Towards Storytelling Animations: Joint Synthesis of Human and Camera Motions

Boyuan Cheng[1] Yingjie Xi[1] Rui He[2] Jinhe Na[3] Ying Cao[4*]
Pengjie Wang[3] Jian J. Zhang[1] Xiaosong Yang[1]
[1]Bournemouth University
[2]The Hong Kong Polytechnic University
[3]Dalian Minzu University
[4]ShanghaiTech University

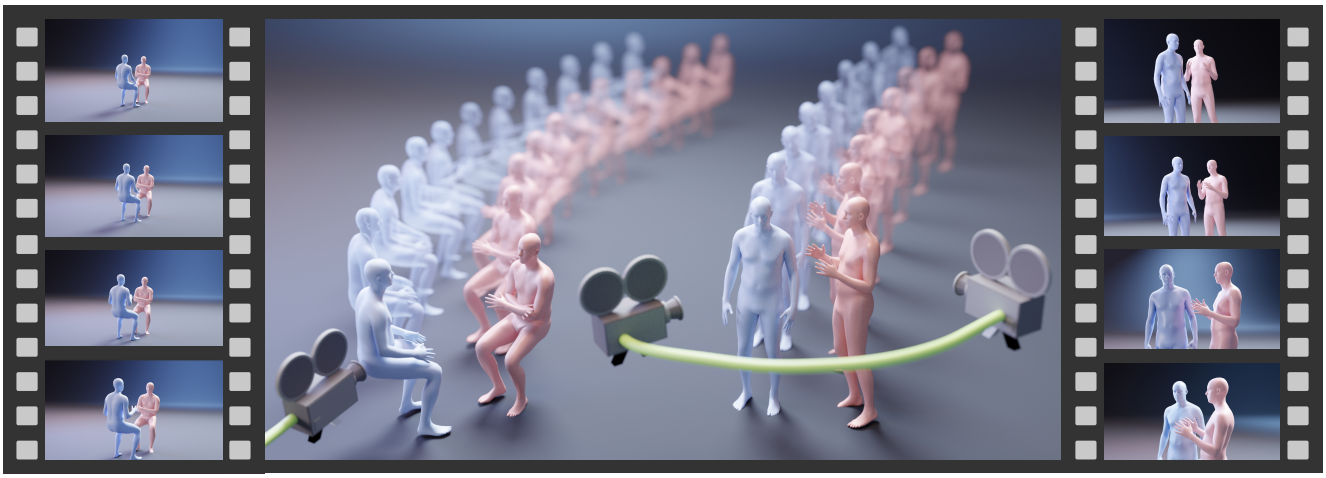

Figure 1. Our framework jointly generates the motion of two interacting characters and the motion of a camera to visually tell a story. The central panel shows two instances of generated character-camera motions. For each pair of character and camera motions, we display rendered viewpoints at four sampled moments along each camera motion trajectory.

Abstract

To tell a story effectively, a 3D animation often necessitates carefully planned behaviors of both characters and the camera in the 3D scene, where the camera placement and movement determine how the characters are displayed on screen. Thus, creating storytelling animations can be challenging. While significant progress has been made in the fields of character motion synthesis and virtual cinematography, previous methods focus on either character motion generation or camera motion generation, falling short of handling the two tasks simultaneously. In this paper, we propose a novel diffusion-based generative model to jointly synthesize character and camera motions in 3D space for creating storytelling 3D animations. Our model treats individual characters and the camera in a 3D scene as independent, equally important entities, and explicitly models pairwise interactions among them in the generation process. By being trained on a mixture dataset of real and synthetic character-camera motion data, our model is capable of generating high-quality multi-character motions coupled with compelling camera motions. We show that our model outperforms existing specialized approaches on the human motion generation and camera motion tasks.

1. Introduction

Creating high-quality character-centric 3D animations that visually tell intended stories is a critical task in a spectrum of fields such as film and video game. To produce a storytelling-effective animation that can engage audiences

*Corresponding author.

and convey message efficiently, the animators need to create the high-fidelity *motion of characters* that well align with the story (e.g., specified by a given script), and meticulously plan the *motion of a camera* (i.e., how the camera is placed and moved) through the 3D scene. The camera motion planning, known as cinematography in filmmaking, determines the size and composition of characters in the frames and can largely influence audiences' attention and feelings—e.g., a camera movement of "push in" can elevate the tension in the scene. The process of creating such storytelling 3D animation, however, is rather challenging, as it requires consideration of realistic individual character actions, complex inter-character interaction, and varying spatial relationship between characters and the camera (i.e., *the coordination of character and camera motions*), all of which are important for successful storytelling.

In recent years, significant advancements have been made in the domain of character motion synthesis, which enables automatic or interactive generation of high-fidelity and diverse character motions. Recent studies have extensively explored single-character motion generation [33, 35, 43], text-driven character motion generation [2, 8, 13, 17, 18, 21, 34, 40, 44, 49, 51], and two-character motion generation [27, 29, 30, 50]. Meanwhile, there is a large body of works on camera motion planning in virtual and real-world environments [12, 20, 22–24, 31, 36, 37, 45–48]. Several recent methods also attempt to generate camera motions from pre-existing 3D character motions [9, 12, 26]. Unfortunately, all of these method are targeted for either character motion generation or camera motion generation, without abilities to solve *both* generation problems in a *single* framework.

Motivated by the close correlation between character and camera motions in storytelling animations, we study the *joint* generation of the two kinds of motions in this paper, which has not yet been explored previously. We propose a novel diffusion-based generative model for modeling a joint distribution of character and camera motions, from which we can sample plausible two-character motions along with their aligned camera motions (as shown in Figure 1). Our diffusion model adopts a novel architectural design, whose core idea is to independently model the characters and the camera, and explicitly learn character-character and camera-character interactions. To train our model, we carefully curate a large dataset of pairs of character and camera motions. Our dataset covers expressive and professional motion data extracted from existing film clips, and high-quality motion data synthesized from a cinematography simulator. Experiments on our dataset show that our model is able to generate plausible, harmonious character and camera motions in a unified framework, exhibiting improved performance on both character and camera motion generation tasks, compared to prior methods that are tailored for one of the tasks. In this paper, we make the following contributions:

- We make the first attempt to study joint camera-character motion generation that, to the best of our knowledge, has not been investigated before.
- We present a unified framework that can simultaneously generate the 3D motions of two characters and a camera, by learning interactions between the characters as well as interactions between the characters and the camera.
- We construct a large-scale character-camera motion dataset, encompassing expressive, diverse and high-quality character and camera motions.

