EDMG: Towards Efficient Long Dance Motion Generation with Fundamental Movements from Dance Genres

Jinming Zhang*
School of Computer Science and Engineering
Nanjing University of Science and Technology
Nanjing, China
zjm@njust.edu.cn

Yunlian Sun*
School of Computer Science and Engineering
Nanjing University of Science and Technology
Nanjing, China
yunlian.sun@njust.edu.cn

Hongwen Zhang
School of Artificial Intelligence
Beijing Normal University
Beijing, China
zhanghongwen@bnu.edu.cn

Jinhui Tang†
College of Artificial Intelligence
Nanjing Forestry University
Nanjing, China
tangjh@njfu.edu.cn

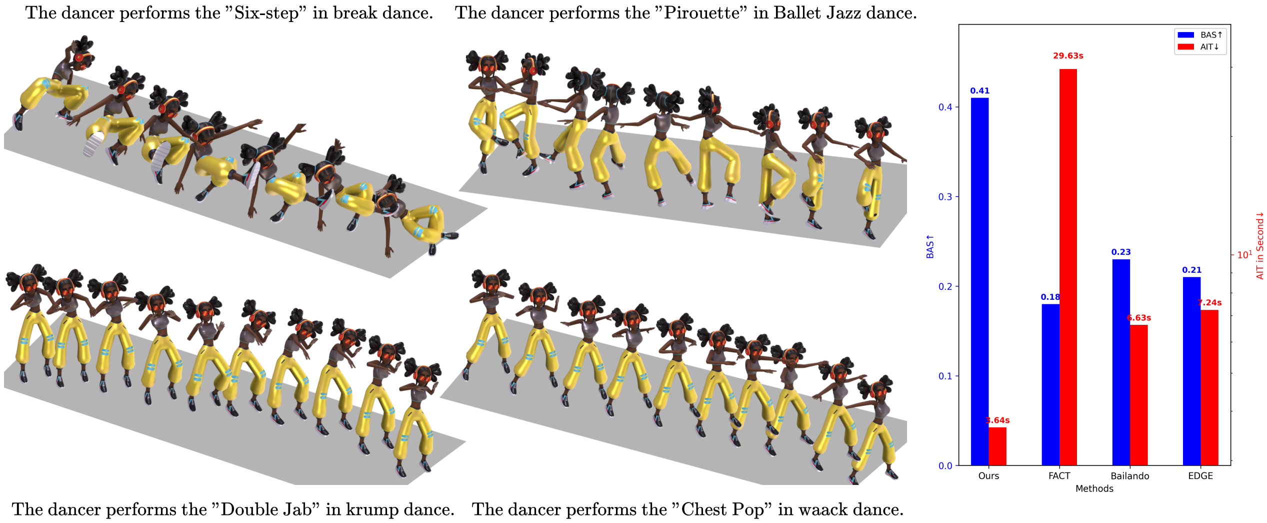

The dancer performs the "Six-step" in break dance.
The dancer performs the "Pirouette" in Ballet Jazz dance.
The dancer performs the "Double Jab" in krump dance.
The dancer performs the "Chest Pop" in waack dance.

Figure 1: EDMG can efficiently generate diverse and physically plausible dance movements based on specific textual prompts describing dance motions, while integrating musical conditions for creation. As illustrated, "Six-step" is a fundamental movement in break dance, while "Pirouette" is a fundamental movement in Ballet Jazz dance. These movements showcase the unique dance styles of different dance genres. EDMG achieves significant performance improvements in long sequence modeling and motion generation through multimodal conditioning and efficient model design based on Mamba2 [3].

Abstract

Dance is an important art form in human culture, but creating new dances can be both challenging and time-consuming. In this paper, we propose a novel dance choreography framework, EDMG, designed to efficiently generate creative and long-lasting dance sequences conditioning on music and dance descriptions. In the first stage, we propose a flexible dance diffusion method, combined with dance genre description and descriptions of fundamental movements to generate the dance sequences. To achieve high computational efficiency and inference speed, EDMG designs a lightweight denoising module by using selective parallel scanning algorithm from Mamba2. This Parallel Mamba Denoiser reduces significantly the number of parameters and accelerates remarkably both the learning and inference processes. In the second stage, by designing a smoothing module with a long receptive field, we mitigate joint error accumulation that causes jittering movements and foot sliding, thereby enhancing the fluency and visual appeal of the dance movements. Furthermore, we extend the AIST++ dataset by adding detailed descriptions of dance genres and fundamental movements, using the Large Language Model (LLM). These descriptions further improve the choreography generation. EDMG is validated through extensive experiments, demonstrating that our method can both effectively and efficiently generate long-term dances suitable for various dance genres. Project URL: https://github.com/neymar277/EDMG.

*Equal contribution
†Corresponding author

MM '25, Dublin, Ireland.

ACM ISBN 979-8-4007-2035-2/2025/10
https://doi.org/10.1145/3746027.3755677

CCS Concepts

• **Computing methodologies → Procedural animation**; *Motion processing*.

Keywords

Human Motion Generation, State Space Model, Multimodal Fusion

ACM Reference Format:

Jinming Zhang, Yunlian Sun, Hongwen Zhang, and Jinhui Tang. 2025. EDMG: Towards Efficient Long Dance Motion Generation with Fundamental Movements from Dance Genres. In *Proceedings of the 33rd ACM International Conference on Multimedia (MM '25), October 27–31, 2025, Dublin, Ireland.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3746027.3755677

1 Introduction

Dance, as an ancient art form, resonates with various age groups and cultures due to its artistic and aesthetic value, playing a crucial role in the cultural and entertainment sectors. Traditional dance choreography requires a complex balance among aesthetic movements, emotional expression, and precise synchronization with musical beats, a process that is often expensive, time-consuming, and challenging even for experienced artists. In the digital age, with the growing demand for 3D dance content, traditional dance creating methods, e.g., motion capture [40, 46, 48, 49], are increasingly inefficient. Therefore, music-based automatic dance sequence generation has become an important research topic, which not only helps the film and television industry quickly produce 3D dance assets, but also assists dancers in their choreography creation.

