Finally, the music embeddings are obtained via the contrastive learning model, which captures the semantic relationship between the audio and motion data.

The multi-condition pairwise fusion network first extracts the query (q), key (k), and value (v) features from each input condition. It then performs pairwise interactions by combining the query of one condition with the keys and values of all other conditions. Through this mechanism, the network enables rich and diverse fusion among multiple conditional inputs, allowing for more comprehensive and context-aware information integration during the generation process.

**Diffusion Model**. The diffusion model is built upon the architecture of [35] [27], with our primary modification focusing on the fusion of input conditional features.

**Losses.** Following [35] [34], we also incorporate the loss terms  $\mathcal{L}_{pos}$ ,  $\mathcal{L}_{vel}$  and  $\mathcal{L}_{contact}$  to enforce constraints on motion position, velocity, and foot-ground contact, respectively.

$$\mathcal{L}_{pos} = \frac{1}{N} \sum_{i=1}^{N} \|FK(x^{(i)}) - FK(\hat{x}^{(i)})\|_{2}^{2},$$
(4)

$$\mathcal{L}_{vel} = \frac{1}{N-1} \sum_{i=1}^{N-1} \| (x^{(i+1)} - x^{(i)}) - (\hat{x}^{(i+1)} - \hat{x}^{(i)}) \|_2^2,$$
(5)

$$\mathcal{L}_{contact} = \frac{1}{N-1} \sum_{i=1}^{N-1} \| (FK(x^{(i)}) - FK(\hat{x}^{(i)})) \cdot \hat{b}^{(i)} \|_2^2, \tag{6}$$

The total objective is as follows, while the setting of  $\lambda_{pos}$ ,  $\lambda_{vel}$ , and  $\lambda_{contact}$  are the same as EDGE [35]:

$$\mathcal{L}_{total} = \mathcal{L}_{simple} + \lambda_{pos} \mathcal{L}_{pos} + \lambda_{vel} \mathcal{L}_{vel} + \lambda_{contact} \mathcal{L}_{contact}.$$
(7)

After the diffusion model processing, we obtain the augmented generated motion, denoted as  $motion_{diff}$ .

# 4 Experiments

For the sake of simplicity, we refer to stage1 as the process that involves contrastive learning and motion graph processing, which generates  $motion_{mg}$ , and stage2 as the diffusion-based refinement process that produces  $motion_{diff}$ . We conduct our experiments on the dataset AIST++ [19] and FineDance [18].

#### 4.1 Dataset

**AIST++**. AIST++ [19] is a large-scale open-source 3D dance dataset containing 1,408 musicsynchronized motion sequences. It consists of 980 training sequences and 40 test sequences. The motion data is represented as 60-FPS 3D poses in SMPL [22] format. All experiments are conducted on the AIST++ dataset following the experimental setup outlined in [29].

**FineDance**. The FineDance dataset [18] provides 7.7 hours of 30-FPS motion data across 22 dance genres, with an average sequence length of 152.3 seconds—far exceeding AIST++'s 13.3 seconds. Captured using high-quality optical motion capture by professional dancers, it ensures both artistic quality and kinematic accuracy. We follow Lodge's protocol [17], generating sequences for 20 test tracks and evaluating 1024 frames. Its long-duration and rich choreography make it a robust benchmark for music-driven motion synthesis.

#### 4.2 Implementation Details

All experiments are conducted on a NVIDIA GeForce RTX 4090 GPU.

**Contrastive Learning Model**. We first pre-train the motion encoder with a batch size of 256, a maximum of 50 epochs, and a learning rate of 2e-4. The training time for this stage is approximately 4 hours. Subsequently, we train the music encoder using a batch size of 256, a maximum of 5000 epochs, and a learning rate of 1e-4. We employ AdamW [23] as our optimizer with the weight decay is 1e-2. The training process takes around 20 hours to complete.

**Diffusion Model**. For the diffusion model, we set the batch size as 64, the maximum number of epochs to 2000, and use a learning rate of 2e-4. We employ Adan [37] as the optimizer and use the Exponential Moving Average (EMA) [13] technique to enhance the stability of loss convergence. The model typically converges within approximately 10 hours.

#### 4.3 Evaluation Metrics

We evaluate our approach using three primary metrics: FID (Fréchet Inception Distance) for motion quality, DIV (Diversity Score) for motion diversity, and BAS (Beat Alignment Score) for music-motion synchronization accuracy. These metrics are applied to assess the performance of our method on both the AIST++ [19] and FineDance [18] datasets.