2. Related Work

2.1. Human Motion Generation

Motion generation can be categorized according to different input conditions. Among these, text-based motion generation [2, 4, 5, 7, 8, 13, 17, 21, 33, 34, 40, 41, 43, 49, 51] has emerged as one of the most compelling directions. Generating realistic motions from textual descriptions is a challenging multimodal problem that requires learning a joint embedding space for text and motion in order to achieve strong cross-modal alignment. To address the fine-grained requirements of practical applications, trajectory-based motion generation [1, 14, 18, 25, 38, 42, 52] has been proposed, in which specific motion properties, such as joint positions at designated times, are explicitly defined. Audio-conditioned motion generation [28, 55] also plays a crucial role, enabling synchronization of body movements with rhythm and sound.

Most of the existing work has concentrated on single-person motion synthesis. Multi-person motion generation, by contrast, presents additional challenges due to the need for modeling interactions. Early attempts often generated each character's motion independently using single-person models, followed by post-hoc interaction constraints or synchronization heuristics. However, treating each character in isolation leads to a loss of fine-grained interaction information. Recent approaches [27, 29, 30, 38, 54] overcome this limitation by jointly generating multi-person motions, thereby capturing more coherent and natural interactions.

2.2. Camera Motion Control

Designing dynamic camera motions [12, 31, 36, 37, 45–47] is a multifaceted challenge, positioned at the intersection of computer vision and graphics. Early approaches framed camera planning as a constraint-satisfaction problem, employing optimization techniques to satisfy predefined cinematic rules [3, 10, 11]. Driven by the proliferation of deep learning, neural network-based approaches have risen to prominence in virtual cinematography. For instance, Jiang et al. curated a comprehensive film clip

dataset encompassing both actor and camera motions, leveraging LSTM and diffusion architectures to synthesize dynamic camera trajectories from cinematic references or textual prompts [20, 22–24]. Building upon these generative priors, Cheng et al. [6] proposed a diffusion-based framework that generates camera trajectories directly from 3D human interactions. Furthermore, Wu et al. introduced a GAN-based controller to tailor camera motions to specific narrative requirements [47]. Additionally, interactive systems like Cinemassist [19] provide AI-driven suggestions to facilitate cinematic compositions.

In the gaming domain, camera automation has been investigated to improve player experience. Rucks and Katzakis proposed CameraAI to minimize occlusions in third-person tracking sequences [37]. Evin et al. further integrated established cinematographic principles into Cine-AI, a semi-automated toolset for generating engaging in-game cutscenes [15].

Dance camera auto-generation poses particular challenges, as it requires balancing shot variation, musical rhythm, and dance. Xie et al. attempted to derive camera motions directly from dance dynamics, although their method did not incorporate music and required additional keyframe inputs [48]. Addressing these constraints, DanceCamera3D [45] was recently proposed as the first framework to jointly model dance and camera trajectories using music-conditioned diffusion Transformers. Although this approach represents significant progress, it still struggles to reconcile smooth continuous shots with abrupt transitions, often relying on post-processing smoothing that can diminish the impact of cinematic cuts.

Unlike prior methods that handle character and camera motion separately, our work is the first to jointly generate 3D motions of two characters and a camera within one diffusion model, explicitly modeling bidirectional interactions among all entities to produce truly storytelling-oriented animations.

3. Method

3.1. Character Motion Representation

The motion of a single character across N frames is represented as a sequence $x^{1:N} = \{x^i\}_{i=1}^N$, where each x^i captures the pose at frame i. The pose comprises the global 3D root translation and the relative rotations of 22 SMPL joints [32], excluding the hands. Following [53], we adopt a continuous 6D rotation representation for each joint to ensure numerical stability and compact encoding. These rotations are derived from estimated 3D joint positions via inverse kinematics. To maintain structural alignment within the motion tensor, the root translation is augmented into a 6D vector by padding with three zeros. Concatenating these representations yields a 138-dimensional vector (23×6) per frame, with the complete sequence stored as a tensor of shape $N \times 138$.

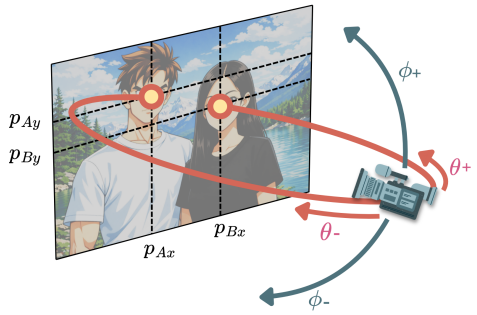

Figure 2. We represent shot composition using the on-screen coordinates of the two principal characters' heads, denoted as (p_{Ax}, p_{Ay}) and (p_{Bx}, p_{By}), along with the camera's orientation in Toric space, defined by the angles θ and ϕ.

When modeling motion involving two characters, their motion sequences are synchronized frame by frame. Since each sequence is initially defined in its own local coordinate system, where the root is positioned at the origin in the first frame, combining them directly in a shared 3D space may cause spatial collisions. To address this, an offset vector $D \in \mathbb{R}^9$ is computed for each character to specify their initial orientation and global position at the first frame. The first six dimensions of D represent the orientation, and the remaining three define the position. The character's facing direction is determined by assuming that the line connecting the shoulder joints lies on the xz-plane, and the angle between this line and the x-axis establishes the initial orientation. Although computed only once from the first frame, this offset vector is appended to every pose vector x^i. To maintain a consistent 6-dimensional representation per joint, D is zero-padded to 12 dimensions, corresponding to two additional virtual joints.

Consequently, each frame is represented by a 25×6 matrix, which is flattened into a 150-dimensional vector. The full two-character motion sequence is thus expressed as $(x_A^{1:N}, x_B^{1:N})$, where each $x^i \in \mathbb{R}^{150}$ contains both local motion features and the global placement of the character.

3.2. Camera Motion Representation

A camera motion is represented as a sequence of camera poses in 3D space. The camera pose is described using the Toric space coordinate system [31], which includes four parameters: $p_A = (p_{Ax}, p_{Ay})$, $p_B = (p_{Bx}, p_{By})$, θ, and ϕ. The variables p_A and p_B represent the normalized on-screen coordinates of two main characters' heads, forming the basis of our shot composition. The angles θ and ϕ describe the camera's yaw and pitch in 3D space relative to these reference points (see Fig. 3 for an illustration). The detailed computation of the Toric coordinates is provided in

the supplementary material.