In recent years, music-driven dance generation has made significant progress with the development of generative models [1, 5, 34]. With their exceptional content generation capabilities, these models have marked a paradigm shift in the field. Tseng et al. [41] introduced a transformer-based diffusion model, which sets a new benchmark in the transformation of music to dance, demonstrating the potential of these models to improve the quality of the generated dance sequences. However, transformer-based models [22, 41, 44] rely largely on subtle positional encodings to capture the order of input elements. At the same time, their computational complexity grows quadratically with the length of the input sequence, resulting in high computational overhead and inefficient sequence generation during the diffusion iteration process. This presents challenges for efficiently generating long-sequence dances.

Previous studies have been attempting to address these challenges. For example, to reduce computational costs during training, some autoregressive architecture-based methods [22, 37, 41] tried to train dance generation models within short time windows (typically 2-5 seconds with 30 FPS). It should be noted that many fundamental movements usually last for several seconds. For example, the "Indian step" in break dance, as a continuous dance movement, represents the smallest unit of the dance content and has the property of not being disassembled. The generative model based on a small time window cannot effectively capture these dance elements because it segments continuous dance movements, making it difficult to effectively learn choreographic patterns of long sequences. In addition, the decomposition of continuous movement into smaller segments can cause the loss of local information, leading to the generation of dance movements that do not meet the requirements of the actual choreography.

With the aim of generating long-term dance sequences, based on the diffusion model [11], we propose a new denoising module, Parallel Mamba Denoiser (PMD). PMD introduces, for the first time, the selective parallel scanning algorithm from Mamba2 [3] to human motion generation. It selectively compresses data into smaller state spaces through State Space Models (SSMs) [9, 10]. PMD also employs hardware-aware algorithm optimization to reduce IO transfers between different levels of the GPU memory hierarchy as well as memory consumption. Owing to these designs, PMD can efficiently generate dance motion sequences at 60 FPS for 10 seconds. Additionally, the local information between adjacent movements is of significant importance to realistic dance generation. However, recent methods like Lodge [23] often overlook the importance of modeling local frame-to-frame information. To address this issue, we further adopt a lightweight convolutional module specifically for processing local information interaction.

For PMD, directly applying Mamba2's selective scanning to long-sequence dance generation can cause jittery outputs. This happens because dance movements typically involve large-amplitude and high-acceleration joint movements. These characteristics can amplify the error accumulation effect. Even small prediction deviations can rapidly expand through consecutive frames [47]. We solve this by adopting a Temporal Refinement Module (TRM) after the PMD module. In order to suppress errors, TRM optimizes joint positions, velocities, and accelerations temporally. It should be noted that, these optimization objectives are all measured in the linear space of joint positions. However, SMPL [27] skeleton rotations are non-linear. We thus convert these rotations to 3D coordinates using forward kinematics function. This module is particularly suitable for dance movements with dramatic changes, which can improve fluency and physical plausibility.

Finally, to increase dancers' choreography freedom, we employ the Large Language Model (LLM) to generate text descriptions of ten dance genres as well as descriptions of fundamental movements for each dance genre. Then we select 1635 motion sequences from AIST++ [22] using a 10s window. For each motion sequence, we add its corresponding dance genre description and descriptions of fundamental movements. With these descriptions, dancers can customize and generate personalized dance movements, thus making choreography creation more flexible and diversified. At the same time, the model can choreograph based on fundamental movements. This helps prevent overfitting and ensures the generation of more realistic and coherent motion sequences. Our contributions are summarized as follows:

- To achieve efficient long-term dance sequence generation, based on the diffusion model, we propose a new denoising module, the Parallel Mamba Denoiser. This module adopts the selective parallel scanning mechanism from Mamba2 and applies it to memorizing long dance sequences.
- To address the jittery outputs caused by directly applying Mamba2's selective scanning mechanism to long-term dance generation, we design TRM. This module calculates 3D joint coordinates through fully connected layers with residual connections. These connections provide a wider temporal

field of view. TRM optimizes the position, velocity, and acceleration of dance movements, which effectively improves the smoothness.

- To enhance dancers' choreography freedom, we use LLM to generate text descriptions of dance genres as well as descriptions of fundamental movements for each dance genre. By utilizing these descriptions, we can generate more flexible fundamental dance movements.

2 Related Work

2.1 Dance Generation

In recent years, with the development of deep learning technologies, research on generating dance synchronized with music has attracted widespread attention. Current research methods encompass a variety of frameworks and model types, including motion-graph approaches [31, 32], sequence models [17, 22], VQ-VAE models [37, 52], Generative Adversarial Networks (GANs) [18], and diffusion models [24, 41]. Traditional motion graph methods calculate matching scores between music and dance segments using cross-modal retrieval networks, and construct motion graphs based on these scores and predefined choreography rules to generate long dance sequences. However, these methods are limited by predefined rules, making them difficult to adapt to different dance genres. Furthermore, they fail to achieve fine-grained beat alignment, resulting in dances that lack diversity and creativity.

Sequence models such as LSTM [12] and Transformer [43] generate dance sequences autoregressively using music and past sequences as input. For example, FACT [22] used a Transformer model to generate dance frames conditioned on music and seed motions, but still faced issues like error accumulation and motion freezing. Danceformer [21] adopted a two-stage framework to generate key poses and interpolate between them. However, due to its single-condition nature, danceformer lacks flexibility in beat choreography and key pose control.

VQ-VAE models [42], such as Bailando [37] and TM2D [6], enhanced rhythm control and semantic generation capabilities in dance generation through encoding and generating dance movements. Although these methods can capture global choreography patterns, their reliance on pre-trained codebooks limits dance diversity and complicates the optimization of beat alignment and motion generation.

GANs [7] have also been applied to dance generation. For example, MNET [18] combined a Transformer-based generator with a multi-genre discriminator to generate dances. Although GAN-based methods can produce realistic dance animations, they often face challenges such as mode collapse and training instability during the training process.

Diffusion models [11, 36] have also been applied to dance generation, such as EDGE [41] and Logde [23], which used diffusion techniques to generate diverse and high-quality dance clips. However, these methods still face challenges in generating long-term dance movements and efficient generation.

To further improve the flexibility and diversity of generated dances, recent studies have attempted dance generation guided by multiple conditions, adding additional conditions beyond music during the generation process, such as dance style labels, text descriptions, and keyframes. For example, DGSDP [44] introduced dance style descriptions to generate predefined dance styles, while Beat-it [13] used keyframes and music beats to address beat alignment and key pose control issues. Despite improving generation controllability, these multi-condition models still suffer from challenges in simultaneously generating both global and local information. In addition, they fail to allow dancers to customize fundamental dance movements corresponding to specific genres.

