# From Script to Shot: Joint Generation of Camera Pose and Dual-Human 3D Actions

ANONYMOUS AUTHOR(S)
SUBMISSION ID: 1234

CCS Concepts: • **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

Additional Key Words and Phrases: Wireless sensor networks, media access control, multi-channel, radio interference, time synchronization

**ACM Reference Format:**
Anonymous Author(s). 2026. From Script to Shot: Joint Generation of Camera Pose and Dual-Human 3D Actions. *ACM Trans. Graph.* 1, 1 (January 2026), 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

## 1 Introduction

## 2 Related Work

### 2.1 Human Motion Generation

Human motion generation has been explored under diverse conditioning signals. Among them, text-driven motion generation [Athanasiou et al. 2024; Chen et al. 2024a,b; Chi et al. 2024; Dabral et al. 2023; Guo et al. 2024; Huang et al. 2024; Lu et al. 2023; Ma et al. 2024; Tevet et al. 2022a,b; Wang 2023; Xie et al. 2024; Zhang et al. 2023b] has attracted significant attention due to its flexibility and accessibility. Mapping natural language descriptions to realistic human motions remains a challenging multimodal problem that requires robust cross-modal alignment between textual intent and motion dynamics. To support fine-grained controllability in practical applications, trajectory-conditioned methods [Athanasiou et al. 2023; Dai et al. 2024; Guo et al. 2025; Kania et al. 2021; Shafir et al. 2023; Wan et al. 2024; Zhang et al. 2023a] explicitly constrain spatial properties such as joint positions or root trajectories. Audio-conditioned motion generation [Li et al. 2024b; Zhu et al. 2023] has also been studied to synchronize body movements with rhythm and sound.

Most existing work focuses on single-person motion synthesis. Multi-person motion generation introduces additional challenges, as it must capture human–human interactions and coordination. Early approaches often generated each character independently using single-person models, followed by post-hoc constraints or heuristic synchronization, which may miss subtle interaction cues. Recent approaches [Li et al. 2024a,c; Liang et al. 2024; Shafir et al. 2023; Zhou et al. 2024] overcome this limitation by jointly generating multi-person motions, capturing more coherent and natural interactions.


ACM 1557-7368/2026/1-ART
https://doi.org/10.1145/nnnnnnn.nnnnnnn

### 2.2 Virtual Cinematography and Camera Control

Designing cinematic camera behaviors [Courant et al. 2024; Lino and Christie 2015; Rao et al. 2023; Rucks and Katzakis 2021; Wang et al. 2024a,b; Wu et al. 2023] has long been studied in computer graphics and computer vision, ranging from constraint-based planning to learning-based camera synthesis. Early work formulated camera control as a constraint satisfaction or optimization problem, producing desired framing and camera behaviors under cinematographic constraints [Bares et al. 2000; Christie and Normand 2005; Christie et al. 2008].

With the rise of deep learning, data-driven approaches have become increasingly prevalent. Jiang et al. constructed film clip datasets with paired actor and camera motions, and explored recurrent and diffusion-based models to synthesize camera movements from film references or textual descriptions [Ho et al. 2020; Jiang et al. 2021, 2020, 2024]. Wu et al. further proposed a GAN-based controller to generate camera motions aligned with narrative requirements [Wu et al. 2023]. In addition, Cinemassist provides interactive AI-driven suggestions to assist users in designing creative and coherent cinematic compositions in 3D scenes [He et al. 2024]. Cheng et al. learns to infer cinematic camera viewpoints from two-person interaction motion, demonstrating how character dynamics can guide framing decisions [Cheng et al. 2025]. Beyond direct prediction, language has also been investigated as an intuitive interface for camera manipulation. For example, ChatCam enables conversational camera control, demonstrating the potential of natural language to specify high-level cinematographic intent [Xiao et al. 2024].

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 [Rucks and Katzakis 2021]. Evin et al. further integrated established cinematographic principles into Cine-AI, a semi-automated toolset for generating engaging in-game cutscenes [Evin et al. 2022].

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 [Xie et al. 2023]. To address these limitations, Wang et al. introduced DanceCamera3D, the first dataset combining dance, camera, and music, together with a transformer-based diffusion model [Wang et al. 2024a]. 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.

### 2.3 Shot-Level Cinematic Semantics

Beyond camera trajectories, another line of work studies *shots* as semantic units governed by cinematographic conventions such as shot scale, camera angle, and camera level. These shot attributes are fundamental to cinematic language, shaping how character actions are perceived and how narrative emphasis is conveyed through framing. To support computational understanding of such conventions, several datasets provide structured shot-level annotations. MovieNet offers a holistic benchmark for movie understanding, including shot-level metadata and cinematic attributes useful for modeling camera behaviors in natural films [Huang et al. 2020]. CineScale introduces shot scale annotations (e.g., close-up, medium shot, long shot), enabling large-scale analysis of framing patterns in movies [Savardi et al. 2021]. CineScale2 extends this taxonomy with richer camera attributes such as camera angle and camera level, providing a more detailed representation of shot composition [Savardi et al. 2023]. Similarly, MovieShots and its associated model SGNet study shot type recognition by modeling subject-centric lens patterns and camera behaviors from film footage [Rao et al. 2020].