Since Toric space is defined based on character positions, it naturally captures the spatial relationship between the camera and the characters. This makes it particularly useful for analyzing how camera motions relate to character interactions. The camera features over N frames are therefore represented in Toric space as:

$$x_c^{1:N} = \{p_A^i, p_B^i, \theta^i, \phi^i\}_{i=1}^N \in \mathbb{R}^{6N} \tag{1}$$

3.3. Joint Character-Camera Generation Model

Diffusion-based motion synthesis. MDM [41] is a diffusion-based framework designed for single-character motion generation. The model operates through two sequential stages: a forward noise-adding process and a reverse denoising procedure, both defined within the single-character motion domain (see Section 4.1). The forward process begins with a clean motion sample $x_0^{1:N} \sim q(x_0^{1:N})$ and incrementally injects Gaussian noise over T discrete steps, eventually transforming it into a sample $x_T^{1:N}$ that approximates standard Gaussian noise, i.e., $x_T^{1:N} \sim \mathcal{N}(0, I)$.

$$q(x_{1:T}^{1:N}|x_0^{1:N}) = \prod_{t=1}^{T} q(x_t^{1:N}|x_{t-1}^{1:N}) \tag{2}$$

$$q(x_t^{1:N}|x_{t-1}^{1:N}) = \mathcal{N}(x_t^{1:N}; \sqrt{1-\beta_t}x_{t-1}^{1:N}, \beta_t \mathbf{I}) \tag{3}$$

Here, β_t are fixed hyper-parameters controlling the noise variance at each step. The reverse process learns to reverse the noise through a denoising network, estimating the original motion from its noisy counterpart:

$$p(x_{0:T}^{1:N}) = p(x_T^{1:N}) \prod_{t=1}^{T} p_\theta(x_{t-1}^{1:N}|x_t^{1:N}) \tag{4}$$

$$p_\theta(x_{t-1}^{1:N}|x_t^{1:N}) = \mathcal{N}(x_{t-1}^{1:N}; \mu_\theta(x_t^{1:N}, t), \Sigma_\theta(x_t^{1:N}, t)) \tag{5}$$

The denoising network $f_\theta(x_t^{1:N}, t)$ is trained to predict the clean motion sample $x_0^{1:N}$ from its noisy version $x_t^{1:N}$ and the timestep t using the simplified loss:

$$\mathcal{L}_{simple} = \mathbb{E}_{x_0^{1:N} \sim q(x_0^{1:N}), t \sim [1,T]}[||x_0^{1:N} - f_\theta(x_t^{1:N}, t)||_2^2] \tag{6}$$

Our Model. Building on the MDM architecture, we extend the diffusion model to operate in a three-instance motion space, consisting of two interacting characters and a dynamic camera. The model takes as input a noisy motion sequence $(x_t^{A,1:N}, x_t^{B,1:N}, x_t^{c,1:N})$ and predicts the denoised sequence $(\hat{x}_0^{A,1:N}, \hat{x}_0^{B,1:N}, \hat{x}_0^{c,1:N})$.

As illustrated in Figure 3, the denoising network comprises three parallel instances of the motion generation backbone—one for each character and one for the camera. These instances are coupled through three types of interaction modules, each modeling the relationship between a pair of entities: character A and character B, character A and the camera, and character B and the camera. Each interaction module refines the intermediate features by predicting residual corrections that capture mutual influences and help recover the original spatial configuration from noisy inputs.

3.4. Interaction Modules

Each motion token sequence is first passed through its respective Transformer encoder, producing initial high-level feature embeddings: h_t^A, h_t^B, and h_t^c for character A, character B, and the camera, respectively. These embeddings serve as the input to the interaction modules.

Character-character interaction. The interaction between characters is captured by a dedicated pairwise interaction module. Taking (h_t^A, h_t^B) as input, this module predicts bidirectional residuals, $\Delta h_t^{B\to A}$ and $\Delta h_t^{A\to B}$, which encode the mutual dependencies and spatial constraints between the two subjects. These residuals are then added to the original embeddings to update their states.

Camera–character interaction. The interaction between the camera and each character is handled by two independent *camera–character interaction modules*, one for (h_t^A, h_t^c) and the other for (h_t^B, h_t^c). Each module produces two directional residuals: from the character to the camera ($\Delta h_t^{A\to c}$, $\Delta h_t^{B\to c}$) and from the camera to the character ($\Delta h_t^{c\to A}$, $\Delta h_t^{c\to B}$). The former captures how character motion and spatial configuration influence the camera's embedding, while the latter encodes how the camera's viewpoint in turn shapes the character representations. For clarity, Figure 3 depicts each type of interaction module only once, as both directions share the same architecture; in practice, the same module is applied multiple times with different input pairs.

After computing all residuals, the hidden states of each entity are updated by summing the corresponding terms:

$$\hat{h}_t^A = h_t^A + \Delta h_t^{B\to A} + \Delta h_t^{c\to A}, \tag{7}$$

$$\hat{h}_t^B = h_t^B + \Delta h_t^{A\to B} + \Delta h_t^{c\to B}, \tag{8}$$

$$\hat{h}_t^c = h_t^c + \Delta h_t^{A\to c} + \Delta h_t^{B\to c}. \tag{9}$$

The refined states $(\hat{h}_t^A, \hat{h}_t^B, \hat{h}_t^c)$ are then passed to the diffusion decoder to predict the denoised motion sequences $(\hat{x}_0^{A,1:N}, \hat{x}_0^{B,1:N}, \hat{x}_0^{c,1:N})$.