**Motion Quality**. This metric primarily evaluates the quality of the generated dance motion. It includes  $FID_k$  and  $FID_g$ , which denote the Fréchet Inception Distance (FID) computed using kinematic features and geometric features, respectively. The subscripts k and g indicate the type of feature used in the distance calculation.

**Motion Diversity**. This metric primarily evaluates the diversity of the generated motion. It includes  $DIV_k$  and  $DIV_g$ , which represent the diversity scores computed based on kinematic and geometric features [6], respectively. We follow the approach proposed in Bailando [29] to calculate these scores by measuring the average feature distances among the generated dance motions. The subscripts k and g denote the type of feature used for diversity estimation.

**Beat Alignment Score(BAS)**. To evaluate the accuracy of music-motion synchronization, we adopt the BAS (Beat Alignment Score) metric following the methodology in [29]. On the AIST++ [19] dataset, our approach achieves the highest score of 0.2874 in *stage1*. On the FineDance [18] dataset, it attains a score of 0.2631 after the *stage2* process, representing the best performance among the compared methods. The BAS is calculated using the following equation:

$$BAS = \frac{1}{|B^m|} \sum_{t^m \in B^m} exp\{-\frac{\min_{t^d \in B^d} ||t^d - t^m||^2}{2\sigma^2}\}.$$
(8)

## 4.4 Comparison to Existing Methods

We primarily compare our approach with state-of-the-art music-to-dance generation methods, including Bailando++ [30], Lodge++ [16], and EDGE [35]. We do not include Beat-It [10] in the comparison due to discrepancies in evaluation settings. Specifically, the reported ground-truth (GT) value for Beat-It is significantly higher than that of other methods (0.384 v.s. 0.2374), suggesting potential differences in metric computation or data normalization. Additionally, we were unable to obtain detailed evaluation protocols from the original work, which would be necessary for a fair and consistent comparison.

**Comparing on AIST++ [19] dataset**. As demonstrated in Table 1, on the AIST++ [19] dataset, we achieve the highest BAS score of 0.2874 on *stage1*. Following the diffusion refinement, our method obtains a higher FID<sub>k</sub> score, which is second only to Bailando++ [30] but significantly outperforms EDGE [35] and Lodge [17], both of which operate in the same long-term music-to-dance generation task. This demonstrates the effectiveness of our hybrid approach in balancing motion quality and temporal coherence.

**Comparing on FineDance [18] dataset.** As demonstrated in Table 2, our proposed framework demonstrates competitive performance in multiple metrics compared to the existing state-of-the-art methods in the FineDance [18] data set. Regarding motion quality, our method achieves the lowest FID<sub>k</sub> (10.51) and FID<sub>g</sub> (20.25) scores in *stage1*, indicating superior fidelity to ground-truth motion distributions. The *stage2* maintains strong quality (FID<sub>k</sub>=32.25) while achieving the highest BAS (0.2631), reflecting its ability to balance semantic alignment with musical inputs.

For motion diversity, the stage1 achieves the highest  $Div_k$  (10.67), outperforming all prior methods, including the baseline Lodge++ [16] ( $Div_k=5.53$ ). However, the stage2 shows a slight trade-off in diversity ( $Div_k=8.94$ ), which is still comparable to top-performing models like EDGE [35] ( $Div_k=8.13$ ). Notably, both stage processing exhibit distinct strengths: stage1 excels in maintaining high-quality and diverse motion patterns, while stage2 prioritizes music-motion semantic consistency as evidenced by its best-in-class BAS.

Method	Motion Quality		Motion Diversity		BAS↑
	$\operatorname{FID}_k\downarrow$	$\mathrm{FID}_g\downarrow$	$\mathrm{Div}_k\uparrow$	$\operatorname{Div}_g\uparrow$	
Ground Truth	17.10	10.60	8.19	7.45	0.2374
Li et al. [15]	86.43	43.46	6.85	3.32	0.1607
DanceNet [41]	69.18	25.49	2.86	2.85	0.1430
DanceRevolution [9]	73.42	25.92	3.52	4.87	0.1950
FACT [19]	35.35	22.11	5.94	6.18	0.2209
Bailando [29]	28.16	9.62	<u>7.83</u>	<u>6.34</u>	0.2332
Bailando++ [30]	17.59	<u>10.10</u>	8.64	6.50	0.2720
EDGE [35]	42.16	22.12	3.96	4.61	0.2334
Lodge [17]	37.09	18.79	5.58	4.85	0.2423
Ours(stage1)	30.17	19.80	5.82	6.07	0.2874
Ours(stage2)	<u>26.23</u>	17.66	5.62	3.79	0.2545

Table 1: Compare with SOTAs on the AIST++ [19] dataset. The best and runner-up values are bold and underlined, respectively.  $\downarrow$  means lower is better.  $\uparrow$  means upper is better.