In summary, while significant progress has been made in the field of dance generation, generating high-quality, long-term dance sequences that maintain global consistency remains a challenge.

2.2 State Space Models

State space models [9, 10] are an emerging class of models that have seen widespread applications in recent years, especially in sequence modeling tasks, where they have shown great potential. Inspired by the classical state space model [14], SSMs offer an efficient solution, particularly in modeling long sequence data. Traditional Transformer models suffer from computational efficiency issues when handling long sequences, while SSMs successfully alleviate this issue with their linear time complexity and sequence modeling capability.

The Structured State Space Sequence (S4) model [9], as an efficient SSM architecture, has garnered attention for its linear scaling relationship with sequence length. The S4 model has demonstrated its advantages in handling long sequences, making it a key focus in sequence modeling research. In recent years, Mamba [8] further enhanced the model's performance by introducing data-dependent SSM layers, surpassing Transformers, especially in large-scale data processing tasks, and showing strong potential. The success of Mamba has sparked deep exploration of its applications across different domains.

For example, Vision Mamba [51] and VMamba [26] have developed unique scanning techniques for two-dimensional image processing tasks, fully leveraging the advantages of SSM in image data. Through innovative scanning mechanisms, Mamba and related models have made significant breakthroughs in computational efficiency and global perception, enabling better handling of complex visual tasks.

Moreover, the efficient computation and memory mechanisms of SSMs have been extended to higher-dimensional data processing. For instance, Mamba-ND [25] explored the combination of SSM with different scanning directions, successfully applying the Mamba model to higher-dimensional tasks. In the field of image generation, DiffuSSM [45] combined SSM with diffusion denoisers, preserving image details while improving generation efficiency.

Overall, the advantages of SSMs are evident not only in their ability to efficiently handle long sequence data but also in their flexible structural design, which can be customized and optimized according to the needs of different tasks. As more research progresses, SSMs are showing strong adaptability and potential across various application scenarios, especially in tasks like visual recognition, language understanding, and sequence generation, establishing themselves as a key technological foundation.

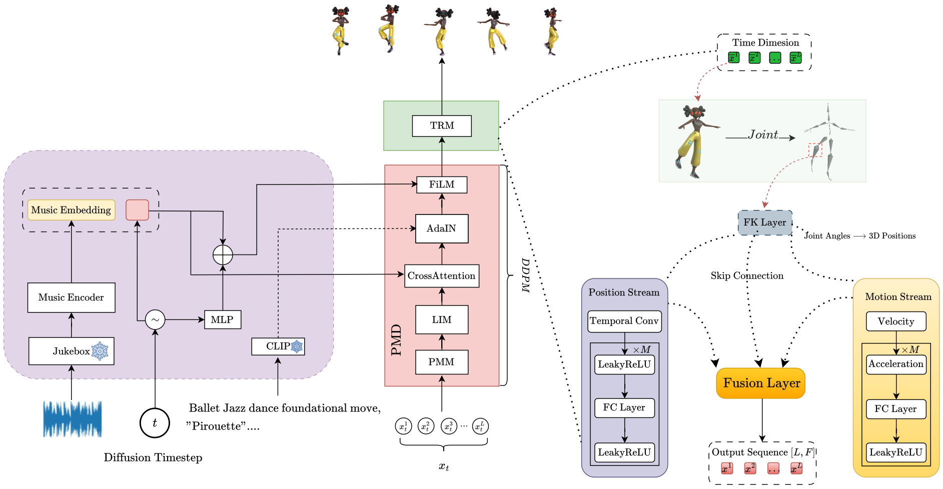


Figure 2: EDMG Pipeline Overview: EDMG learns to denoise dance sequences x from timestep t to 0, guided by dance genre description and descriptions of fundamental movements. The model framework is divided into two parts: PMD mainly denoises and outputs unrefined motion sequences $\overline{x}$, which serve as input for the second-stage TRM module. Our goal is to generate long-term dances featuring genre-specific fundamental movements. The figure demonstrates the process of generating the fundamental "Pirouette" movement in Ballet Jazz dance.

3 Method

3.1 Preliminaries

Selective State Space Model. The SSM-based models, e.g., S4 and Mamba, are inspired by the continuous system. This system maps a 1-D function or sequence $x \in \mathbb{R} \mapsto y \in \mathbb{R}$ through a hidden state h. Mamba is the discrete versions of the continuous system, which include a timescale parameter to transform the continuous evolution parameters A and projection B to discrete parameters $\overline{A} \in \mathbb{R}^{L\times F\times N}$, $\overline{B} \in \mathbb{R}^{L\times F\times N}$. L is the input sequence length, F is the feature dimension of input x, and N is the state dimension of $\overline{A}$ and $\overline{B}$. The commonly used method for transformation is zero-order hold (ZOH).

After the discretization, the discretized version can be written as:

$$h^l = \overline{A}h^{l-1} + \overline{B}x^l \tag{1}$$

$$y^l = Ch^l \tag{2}$$

where $C \in \mathbb{R}^{L\times F\times N}$ as the projection parameters.

In our approach, we adopt the Mamba model and incorporate several improvements inspired by Mamba2 [3]. Specifically, we introduce the Structured State Space Duality (SSD) and Multi-Input SSM mechanisms. SSD demonstrates the duality between SSM and special attention, allowing the use of matrix optimization algorithms from transformers in SSM calculations. The Multi-Input SSM mechanism implements parameter sharing and parallel input operations when generating SSM parameters ($\bar{A}$, $\bar{B}$, C) from x. These enhancements significantly improve the model's parallelism and scalability while enabling tensor parallelism, allowing the model to scale to larger dimensions and longer contexts, thereby substantially boosting training performance and overall efficiency.