4. Our Dataset

We construct a dataset that integrates both real cinematic footage and synthetic recordings to jointly capture character

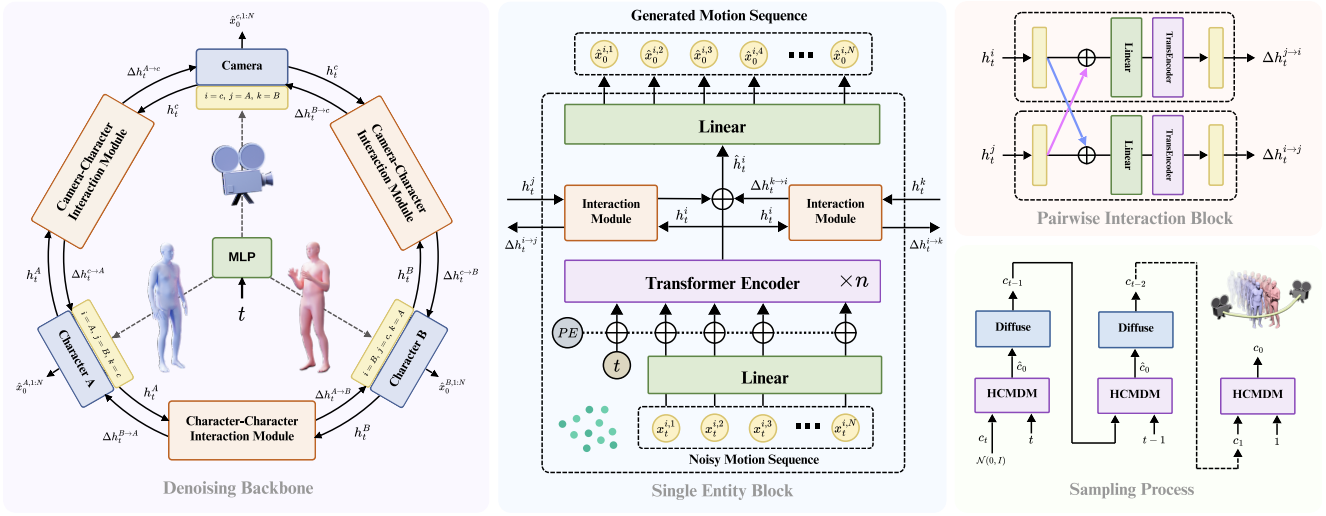

Figure 3. Overview of our joint character-camera motion generation framework. The model takes Gaussian noise as input and jointly generates motion sequences for two interacting characters and a dynamic camera. Each instance is processed through a Transformer encoder to extract high-level motion embeddings. Three pairwise interaction modules model the relationships between each pair of instances: A↔ B, A↔ Camera, and B↔ Camera. These modules produce residuals that are added to the original embeddings to enable mutual influence among all agents.

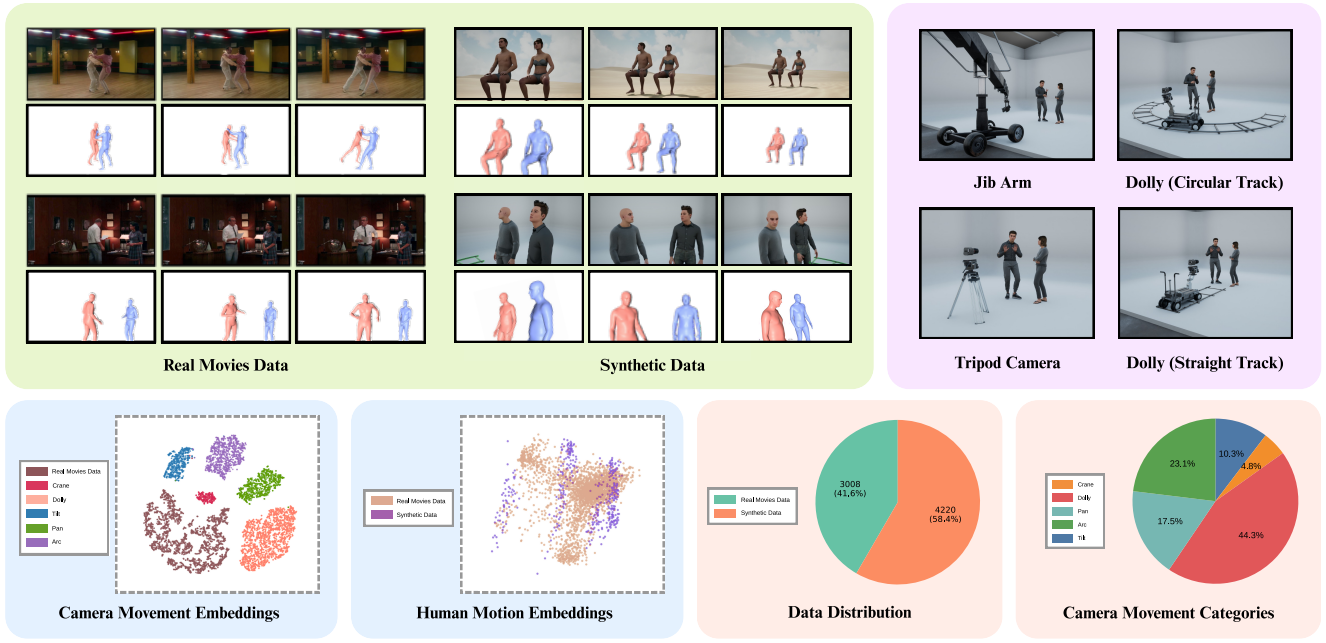

Figure 4. Overview of our dataset. The green region illustrates sample motion sequences. The pink region shows the camera rigs available in Cine Tracer for generating synthetic data. The blue region visualizes the embedding spaces of character and camera motions. The orange region presents the overall distribution of camera movement categories in the dataset.

and camera motions. The dataset consists of two sources: (i) Real Movies Data, extracted from films, television series, and stage performances, containing diverse actor interactions; and (ii) Synthetic Data, generated in the virtual production software Cine Tracer, where multiple camera rigs are employed, including a tripod camera, dolly with straight and circular tracks, and a jib arm (pink region in Figure 4).

Each sequence spans 6 seconds (120 frames at 20 fps). Motion sequences (green region in Figure 4) are extracted from video clips using MeTRAbs [**?**], which estimates 2D

Table 1. Comparison with baselines on character motion.

Method	FID ↓	Diversity ↑	InterFID ↓	Coverage ↑	Density ↑
ComMDM	$0.156^{\pm.007}$	$0.231^{\pm.020}$	$0.746^{\pm.020}$	$0.160^{\pm.004}$	$0.627^{\pm.048}$
InterGen	$0.354^{\pm.066}$	$0.577^{\pm.012}$	$0.897^{\pm.340}$	$0.018^{\pm.003}$	$0.226^{\pm.068}$
RIG	$0.495^{\pm.022}$	$\mathbf{0.609}^{\pm.009}$	$1.790^{\pm.222}$	$0.011^{\pm.004}$	$0.176^{\pm.134}$
Ours	$\mathbf{0.113}^{\pm.004}$	$0.565^{\pm.017}$	$\mathbf{0.651}^{\pm.025}$	$\mathbf{0.264}^{\pm.025}$	$\mathbf{0.990}^{\pm.047}$

Table 2. Comparison with baselines on camera motion.