Table 2: Compare with SOTAs on the FineDance [18] dataset. The best and runner-up values are bold and underlined, respectively.  $\downarrow$  means lower is better.  $\uparrow$  means upper is better.

Method	Motion Quality		Motion Diversity		BAS↑
	$\operatorname{FID}_k\downarrow$	$\mathrm{FID}_g\downarrow$	$\operatorname{Div}_k \uparrow$	$\operatorname{Div}_g\uparrow$	-~
Ground Truth	/	/	9.73	7.44	0.2120
FACT [19]	113.38	97.05	3.36	<u>6.37</u>	0.1831
MNET [12]	104.71	90.31	3.12	6.14	0.1864
Bailando [29]	82.81	28.17	7.74	6.25	0.2029
EDGE [35]	94.34	50.38	8.13	6.45	0.2116
Lodge [17]	50.00	35.52	5.67	4.96	0.2269
Lodge++ [16]	40.77	30.79	5.53	5.01	0.2423
Ours(stage1)	10.51	20.25	10.67	5.24	0.2612
Ours(stage2)	<u>32.25</u>	57.63	<u>8.94</u>	3.75	0.2631

Compared to the previous SOTA (Lodge++ [16]), our hybrid approach combining motion graph retrieval and diffusion-based refinement achieves significant improvements in both quality (FID<sub>k</sub> reduced by 73.1%) and semantic alignment (BAS increased by 11.6%), demonstrating the effectiveness of integrating retrieval-augmented generation with diffusion modeling for long-term music-driven dance synthesis.

# 5 Conclusion and Limitation

In this paper, we present a hybrid framework that combines motion graph retrieval with diffusionbased generation for long-term music-conditioned dance motion synthesis. By integrating contrastive learning, an optimized motion graph, and a DiT-based diffusion model, our method achieves superior performance in both motion quality and music alignment, as demonstrated on the AIST++ [19] and FineDance [18] datasets.

Despite these promising results, our approach has certain limitations. First, the motion diversity is still constrained by the pre-built motion graph. Second, complex or ambiguous musical inputs may challenge the current alignment mechanism. Lastly, the two-stage pipeline increases computational cost, which limits real-time deployment. Future work will focus on improving efficiency, reducing dependency on large motion libraries, and enabling interactive control over generated motions.

## References

- Jean-Baptiste Alayrac, Adria Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. *Advances in neural information processing systems*, 33:25–37, 2020.
- [2] Omid Alemi, Jules Françoise, and Philippe Pasquier. Groovenet: Real-time music-driven dance movement generation using artificial neural networks. *networks*, 8(17):26, 2017.
- [3] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460, 2020.
- [4] Kang Chen, Zhipeng Tan, Jin Lei, Song-Hai Zhang, Yuan-Chen Guo, Weidong Zhang, and Shi-Min Hu. Choreomaster: choreography-oriented music-driven dance synthesis. *ACM Transactions on Graphics (TOG)*, 40(4):1–13, 2021.
- [5] Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. Jukebox: A generative model for music. arXiv preprint arXiv:2005.00341, 2020.
- [6] Deepak Gopinath and Jungdam Won. Fairmotion-tools to load, process and visualize motion capture data. 2020.
- [7] Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. Momask: Generative masked modeling of 3d human motions. In *Proceedings of the IEEE/CVF Conference* on Computer Vision and Pattern Recognition (CVPR), pages 1900–1910, June 2024.
- [8] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.
- [9] Ruozi Huang, Huang Hu, Wei Wu, Kei Sawada, Mi Zhang, and Daxin Jiang. Dance revolution: Long-term dance generation with music via curriculum learning. In *International conference on learning representations*, 2020.
- [10] Zikai Huang, Xuemiao Xu, Cheng Xu, Huaidong Zhang, Chenxi Zheng, Jing Qin, and Shengfeng He. Beat-it: Beat-synchronized multi-condition 3d dance generation. In *European Conference on Computer Vision*, pages 273–290. Springer, 2024.
- [11] Sai Shashank Kalakonda, Shubh Maheshwari, and Ravi Kiran Sarvadevabhatla. Morag multifusion retrieval augmented generation for human motion. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025.
- [12] Jinwoo Kim, Heeseok Oh, Seongjean Kim, Hoseok Tong, and Sanghoon Lee. 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*, pages 3490–3500, 2022.
- [13] Frank Klinker. Exponential moving average versus moving exponential average. *Mathematische Semesterberichte*, 58:97–107, 2011.
- [14] 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, pages 1272–1279, 2022.
- [15] Jiaman Li, Yihang Yin, Hang Chu, Yi Zhou, Tingwu Wang, Sanja Fidler, and Hao Li. Learning to generate diverse dance motions with transformer. arXiv preprint arXiv:2008.08171, 2020.
- [16] Ronghui Li, Hongwen Zhang, Yachao Zhang, Yuxiang Zhang, Youliang Zhang, Jie Guo, Yan Zhang, Xiu Li, and Yebin Liu. Lodge++: High-quality and long dance generation with vivid choreography patterns. arXiv preprint arXiv:2410.20389, 2024.