Diffusion Model. The diffusion model consists of a diffusion process and a denoising process. The diffusion process perturbs the ground truth data x into x_t over t steps. Following [11], we simplify this multi-step diffusion process into one step, which can be formulated as:

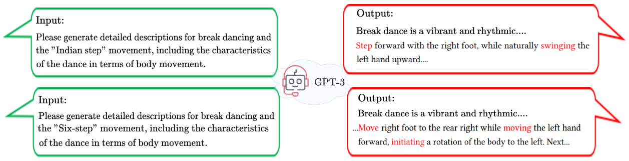
$$q(x_t|x) = \mathcal{N}(\sqrt{\alpha_t}x, (1-\alpha_t)I) \tag{3}$$

where α_t is within the range of $(0, 1)$ and follows a monotonically decreasing schedule. α_t converges to 0 as t goes to infinity, making x_t converge to a sample from the standard normal distribution I. The denoising process employs a base network f_θ to gradually recover the input x, generating $\overline{x}$ conditioned on given signal c and denoising timestep t. Instead of predicting the noise, we directly predict x like [41]. Therefore, the training process can be formulated as:

$$\mathcal{L}_{recon} = \mathbb{E}_{x,t}\left[\|x-\overline{x}\|_2^2\right] = \mathbb{E}_{x,t}\left[\|x - f_\theta(x_t,t,c)\|_2^2\right] \tag{4}$$

3.2 EDMG

Overview Architecture. The proposed EDMG utilizes a two-stage choreographic framework as shown in Figure 2. In the first stage, a multi-conditional denoising network PMD is designed to follow the Denoising Diffusion Probabilistic Model (DDPM) diffusion denoising process. The noised data is first converted into an ordered sequence after the serialization process. The sequence is then encoded by the well-designed Mamba2 block, i.e., Parallel Mamba Module (PMM), which efficiently compresses the state space of the sequence to memorize all of them. Subsequently, the sequence is fed into a lightweight Local Information Modeling module (LIM) to downsample the sequence through 1-D convolution [20]. Meanwhile, the system pre-processes music tokens and text tokens, and


Input:
Please generate detailed descriptions for break dancing and the ”Indian step” movement, including the characteristics of the dance in terms of body movement.

Input:
Please generate detailed descriptions for break dancing and the ”Six-step” movement, including the characteristics of the dance in terms of body movement.

Output:
Break dance is a vibrant and rhythmic....
Step forward with the right foot, while naturally swinging the left hand upward....

Output:
Break dance is a vibrant and rhythmic....
...Move right foot to the rear right while moving the left hand forward, initiating a rotation of the body to the left. Next...

Figure 3: Dance generation with dance genre and fundamental movement hints. When a dancer wants to choreograph a specific genre of dance (e.g., Break Dance) or wants to generate fundamental dance movements like “Indian step” or “Six-step” from break dance, he/she can use EDMG with description prompts to guide the generation based on multiple conditions, thereby creating choreographic works that both conform to specific dance genre and incorporate the desired fundamental movements.

inputs the downsampled sequence to a Multi-Head Cross Attention module for conditional generation. In the second stage, TRM process the motion generated from the first stage using forward kinematics function to extract joint positions, velocities, and accelerations. Each joint is then optimized and smoothed across the global sequence.

Text Prompts. It is well known that there are many different dance genres. To obtain detailed descriptions of the fundamental movements for each music genre, we use GPT-3 [2] to generate descriptions for dances. We use the following prompt to acquire these descriptions: “Please generate detailed descriptions for dance genre s and fundamental dance movement f, including the characteristics of the dance in terms of body movement”, where ‘s’ represents the dance genre and ‘f’ represents the fundamental movement. In Figure 3, we give two examples. With these descriptions, our model can enhance dancers’ choreography freedom.

Motion Representation. We represent the dance as a sequence of poses using the 24-joint SMPL skeleton [28], with a 6-DOF rotation representation [50] for each joint and a single root translation: $w \in \mathbb{R}^{24\cdot 6+3=147}$. For the heel and toe of each foot, we also include binary contact labels: $b \in \{0,1\}^{2\cdot 2=4}$. The total dance pose representation is therefore $p = \{b, w\} \in \mathbb{R}^{4+147=151}$. EDMG uses a diffusion-based framework to learn to synthesize a sequence of L frames, $x \in \mathbb{R}^{L\times 151}$, conditioned on arbitrary music features c extracted through a Jukebox encoder [4], as well as text conditions g extracted via a CLIP encoder [33].

3.3 Parallel Mamba Denoiser

The parallel mamba module, as shown in Figure 4, is an architecture designed for efficient sequence modeling through a parameter sharing mechanism. In the parallel SSM model, each input head is assigned to an independent dynamic parameter $\overline{A}$, while the input projection parameter $\overline{B}$ and output projection parameter C are shared. The input motion sequence x is first projected to a latent dimension F, then split and transformed into $\hat{x} \in \mathbb{R}^{L\times H\times P}$, where H is the number of heads, P is the feature dimension of each head. For all heads $i = 1, 2, ..., H$, each has its independent dynamic parameter $\overline{A}_i \in \mathbb{R}$. The input projection parameter $\overline{B} \in \mathbb{R}^{L\times N}$ and output projection parameter $C \in \mathbb{R}^{L\times N}$ are shared among all heads. N is the state dimension.

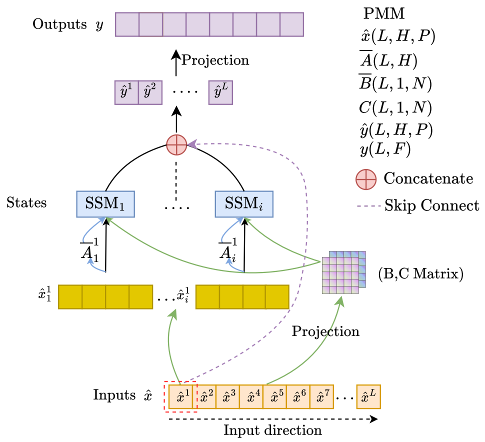


Figure 4: The PMM module encodes the input motion sequence x^l through its powerful content memorizing capability and efficient parallel computing capacity; PMM has multiple parallel SSM modules, where the $\overline{B}$ and C matrices share parameters across multiple SSM modules, the $\overline{A}$ matrix is initialized before input, and each SSM has its own unique $\overline{A}_i$ matrix.

For each head i at frame l, with input $\hat{x}_i^l$, the state update is given by:

$$h_i^l = \overline{A}_i^l h_i^{l-1} + \overline{B}^l \hat{x}_i^l \tag{5}$$

The output for each head y_i^l is:

$$y_i^l = C^l \cdot h_i^l \tag{6}$$

The outputs from all heads are combined through a concatenation operation and then projected to produce the final output $y \in \mathbb{R}^{L\times F}$. To achieve local information modeling while reducing the number of parameters and FLOPs, LIM utilizes 1-D convolution to perform dimensionality reduction on y along the L dimension.