Method	SeqFID ↓	FrameFID ↓	Diversity ↑	Coverage ↑	Density ↑
CDM	$0.629^{\pm.048}$	$0.638^{\pm.062}$	$0.765^{\pm.095}$	$0.072^{\pm.010}$	$0.538^{\pm.086}$
DC3D	$0.417^{\pm.027}$	$0.605^{\pm.054}$	$\mathbf{1.036}^{\pm.019}$	$0.028^{\pm.006}$	$0.227^{\pm.045}$
Ours	$\mathbf{0.256}^{\pm.024}$	$\mathbf{0.268}^{\pm.036}$	$0.771^{\pm.052}$	$\mathbf{0.081}^{\pm.010}$	$\mathbf{1.937}^{\pm.062}$

Table 3. Comparison with baselines on character-camera motion coordination.

Method	Character-Camera Alignment ↓
M2C-T	$6.136^{\pm.002}$
AutoVisNarr	$6.019^{\pm.004}$
Ours	$\mathbf{5.885}^{\pm.004}$

keypoints and applies perspective geometry optimization to recover absolute 3D root positions. Motions are represented with the SMPL-22 joint model [32]. While MeTRAbs defines the camera origin at $(0, 0, 0)$, we do not directly use these raw coordinates; the feature transformations for camera representation are detailed in Section 3.2.

The two sources are complementary: real data contains rich motion categories but predominantly static cameras, whereas synthetic data offers diverse camera movements but fewer motion variations. To illustrate this, we visualize the embedding spaces of character and camera motions (blue region in Figure 4), showing cross-domain alignment. The dataset includes 7,228 clips, with 3,008 real-world samples and 4,220 synthetic samples. The distribution of camera movement categories is shown in the orange region of Figure 4.

5. Experiments

5.1. Experimental Setup

Implementation details. All experiments are performed on a workstation featuring a 13th Gen Intel® Core™ i7-13700 processor with 24 threads. To initialize our architecture, we first pre-train the single-character motion backbone using the HumanML3D dataset [16], which comprises 17,684 motion sequences. After this, we train the entire model on our proposed character-camera motion dataset. The training is conducted for a total of 180,000 steps, with 1,000 diffusion steps per sample, a batch size of 64, and a learning rate set to 1e-3.

Datasets. We use 85% of the samples in our dataset for training and the remaining 15% for testing. As the training is conducted in an unconditional manner, there is no requirement for paired inputs and outputs. The model directly learns to sample diverse multi-agent motion sequences from the underlying distribution of the training data. The test set is used to compare the sampled generated motions against unseen real character-camera motion sequences, allowing us to evaluate how well the model captures the distributional characteristics of our dataset.

5.2. Evaluation Metrics

To comprehensively assess the performance of our unconditional multi-entity motion generation framework, we divide the evaluation into three components: (1) the quality of the two characters' motions, (2) the quality of the camera motion, and (3) the coordination between the characters and the camera.

For evaluating two-character motions, we use Fréchet Inception Distance (*FID*) for distributional similarity, *Diversity* for variability, and *InterFID* for interaction plausibility, computed from interaction-specific features that combine frame-wise orientation and root distance differences. We also report *Coverage* and *Density* to assess coverage and fidelity. All metrics use representations extracted by a self-supervised VAE-Transformer trained on our dataset.

For camera motions, we adopt *FrameFID* and *SeqFID* as complementary metrics. *FrameFID* measures the framing quality of individual frames based on the ratio of visible body parts and body projections within the camera viewport, following Wang et al. [45]. *SeqFID* quantifies the

Table 4. Ablation on interaction modeling for character motion.

Method	FID ↓	Diversity ↑	InterFID ↓	Coverage ↑	Density ↑
w/o Interaction	$0.168^{\pm.006}$	$\mathbf{0.580}^{\pm.018}$	$0.147^{\pm.007}$	$0.582^{\pm.049}$	$1.241^{\pm.133}$
w/o Separation	$0.214^{\pm.005}$	$0.250^{\pm.022}$	$0.628^{\pm.115}$	$0.530^{\pm.055}$	$0.464^{\pm.147}$
Ours	$\mathbf{0.143}^{\pm.004}$	$0.550^{\pm.015}$	$\mathbf{0.083}^{\pm.003}$	$\mathbf{0.621}^{\pm.062}$	$\mathbf{1.569}^{\pm.166}$

Table 5. Ablation on interaction modeling for camera motion.

Method	SeqFID ↓	FrameFID ↓	Diversity ↑	Coverage ↑	Density ↑
w/o Interaction	$0.081^{\pm.010}$	$0.335^{\pm.041}$	$0.370^{\pm.026}$	$0.112^{\pm.012}$	$1.241^{\pm.178}$
w/o Separation	$0.085^{\pm.006}$	$0.430^{\pm.037}$	$0.212^{\pm.023}$	$0.100^{\pm.008}$	$1.180^{\pm.231}$
Ours	$\mathbf{0.077}^{\pm.014}$	$\mathbf{0.268}^{\pm.038}$	$\mathbf{0.456}^{\pm.029}$	$\mathbf{0.121}^{\pm.011}$	$\mathbf{1.350}^{\pm.155}$

Table 6. Ablation on interaction modeling for character-camera motion coordination.

Method	Character-Camera Alignment ↓
w/o Interaction	$2.298^{\pm.039}$
w/o Separation	$2.310^{\pm.016}$
Ours	$\mathbf{2.284}^{\pm.045}$

distributional distance between generated and real camera motions. We also compute *Diversity*, *Coverage*, and *Density* on camera motions using the same VAE-Transformer encoder.

Finally, to evaluate character–camera motion coordination, we adopt a CLIP-inspired metric, *Character-Camera Alignment*. A dual-encoder model is trained on our dataset, where the character motion encoder $f_m(\cdot)$ and camera motion encoder $f_c(\cdot)$ map corresponding sequences into a shared embedding space. The model maximizes cosine similarity for matched pairs and minimizes it for mismatched ones.