- [17] 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*, pages 1524–1534, 2024.
- [18] 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*, pages 10234–10243, 2023.
- [19] 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*, pages 13401–13412, 2021.
- [20] Haiyang Liu, Zhan Xu, Fa-Ting Hong, Hsin-Ping Huang, Yi Zhou, and Yang Zhou. Video motion graphs. *arXiv preprint arXiv:2503.20218*, 2025.
- [21] Haiyang Liu, Xingchao Yang, Tomoya Akiyama, Yuantian Huang, Qiaoge Li, Shigeru Kuriyama, and Takafumi Taketomi. Tango: Co-speech gesture video reenactment with hierarchical audio motion embedding and diffusion interpolation. *arXiv preprint arXiv:2410.04221*, 2024.
- [22] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. ACM Trans. Graphics (Proc. SIGGRAPH Asia), 34(6):248:1–248:16, October 2015.
- [23] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.
- [24] 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.
- [25] Ferda Ofli, Engin Erzin, Yücel Yemez, and A Murat Tekalp. Learn2dance: Learning statistical music-to-dance mappings for choreography synthesis. *IEEE Transactions on Multimedia*, 14(3):747–759, 2011.
- [26] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings* of the IEEE/CVF international conference on computer vision, pages 4195–4205, 2023.
- [27] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.
- [28] Qiaosong Qi, Le Zhuo, Aixi Zhang, Yue Liao, Fei Fang, Si Liu, and Shuicheng Yan. Diffdance: Cascaded human motion diffusion model for dance generation. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 1374–1382, 2023.
- [29] Li Siyao, Weijiang Yu, Tianpei Gu, Chunze Lin, Quan Wang, Chen Qian, Chen Change Loy, and Ziwei Liu. Bailando: 3d dance generation via actor-critic gpt with choreographic memory. In CVPR, 2022.
- [30] Li Siyao, Weijiang Yu, Tianpei Gu, Chunze Lin, Quan Wang, Chen Qian, Chen Change Loy, and Ziwei Liu. Bailando++: 3d dance gpt with choreographic memory. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12):14192–14207, 2023.
- [31] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. pmlr, 2015.
- [32] 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*, pages 1598–1606, 2018.

- [33] Guy Tevet, Brian Gordon, Amir Hertz, Amit H Bermano, and Daniel Cohen-Or. Motionclip: Exposing human motion generation to clip space. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII*, pages 358–374. Springer, 2022.
- [34] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022.
- [35] 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, pages 448–458, 2023.
- [36] Ho-Hsiang Wu, Prem Seetharaman, Kundan Kumar, and Juan Pablo Bello. Wav2clip: Learning robust audio representations from clip. In *ICASSP 2022-2022 IEEE International Conference* on Acoustics, Speech and Signal Processing (ICASSP), pages 4563–4567. IEEE, 2022.
- [37] Xingyu Xie, Pan Zhou, Huan Li, Zhouchen Lin, and Shuicheng Yan. Adan: Adaptive nesterov momentum algorithm for faster optimizing deep models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.
- [38] Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. Motiondiffuse: Text-driven human motion generation with diffusion model. *IEEE* transactions on pattern analysis and machine intelligence, 46(6):4115–4128, 2024.
- [39] Mingyuan Zhang, Xinying Guo, Liang Pan, Zhongang Cai, Fangzhou Hong, Huirong Li, Lei Yang, and Ziwei Liu. Remodiffuse: Retrieval-augmented motion diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 364–373, 2023.
- [40] Yang Zhou, Jimei Yang, Dingzeyu Li, Jun Saito, Deepali Aneja, and Evangelos Kalogerakis. Audio-driven neural gesture reenactment with video motion graphs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3418–3428, 2022.
- [41] 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.