We use the music input c as the main guidance. By using cross attention to combine y with musical conditions, we guide the generation of dance movements that align with the musical conditions. We then upsample y to $\mathbb{R}^{L\times F}$. Additionally, we employ the textual condition g as the secondary guidance to facilitate the customization of specific dance movement styles. The design considers both the coordination of the music and the specification of the textual style. we adopt the adaptive instance normalization used in StyleGAN [15, 16] for integrating the text condition g.

The training objective is:

$$\mathcal{L}_{recon} = \mathbb{E}_{x,t}\left[\|x - \overline{x}\|_2^2\right] = \mathbb{E}_{x,t}\left[\|x - f_\theta(x_t, t, c, g)\|_2^2\right] \tag{7}$$

where f_θ denotes the PMD block, $\overline{x}$ denotes the predicted results from PMD, x_t is the sequence after noise addition, t is the denoising time step. During training, we randomly drop the conditioning input with a 25% probability. This exposes the model to unconditioned generation scenarios and improves generalization to unseen prompts. We also incorporate auxiliary losses similar to those in Tevet et al. [39], which enforce physical properties [35, 38]: joint positions, velocities, and foot velocities through our Contact Consistency Loss:

$$\mathcal{L}_{\text{joint}} = \frac{1}{L}\sum_{l=1}^{L} \|FK(x^l) - FK(\overline{x}^l)\|_2^2 \tag{8}$$

$$\mathcal{L}_{\text{vel}} = \frac{1}{L-1}\sum_{l=1}^{L-1} \|(x^{l+1} - x^l) - (\overline{x}^{l+1} - \overline{x}^l)\|_2^2 \tag{9}$$

$$\mathcal{L}_{\text{contact}} = \frac{1}{L-1}\sum_{l=1}^{L-1} \|(FK(\overline{x}^{l+1}) - FK(\overline{x}^l)) \cdot \overline{b}^l\|_2^2 \tag{10}$$

where l denotes the l-th frame, L is the number of motion frames, $\overline{x}$ is the predicted motion sequence, and $\overline{b}$ is the predicted foot contact label. The function FK$(\cdot)$ denotes forward kinematics.

3.4 Temporal Refinement Module

In long-term motion generation, the generated motion sequences often suffer from jitter and noise artifacts. Traditional smoothing methods face two main challenges: over-smoothing that leads to detailed loss, and global incoherence due to localized processing. To address these issues, we propose TRM, a novel motion sequence smoothing approach based on Fully Connected Neural Networks (FCN). Our method focuses exclusively on temporal refinement with global receptive fields, smoothing each joint while addressing both local detail preservation and global motion coherence.

Temporal refinement module proposes a dual-branch network architecture that models and optimizes motion sequences from both positional and kinematic perspectives. Given the motion sequence $\overline{x}$ arranged along the temporal dimension from the first stage, the forward kinematics function transforms $\overline{x}$ into a linear space $p \in \mathbb{R}^{L\times(24\times3)}$. L is the length of the dance sequence, 24 is the number of joints, and 3 represents the three-dimensional coordinates of each joint. The Position Stream processes spatial features through FCN with global receptive fields , while the Motion Stream computes velocity $v_j^l = p_j^l - p_j^{l-1}$ and acceleration $a_j^l = v_j^l - v_j^{l-1}$ of the input for position refinement. j represents the joint index. The optimization objective is to minimize both position and acceleration errors:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{pos}} + \mathcal{L}_{\text{acc}} \tag{11}$$

where

$$\mathcal{L}_{\text{pos}} = \frac{1}{L\times J}\sum_{l=1}^{L}\sum_{j=1}^{J} |p_j^l - \overline{p}_j^l| \tag{12}$$

$$\mathcal{L}_{\text{acc}} = \frac{1}{(L-2)\times J}\sum_{l=1}^{L-2}\sum_{j=1}^{J} |a_j^l - \overline{a}_j^l| \tag{13}$$

where $\overline{p}_j^l$ is the predicted pose, $\overline{a}_j^l$ is the computed acceleration from predicted pose. $J = 24$ is the number of joints.

4 Experiment

4.1 Experimental Setups

Dataset: In this work, we use the AIST++ dataset [22], which consists of 1,408 high-quality dance motions paired with music from various genres. To meet the task requirements (i.e., long-term dance generation), we divide all training samples into segments of 10 seconds at 60 FPS. Music and motion segment pairs with durations less than 10 seconds are repeatedly spliced to reach the required length. The dance sequences in the dataset are divided into 10 dance genres, each corresponding to dozens of fundamental dance movements. Our annotations are shown in Table 1. In the experiments, we use the train/test splits provided by the original dataset. FineDance [24] is also a widely used dance dataset, which includes 211 dance motion sequences and contains 7.7 hours of dance data with a frame rate of 30fps.

Table 1: Statistics of sequence counts and corresponding fundamental movements for each dance genre.

Genre	Sequences	Movements
Breaking	172	26
Popping	158	31
Locking	170	22
Waacking	165	26
Medium Hip-hop	160	23
LA-style Hip-hop	168	18
House dance	155	21
Krumping	163	24
Street Jazz	159	22
Ballet Jazz	165	24

Implementation details: The model is trained on 2 NVIDIA RTX 3090 GPUs for 29 hours with a batch size of 32. For the PMD module, the dimension of the motion latent vector F is 512, the number of heads H is 128, the projection dimension of the state matrix N is 64, the down-sampling rate of the local information modeling module is set to 2, the number of attention heads is 4, and the dropout is 0.1. For the TRM module, the number of modules M is set to 5, using LeakyReLU [30] as the non-linear activation function. For DDPM [11], the diffusion time steps are configured to 1000. We use Adam [19] as the optimizer, with a learning rate set to 0.0002.