5.3. Comparison with Baselines

We evaluate our method against three groups of baselines: (1) two-character motion generation models, (2) camera motion generation models, and (3) character–camera motion coordination methods. Specifically, for two-character motion, we compare against InterGen [30], ComMDM [1] and RIG [39]. For camera motion, we evaluate against CDM [24] and DanceCamera3D [45]. For motion–camera coordination, we evaluate against two baselines: AutoVisNarr [6] and M2C-T, a dual-person motion-to-camera Transformer we implemented to directly predict camera motions from paired sequences, serving as a regression-based baseline.

As shown in Table 3, our method outperforms existing baselines across all dimensions. For character motion, it achieves the best realism and interaction quality while maintaining a superior balance of diversity and plausibility. Our model also generates more cinematic and plausible camera trajectories compared to specialized camera synthesis methods. Notably, in terms of character–camera coordination, our approach yields a lower CLIP loss. While *M2C-T* and *AutoVisNarr* capture fundamental behaviors, our approach more effectively encodes the semantic correspondence between character interactions and cinematography.

Qualitative analysis. Figure 5 compares the synthesis quality of different methods. M2C-T, which relies on a deterministic regression mapping, frequently produces flawed framing; it often fails to maintain a proper field of view, resulting in excessive zooming or losing track of interacting subjects. While AutoVisNarr improves trajectory stability, its viewpoints remain somewhat static, lacking the expressive narrative flow required for storytelling. In contrast, our joint diffusion framework yields more plausible and cinematic compositions.

5.4. Ablation Study

To further validate our core design of treating characters and the camera as distinct entities with explicit pairwise interaction modules, we compare our full model against two variants. (1) No Interaction (w/o Interaction): The three entities are processed by independent Transformer encoders without interaction modules, which disables cross-entity feature exchange. (2) Single-Encoder (w/o Separation): All entities are concatenated into a single sequence and passed through a unified encoder. This configuration omits explicit bidirectional exchange and relies solely on shared self-attention to implicitly capture dependencies.

Quantitative results in Table 6 confirm the efficacy of our architecture. The Single-Encoder variant performs the

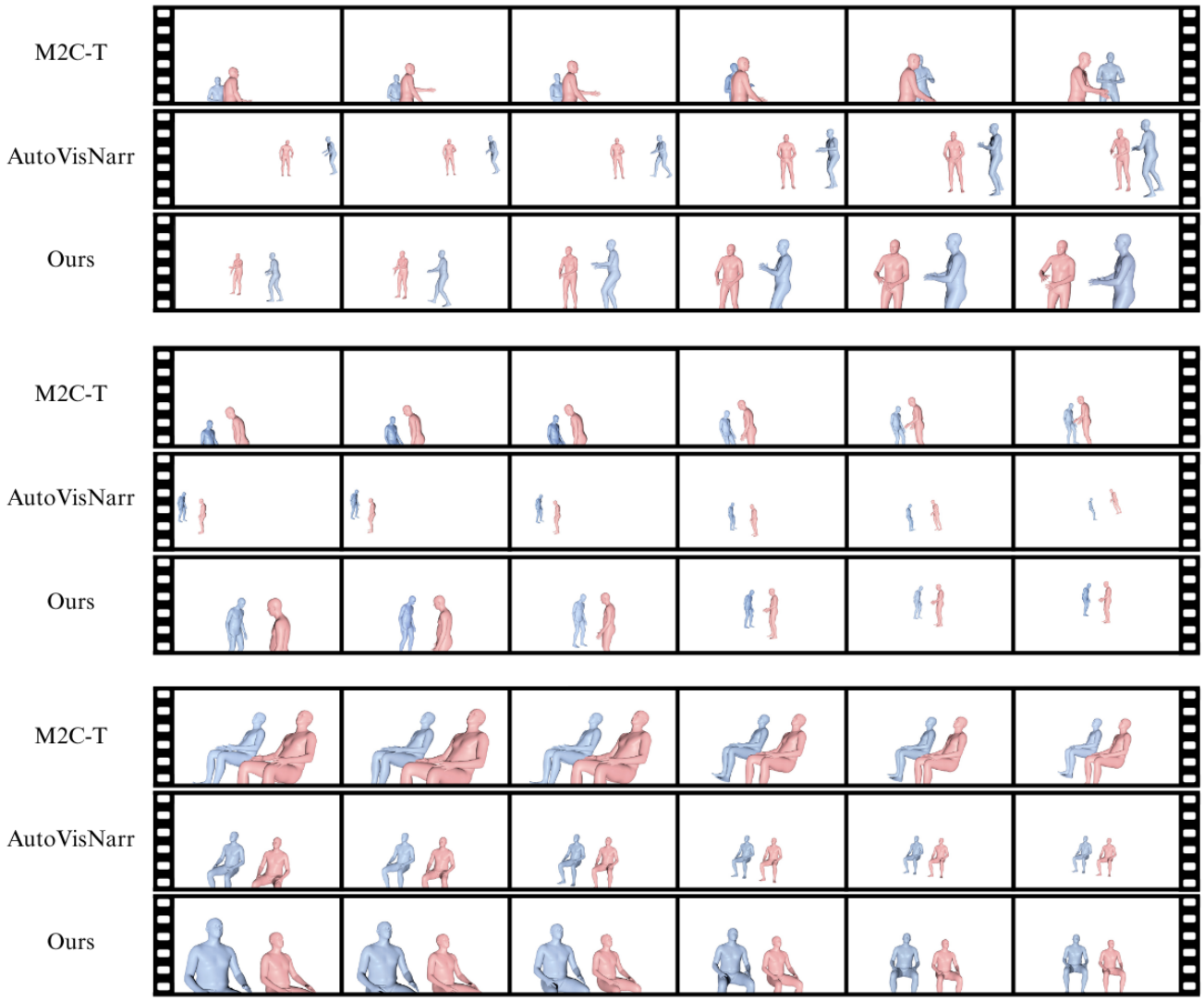

Figure 5. Qualitative comparison. While M2C-T and AutoVisNarr yield fragmented or static compositions, our method maintains expressive framing and cinematic flow, resulting in superior coordination between characters and camera motion.

worst, suggesting that a lack of entity separation degrades both motion fidelity and coordination. No Interaction improves results by preserving entity distinctness but remains suboptimal due to the absence of explicit communication. In contrast, our full model consistently achieves the best performance. This demonstrates that modeling characters and cameras as separate yet interacting entities is essential for generating expressive and cinematically synergistic storytelling animations.