4.2 Evaluation Metrics

We comprehensively evaluate our method from the following aspects. **Motion Quality** primarily measures the quality of dance movements, analyzing their natural smoothness and physical realism. It includes FID$_k$ [22], FID$_g$ [22], Physical Foot Contact (PFC) [41], and Physical Body Contact (PBC) [29]. FID$_k$ and FID$_g$ are the Fréchet Inception Distances (FID) of generated dance and ground truth dance in the dataset, with subscripts k and g representing FID calculated using kinematic and geometric features, respectively. EDGE introduced the PFC score, which assesses the plausibility of dance movements directly through the acceleration of the hips and the velocity of the feet. PBC extends the evaluation to the entire body, including the neck and hands, to create a body contact score. **Motion diversity** primarily focuses on the richness and diversity of movements, including DIV$_k$ and DIV$_g$. **Beat Alignment Score (BAS)** [22] measures the degree of rhythmic alignment between music and dance. It evaluates how well the timing of dance movements corresponds to the beats of the music. **Model Efficiency** is evaluated based on aspects such as model average inference time (AIT) and GPU utilization. **Prompt-to-motion alignment** uses MultiModal Distance (MM) to measure how accurately the generated motion reflects the given textual condition. **User Study**

Table 2: Comparison with the state-of-the-art methods on AIST++. ↑ indicates higher is better, ↓ indicates lower is better.

Method	Motion Quality				Motion Diversity		BAS↑	AIT↓	MM↓
	FID_k ↓	FID_g ↓	PFC↓	PBC↓	Div_k ↑	Div_g ↑			
Ground Truth	6.83	6.53	1.23	2.84	6.53	7.16	0.51	/	3.7
FACT [22]	55.35	30.34	1.42	8.53	5.67	5.94	0.18	29.63s	9.3
Bailando [37]	28.17	9.43	1.32	8.92	6.17	**9.42**	0.23	6.63s	8.6
Lodge [23]	37.09	18.79	1.07	3.23	5.58	4.85	0.24	5.64s	6.9
EDGE [41]	14.81	7.69	2.34	5.71	**9.06**	7.49	0.21	7.24s	6.5
EDMG	**10.69**	**7.35**	**0.96**	**2.34**	8.75	7.35	**0.41**	**3.64s**	**4.5**

Table 3: Comparison with the state-of-the-art methods on FineDance.

Method	FID_k ↓	FID_g ↓	BAS ↑
EDGE	94.34	50.38	0.21
Lodge	45.56	**34.29**	0.23
EDMG	**40.23**	35.75	**0.32**

collects human ratings on Genre Matching (GM), Rhythm Matching (RM), and Diversity (Div) to further assess the generated results.

4.3 Comparison to Existing Methods

As shown in Table 2, we compare EDMG with four state-of-the-art methods. FACT [22] is a classic autoregressive generation algorithm, while Bailando [37] is an outstanding dance generation network based on VQ-VAE and GPT, demonstrating excellent performance. Lodge [23] is a coarse-to-fine two-stage diffusion generation method. EDGE [41] is a diffusion-based dance generation method that supports dance editing operations such as inbetweening, continuation, and multi-conditional generation, showing high-quality generation in short-term dance generation. For a fair comparison, We train FACT, Bailando, EDGE and our EDMG under the same experimental setting, using sequences of 10 seconds at 60 FPS. For Lodge, we directly adopt the results reported in their original paper. Compared to these methods, EDMG achieves significant improvements in motion quality, beat alignment, and efficiency for long sequence dance generation. In our experimental results, we achieve the best scores in FID_k and FID_g for motion generation quality, demonstrating that our method can generate high-quality motion sequences. We observe significant improvements in physical plausibility metrics, with PFC and PBC scores reduced to 0.96 and 2.34, respectively. Notably, in terms of lower body measured by PFC, we outperform methods like Lodge which uses foot positions as conditions for diffusion denoising. Our method also demonstrates excellent performance in diversity. In terms of beat alignment score, we achieve 0.41 compared to EDGE, indicating that our generated motions better correspond to musical beats. We also obtain a low MM score of 4.5, suggesting better alignment with the given textual conditions. Regarding model efficiency, our average inference time is 3.64 seconds, showing a 50% improvement compared to EDGE, thanks to our adoption of the mamba2 ssm-based denoiser, which is more suitable for fast inference. Similarly, as shown in Table 3, EDMG also demonstrates competitive performance on the FineDance dataset.

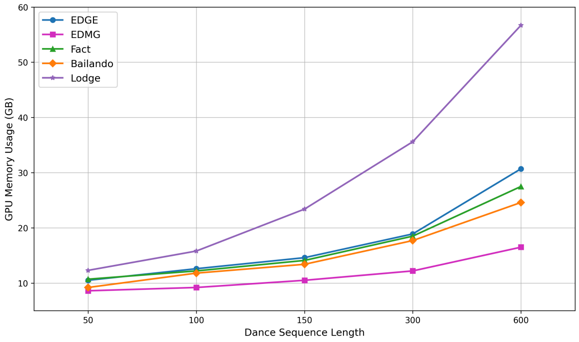


Figure 5: GPU Memory Efficiency Comparison on AIST++.

Table 4: Ablation results of PMD on AIST++.

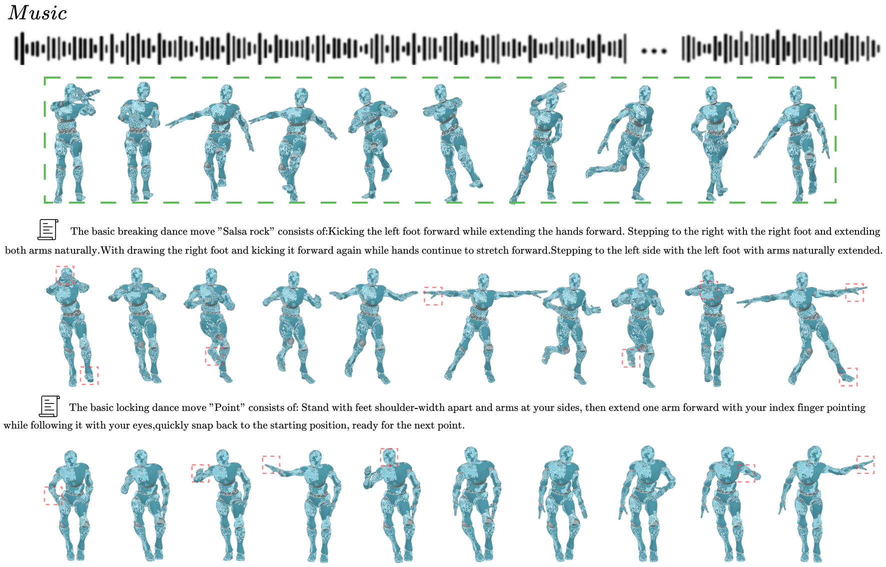
Method	FID_k ↓	FID_g ↓	PFC ↓	PBC ↓	Train Time ↓
EDMG	**10.69**	**7.35**	**0.96**	**2.34**	**29h**
w/o LIM	13.05	8.24	1.26	2.75	36h
w/o PMM	10.75	7.56	1.13	2.64	294h