6. Conclusion

We present a novel diffusion-based framework and a dedicated dataset for the joint synthesis of two-character interactions and camera movements. Our approach treats characters and the camera as equal motion entities, introducing pairwise interaction modules to capture fine-grained spatio-temporal dependencies across multiple actors. By leveraging a unified diffusion-based architecture and training on a specialized dataset of character–camera sequences, our framework generates coherent and expressive animations end-to-end, without relying on external conditioning. Experimental evaluations demonstrate that our method consistently outperforms established baselines. Moving forward, we plan to extend the framework to scenarios involving multi-character dynamics. Furthermore, we aim to explore conditional generation driven by textual narratives, broadening the scope for automated cinematic content creation and interactive media.

References

[1] Nikos Athanasiou, Mathis Petrovich, Michael J Black, and Gül Varol. Sinc: Spatial composition of 3d human motions for simultaneous action generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9984–9995, 2023. 2, 7

[2] Nikos Athanasiou, Alpár Cseke, Markos Diomataris, Michael J Black, and Gül Varol. Motionfix: Text-driven 3d human motion editing. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–11, 2024. 2

[3] William Bares, Scott McDermott, Christina Boudreaux, and Somying Thainimit. Virtual 3d camera composition from frame constraints. In *Proceedings of the eighth ACM international conference on Multimedia*, pages 177–186, 2000. 2

[4] Ling-Hao Chen, Shunlin Lu, Wenxun Dai, Zhiyang Dou, Xuan Ju, Jingbo Wang, Taku Komura, and Lei Zhang. Pay attention and move better: Harnessing attention for interactive motion generation and training-free editing. *arXiv preprint arXiv:2410.18977*, 2024. 2

[5] Rui Chen, Mingyi Shi, Shaoli Huang, Ping Tan, Taku Komura, and Xuelin Chen. Taming diffusion probabilistic models for character control. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–10, 2024. 2

[6] Boyuan Cheng, Shang Ni, Jian Jun Zhang, and Xiaosong Yang. Automating visual narratives: Learning cinematic camera perspectives from 3d human interaction. *Computers & Graphics*, page 104484, 2025. 3, 7

[7] Boyuan Cheng, Zixuan Zhou, Yajie Deng, Siyao Du, Zhengyezi Wang, Yi Wen, Yixuan Zhang, Shang Ni, Yanzhe Kong, Liuxuan Xie, et al. Context-aware human motion generation: a comprehensive survey. *Design and Artificial Intelligence*, page 100007, 2025. 2

[8] Seunggeun Chi, Hyung-gun Chi, Hengbo Ma, Nakul Agarwal, Faizan Siddiqui, Karthik Ramani, and Kwonjoon Lee. M2d2m: Multi-motion generation from text with discrete diffusion models. In *European conference on computer vision*, pages 18–36. Springer, 2024. 2

[9] Jinyoung Choi, Kangmin Kim, Seongil Kim, Minseok Kim, Taekgwan Nam, and Youngjin Park. Camera path generation for triangular mesh using toroidal patches. *Applied Sciences*, 14(2):490, 2024. 2

[10] Marc Christie and Jean-Marie Normand. A semantic space partitioning approach to virtual camera composition. In *Computer Graphics Forum*, pages 247–256. Amsterdam: North Holland, 1982-, 2005. 2

[11] Marc Christie, Patrick Olivier, and Jean-Marie Normand. Camera control in computer graphics. In *Computer graphics forum*, pages 2197–2218. Wiley Online Library, 2008. 2

[12] Robin Courant, Nicolas Dufour, Xi Wang, Marc Christie, and Vicky Kalogeiton. Et the exceptional trajectories: Text-to-camera-trajectory generation with character awareness. In *European Conference on Computer Vision*, pages 464–480. Springer, 2024. 2

[13] Rishabh Dabral, Muhammad Hamza Mughal, Vladislav Golyanik, and Christian Theobalt. Mofusion: A framework for denoising-diffusion-based motion synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9760–9770, 2023. 2

[14] Wenxun Dai, Ling-Hao Chen, Jingbo Wang, Jinpeng Liu, Bo Dai, and Yansong Tang. Motionlcm: Real-time controllable motion generation via latent consistency model. In *European Conference on Computer Vision*, pages 390–408. Springer, 2024. 2

[15] Inan Evin, Perttu Hämäläinen, and Christian Guckelsberger. Cine-ai: Generating video game cutscenes in the style of human directors. *Proceedings of the ACM on Human-Computer Interaction*, 6(CHI PLAY):1–23, 2022. 3

[16] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5152–5161, 2022. 6

[17] 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*, pages 1900–1910, 2024. 2

[18] Ziyan Guo, Zeyu Hu, De Wen Soh, and Na Zhao. Motionlab: Unified human motion generation and editing via the motion-condition-motion paradigm. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13869–13879, 2025. 2

[19] Rui He, Huaxin Wei, and Ying Cao. An interactive system for supporting creative exploration of cinematic composition designs. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, pages 1–15, 2024. 3

[20] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 2, 3

[21] Yiheng Huang, Hui Yang, Chuanchen Luo, Yuxi Wang, Shibiao Xu, Zhaoxiang Zhang, Man Zhang, and Junran Peng. Stablemofusion: Towards robust and efficient diffusion-based motion generation framework. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 224–232, 2024. 2

[22] Hongda Jiang, Bin Wang, Xi Wang, Marc Christie, and Baoquan Chen. Example-driven virtual cinematography by learning camera behaviors. *ACM Trans. Graph.*, 39(4):45, 2020. 2, 3

[23] Hongda Jiang, Marc Christie, Xi Wang, Libin Liu, Bin Wang, and Baoquan Chen. Camera keyframing with style and control. *ACM Transactions on Graphics (TOG)*, 40(6): 1–13, 2021.

[24] Hongda Jiang, Xi Wang, Marc Christie, Libin Liu, and Baoquan Chen. Cinematographic camera diffusion model. In *Computer Graphics Forum*, page e15055. Wiley Online Library, 2024. 2, 3, 7

[25] Kacper Kania, Marek Kowalski, et al. Trajevae: Controllable human motion generation from trajectories. *arXiv preprint arXiv:2104.00351*, 2021. 2

[26] Hyewon Lee, Christopher Bannon, and Andrea Bianchi. Camara: Exploring and creating camera movements with spatial reference in augmented reality. In *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, pages 1–5, 2025. 2

[27] Baiyi Li, Edmond SL Ho, Hubert PH Shum, and He Wang. Two-person interaction augmentation with skeleton priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1900–1910, 2024. 2

[28] 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. 2

[29] Ronghui Li, Youliang Zhang, Yachao Zhang, Yuxiang Zhang, Mingyang Su, Jie Guo, Ziwei Liu, Yebin Liu, and Xiu Li. Interdance: Reactive 3d dance generation with realistic duet interactions. *arXiv preprint arXiv:2412.16982*, 2024. 2

[30] Han Liang, Wenqian Zhang, Wenxuan Li, Jingyi Yu, and Lan Xu. Intergen: Diffusion-based multi-human motion generation under complex interactions. *International Journal of Computer Vision*, 132(9):3463–3483, 2024. 2, 7

[31] Christophe Lino and Marc Christie. Intuitive and efficient camera control with the toric space. *ACM Transactions on Graphics (TOG)*, 34(4):1–12, 2015. 2, 3

[32] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 851–866. 2023. 3, 6

[33] Shunlin Lu, Ling-Hao Chen, Ailing Zeng, Jing Lin, Ruimao Zhang, Lei Zhang, and Heung-Yeung Shum. Humantomato: Text-aligned whole-body motion generation. *arXiv preprint arXiv:2310.12978*, 2023. 2

[34] Sihan Ma, Qiong Cao, Jing Zhang, and Dacheng Tao. Contact-aware human motion generation from textual descriptions. *arXiv preprint arXiv:2403.15709*, 2024. 2

[35] Mathis Petrovich, Michael J Black, and Gül Varol. Temos: Generating diverse human motions from textual descriptions. In *European conference on computer vision*, pages 480–497. Springer, 2022. 2

[36] Anyi Rao, Xuekun Jiang, Yuwei Guo, Linning Xu, Lei Yang, Libiao Jin, Dahua Lin, and Bo Dai. Dynamic storyboard generation in an engine-based virtual environment for video production. In *ACM SIGGRAPH 2023 Posters*, pages 1–2. 2023. 2

[37] James Rucks and Nikolaos Katzakis. Camerai: Chase camera in a dense environment using a proximal policy optimization-trained neural network. In *2021 IEEE Conference on Games (CoG)*, pages 1–8. IEEE, 2021. 2, 3

[38] Yonatan Shafir, Guy Tevet, Roy Kapon, and Amit H Bermano. Human motion diffusion as a generative prior. *arXiv preprint arXiv:2303.01418*, 2023. 2

[39] Mikihiro Tanaka and Kent Fujiwara. Role-aware interaction generation from textual description. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 15999–16009, 2023. 7

[40] Guy Tevet, Brian Gordon, Amir Hertz, Amit H Bermano, and Daniel Cohen-Or. Motionclip: Exposing human motion generation to clip space. In *European Conference on Computer Vision*, pages 358–374. Springer, 2022. 2

[41] 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. 2, 4

[42] Weilin Wan, Zhiyang Dou, Taku Komura, Wenping Wang, Dinesh Jayaraman, and Lingjie Liu. Tlcontrol: Trajectory and language control for human motion synthesis. In *European Conference on Computer Vision*, pages 37–54. Springer, 2024. 2

[43] Congyi Wang. T2m-hifigpt: generating high quality human motion from textual descriptions with residual discrete representations. *arXiv preprint arXiv:2312.10628*, 2023. 2

[44] Sen Wang, Jiangning Zhang, Xin Tan, Zhifeng Xie, Chengjie Wang, and Lizhuang Ma. Mmofusion: Multi-modal co-speech motion generation with diffusion model. *Pattern Recognition*, 169:111774, 2026. 2

[45] Zixuan Wang, Jia Jia, Shikun Sun, Haozhe Wu, Rong Han, Zhenyu Li, Di Tang, Jiaqing Zhou, and Jiebo Luo. Dancecamera3d: 3d camera movement synthesis with music and dance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7892–7901, 2024. 2, 3, 6, 7

[46] Zixuan Wang, Jiayi Li, Xiaoyu Qin, Shikun Sun, Songtao Zhou, Jia Jia, and Jiebo Luo. Dancecamanimator: Keyframe-based controllable 3d dance camera synthesis. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 10200–10209, 2024.

[47] Xinyi Wu, Haohong Wang, and Aggelos K Katsaggelos. The secret of immersion: actor driven camera movement generation for auto-cinematography. *arXiv preprint arXiv:2303.17041*, 4, 2023. 2, 3

[48] Chun Xie, Isao Hemmi, Hidehiko Shishido, and Itaru Kitahara. Camera motion generation method based on performer's position for performance filming. In *2023 IEEE 12th Global Conference on Consumer Electronics (GCCE)*, pages 957–960. IEEE, 2023. 2, 3

[49] Yiming Xie, Varun Jampani, Lei Zhong, Deqing Sun, and Huaizu Jiang. Omnicontrol: Control any joint at any time for human motion generation. *arXiv preprint arXiv:2310.08580*, 2023. 2

[50] Liang Xu, Xintao Lv, Yichao Yan, Xin Jin, Shuwen Wu, Congsheng Xu, Yifan Liu, Yizhou Zhou, Fengyun Rao, Xingdong Sheng, et al. Inter-x: Towards versatile human-human interaction analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22260–22271, 2024. 2

[51] Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Yong Zhang, Hongwei Zhao, Hongtao Lu, Xi Shen, and Ying Shan. Generating human motion from textual descriptions with discrete representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14730–14740, 2023. 2

[52] Mingyuan Zhang, Huirong Li, Zhongang Cai, Jiawei Ren, Lei Yang, and Ziwei Liu. Finemogen: Fine-grained spatio-

temporal motion generation and editing. *Advances in Neural Information Processing Systems*, 36:13981–13992, 2023. 2

[53] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5745–5753, 2019. 3

[54] Zixiang Zhou, Yu Wan, and Baoyuan Wang. Avatargpt: All-in-one framework for motion understanding planning generation and beyond. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1357–1366, 2024. 2

[55] Wentao Zhu, Xiaoxuan Ma, Dongwoo Ro, Hai Ci, Jinlu Zhang, Jiaxin Shi, Feng Gao, Qi Tian, and Yizhou Wang. Human motion generation: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(4):2430–2449, 2023. 2