A systematic evaluation is also conducted on EDMG and other algorithms in terms of computational resource utilization, as shown in Figure 5. In the short dance generation scenario (sequence length of approximately 50 frames), EDMG and other methods exhibit similar GPU memory consumption. However, as the sequence length increased, the GPU memory consumption of EDMG exhibits a linear growth characteristic, reducing resource consumption compared to other methods.

4.4 Ablation Study

Parallel Mamba Denoiser: We conduct experiments, by comparing the variant without LIM and the variant without the parallel mamba module (i.e., using the standard Mamba sequence generation method). As shown in Table 4, compared to the variant without LIM, our complete model achieves significant improvements in dance generation quality, indicating that our method successfully captures temporal correlations between dance movements. Additionally, the one-dimensional convolutional dimensionality reduction operation used in LIM effectively improves the computational efficiency of the model. In another experiment, we compare our PMD using PMM (parameter sharing Multi-Input SSM) with the standard Mamba method. The result demonstrates that our method improves the model training efficiency by an order of magnitude.

Temporal Refinement Module: EDMG reduces accumulated joint motion errors in the temporal dimension through TRM. As



The basic breaking dance move "Salsa rock" consists of:Kicking the left foot forward while extending the hands forward. Stepping to the right with the right foot and extending both arms naturally.With drawing the right foot and kicking it forward again while hands continue to stretch forward.Stepping to the left side with the left foot with arms naturally extended.

The basic locking dance move "Point" consists of: Stand with feet shoulder-width apart and arms at your sides, then extend one arm forward with your index finger pointing while following it with your eyes,quickly snap back to the starting position, ready for the next point.

Figure 6: Visual comparison of dance sequences generated by different dance genre descriptions and descriptions of fundamental movements. The green boxes show dance sequences generated solely based on music conditions. The other sequences are generated jointly conditioned on both music and dance descriptions. The red boxes highlight the movement details in the generated dance sequences that correspond to the textual descriptions.

Table 5: Ablation results of TRM on AIST++.

Method	$\text{FID}_k \downarrow$	$\text{FID}_g \downarrow$	PFC ↓	PBC ↓
Ground Truth	6.83	6.53	1.23	2.84
EDMG	**10.69**	7.35	**0.96**	**2.34**
w/o TRM	11.24	**7.34**	2.42	5.63

Table 6: User study results comparing EDMG and EDGE on AIST++ (A) and in-the-wild (W) music.

Method	GM-A ↑	GM-W ↑	RM-A ↑	RM-W ↑	Div-A ↑	Div-W ↑
EDGE	7.5	6.9	7.4	7.2	7.3	7.2
EDMG	**8.3**	**8.1**	**8.4**	**8.1**	**8.2**	**7.9**

reported in Table 5, our method significantly outperforms the variant without TRM on PFC and PBC metrics. Since these metrics relate to motion acceleration characteristics, our TRM effectively enhances dance fluency and rhythmic quality by modeling temporal dependencies.

4.5 User Study

We run a user study with 30 participants, including 10 professionals. We compare EDMG with EDGE on dance generation using AIST++ ("A") music and in-the-wild ("W") music. As shown in Table 6, participants generally prefer EDMG in terms of aesthetic preference and perceived generation quality.

4.6 Text-Driven Visualization Analysis

For the same audio input, our proposed method can generate diverse style-specific fundamental movements through text-guided conditioning. Figure 6 presents the visualization results, systematically comparing unconditionally generated fundamental movements with specific genre fundamental dance movement units generated through text prompts such as “Salsa rock” and “Point”. Experimental results demonstrate that the proposed method can well capture and generate the fundamental movements of various dance genres.

5 Conclusion

In this research, we presented a diffusion-based two-stage choreography generation model that effectively addressed the current challenge of utilizing generative models to efficiently produce realistic and extended dance sequences. Our model demonstrates state-of-the-art performance on the AIST++ and FineDance dataset. Concurrently, we substantially improve both training and inference processes, achieving a reduction in GPU memory utilization and AIT. Particularly significant is our model's personalization capability, which enables users to select dance genres and fundamental movements that better align with practical requirements for personalized generation.

Despite these advantages, our approach exhibits certain limitations. It lacks fine-grained facial expressions and finger movements, which are essential components of dance performance. Furthermore, our model fails to achieve optimal results when generating choreography for in-the-wild music with complex, rapidly changing rhythms. Future work will address these issues, with the objective of developing more immersive and comprehensive dance generation systems that can capture the full expressivity of human movement across diverse musical contexts.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 62332010, 62476131, 62377004, 62076131), the Jiangsu Provincial Science and Technology Major Project (No. BG2024042), and the Fundamental Research Funds for the Central Universities (No. 2233100028).

References

[1] Yoshua Bengio, Aaron Courville, and Pascal Vincent. 2013. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence* 35, 8 (2013), 1798–1828.

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[3] Tri Dao and Albert Gu. 2024. Transformers are ssms: Generalized models and efficient algorithms through structured state space duality. *arXiv preprint arXiv:2405.21060* (2024).

[4] Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. 2020. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341* (2020).

[5] Prafulla Dhariwal and Alexander Nichol. 2021. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems* 34 (2021), 8780–8794.

[6] Kehong Gong, Dongze Lian, Heng Chang, Chuan Guo, Zihang Jiang, Xinxin Zuo, Michael Bi Mi, and Xinchao Wang. 2023. Tm2d: Bimodality driven 3d dance generation via music-text integration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9942–9952.

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.

[8] Albert Gu and Tri Dao. 2023. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752* (2023).

[9] Albert Gu, Karan Goel, and Christopher Ré. 2021. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396* (2021).

[10] Albert Gu, Isys Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Ré. 2021. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *Advances in neural information processing systems* 34 (2021), 572–585.

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

[12] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[13] Zikai Huang, Xuemiao Xu, Cheng Xu, Huaidong Zhang, Chenxi Zheng, Jing Qin, and Shengfeng He. 2024. Beat-It: Beat-Synchronized Multi-Condition 3D Dance Generation. In *European Conference on Computer Vision*. Springer, 273–290.

[14] Rudolph Emil Kalman. 1960. A new approach to linear filtering and prediction problems. (1960).

[15] Tero Karras, Samuli Laine, and Timo Aila. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4401–4410.

[16] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. 2020. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 8110–8119.

[17] Jinwoo Kim, Heeseok Oh, Seongjean Kim, Hoseok Tong, and Sanghoon Lee. 2022. 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*. 3490–3500.

[18] Jinwoo Kim, Heeseok Oh, Seongjean Kim, Hoseok Tong, and Sanghoon Lee. 2022. 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*. 3490–3500.

[19] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[20] Durk P Kingma and Prafulla Dhariwal. 2018. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems* 31 (2018).

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

[22] Ruilong Li, Shan Yang, David A Ross, and Angjoo Kanazawa. 2021. Ai choreographer: Music conditioned 3d dance generation with aist++. In *Proceedings of the IEEE/CVF international conference on computer vision*. 13401–13412.

[23] Ronghui Li, YuXiang Zhang, Yachao Zhang, Hongwen Zhang, Jie Guo, Yan Zhang, Yebin Liu, and Xiu Li. 2024. 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*. 1524–1534.

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

[25] Shufan Li, Harkanwar Singh, and Aditya Grover. 2024. Mamba-nd: Selective state space modeling for multi-dimensional data. In *European Conference on Computer Vision*. Springer, 75–92.

[26] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan Liu. 2024. Vmamba: Visual state space model. *Advances in neural information processing systems* 37 (2024), 103031–103063.

[27] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. 2023. SMPL: A skinned multi-person linear model. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*. 851–866.

[28] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. 2023. SMPL: A skinned multi-person linear model. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*. 851–866.

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

[30] Andrew L Maas, Awni Y Hannun, Andrew Y Ng, et al. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, Vol. 30. Atlanta, GA, 3.

[31] Adriano Manfrè, Ignazio Infantino, Filippo Vella, and Salvatore Gaglio. 2016. An automatic system for humanoid dance creation. *Biologically Inspired Cognitive Architectures* 15 (2016), 1–9.

[32] Ferda Ofli, Engin Erzin, Yücel Yemez, and A Murat Tekalp. 2011. Learn2dance: Learning statistical music-to-dance mappings for choreography synthesis. *IEEE Transactions on Multimedia* 14, 3 (2011), 747–759.

[33] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. PmLR, 8748–8763.

[34] Alec Radford, Luke Metz, and Soumith Chintala. 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434* (2015).

[35] Mingyi Shi, Kfir Aberman, Andreas Aristidou, Taku Komura, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. 2020. Motionet: 3d human motion reconstruction from monocular video with skeleton consistency. *Acm transactions on graphics (tog)* 40, 1 (2020), 1–15.

[36] Xu Shi, Wei Yao, Chuanchen Luo, Junran Peng, Hongwen Zhang, and Yunlian Sun. 2024. FG-MDM: Towards Zero-Shot Human Motion Generation via ChatGPT-Refined Descriptions. In *International Conference on Pattern Recognition*. Springer, 446–461.

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

[38] Xiangjun Tang, He Wang, Bo Hu, Xu Gong, Ruifan Yi, Qilong Kou, and Xiaogang Jin. 2022. Real-time controllable motion transition for characters. *ACM Transactions on Graphics (TOG)* 41, 4 (2022), 1–10.

[39] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. 2022. Human motion diffusion model. *arXiv preprint arXiv:2209.14916* (2022).

[40] Yating Tian, Hongwen Zhang, Yebin Liu, and Limin Wang. 2023. Recovering 3d human mesh from monocular images: A survey. *IEEE transactions on pattern analysis and machine intelligence* 45, 12 (2023), 15406–15425.

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

[42] Aaron Van Den Oord, Oriol Vinyals, et al. 2017. Neural discrete representation learning. *Advances in neural information processing systems* 30 (2017).

[43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[44] Hongsong Wang, Yin Zhu, and Xin Geng. 2024. Flexible Music-Conditioned Dance Generation with Style Description Prompts. *arXiv preprint arXiv:2406.07871* (2024).

[45] Jing Nathan Yan, Jiatao Gu, and Alexander M Rush. 2024. Diffusion models without attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8239–8249.

[46] Wei Yao, Hongwen Zhang, Yunlian Sun, and Jinhui Tang. 2024. STAF: 3D human mesh recovery from video with spatio-temporal alignment fusion. *IEEE Transactions on Circuits and Systems for Video Technology* (2024).

[47] Ailing Zeng, Lei Yang, Xuan Ju, Jiefeng Li, Jianyi Wang, and Qiang Xu. 2022. Smoothnet: A plug-and-play network for refining human poses in videos. In *European Conference on Computer Vision*. Springer, 625–642.

[48] Hongwen Zhang, Yating Tian, Yuxiang Zhang, Mengcheng Li, Liang An, Zhenan Sun, and Yebin Liu. 2023. Pymaf-x: Towards well-aligned full-body model regression from monocular images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 10 (2023), 12287–12303.

[49] Hongwen Zhang, Yating Tian, Xinchi Zhou, Wanli Ouyang, Yebin Liu, Limin Wang, and Zhenan Sun. 2021. Pymaf: 3d human pose and shape regression with pyramidal mesh alignment feedback loop. In *Proceedings of the IEEE/CVF international conference on computer vision*. 11446–11456.

[50] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. 2019. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 5745–5753.

[51] Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. 2024. Vision mamba: Efficient visual representation learning with bidirectional state space model. *arXiv preprint arXiv:2401.09417* (2024).

[52] Haolin Zhuang, Shun Lei, Long Xiao, Weiqin Li, Liyang Chen, Sicheng Yang, Zhiyong Wu, Shiyin Kang, and Helen Meng. 2023. Gtn-bailando: Genre consistent long-term 3d dance generation based on pre-trained genre token network. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1–5.