Despite significant progress, existing approaches typically focus on either generating camera motion, or analyzing shot attributes from existing videos, while human motion generation methods commonly treat character motion independently from camera framing. As a result, they do not capture the interdependencies between camera placement and multi-character actions. In contrast, our work bridges these directions by introducing a unified diffusion framework that jointly generates camera pose and two-character 3D actions from text, producing shot-level cinematic configurations.

## 3 Method

### 3.1 Character Pose Representation

We represent human pose using SMPL parameters. For a single character, we define (i) the root translation in $\mathbb{R}^3$ and (ii) the articulated rotations of 22 SMPL body joints [Loper et al. 2015] (excluding two hand joints). Joint rotations are expressed using the continuous 6D representation [Zhou et al. 2019], which is stable for regression and avoids angle wrap-around issues. We obtain these 6D rotations through inverse kinematics based on reconstructed 3D joint locations. To build a uniform tensor representation, we embed the root translation into the same 6D vector by concatenating three zeros, and stack it with the 22 joint rotations. This yields a pose matrix of size $23 \times 6$, which we vectorize into $x \in \mathbb{R}^{138}$.

For two-character pose modeling, we represent each character with the same pose encoding and combine them at the pose level. Since each character is initially defined in its own local coordinate system (with the root centered at the origin), directly combining two poses may lead to ambiguous relative placement and spatial collisions. We address this by attaching a placement vector $D \in \mathbb{R}^9$ to each character, which anchors the pose in a shared global space. The vector contains a global facing orientation in 6D and a global root translation in $\mathbb{R}^3$. The facing direction is estimated from the shoulder axis projected onto the ground plane ($xz$-plane), which provides a consistent reference for left–right body orientation. For compatibility with the per-joint 6D layout, we extend $D$ to 12 dimensions via zero padding, considered as two auxiliary joints.

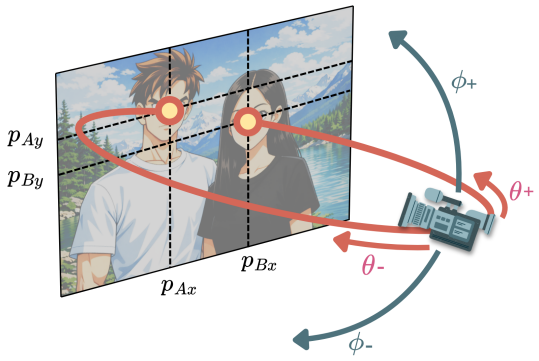


Fig. 1. Shot composition is parameterized by the normalized screen positions of the two characters' heads, $(p_{Ax}, p_{Ay})$ and $(p_{Bx}, p_{By})$, together with the camera orientation in Toric space, specified by $(\theta, \phi)$.

After augmentation, each character is represented as a $25 \times 6$ matrix and flattened into a 150-dimensional vector. We denote the two-character pose representation as $(x_A, x_B)$ with $x_A, x_B \in \mathbb{R}^{150}$, which preserves both articulated pose details and the global placement necessary for modeling interactions.

### 3.2 Camera Representation

We encode each camera state using the Toric parameterization [Lino and Christie 2015], which expresses viewpoint and framing relative to two reference subjects. Given the two principal characters, we extract their head locations on the image plane and represent them as normalized on-screen coordinates $p_A = (p_{Ax}, p_{Ay})$ and $p_B = (p_{Bx}, p_{By})$. These two points capture the shot composition in terms of where the subjects appear within the frame. In addition, the camera orientation is described by two Toric angles $(\theta, \phi)$, corresponding to the azimuth (yaw) and elevation (pitch) of the camera in 3D space with respect to the subject-centered Toric frame (Fig. 1). A detailed derivation of the Toric coordinates and the conversion to 3D camera parameters are provided in the supplementary material.

This representation couples camera parameters with subject layout by construction, making it suitable for modeling shot framing decisions in two-character scenarios. In our setting, each sample corresponds to a single shot configuration and is therefore represented as

$$x_C = \{p_A, p_B, \theta, \phi\} \in \mathbb{R}^6. \tag{1}$$

### 3.3 Joint Character–Camera Generation Model

**Preliminaries.** We build upon the denoising diffusion formulation used in recent human pose generators such as MDM [Tevet et al. 2022b]. Unlike sequence-based settings, our goal is to generate a *single-shot configuration* consisting of the camera state and two character poses. We denote the full state as

$$y = (x_A, x_B, x_C), \tag{2}$$

where $x_A, x_B \in \mathbb{R}^{150}$ are the two-character pose vectors (Sec. 3.1) and $x_C \in \mathbb{R}^6$ is the camera representation in Toric space (Sec. 3.2). Given a text prompt $s$, our goal is to sample $y_0$ from the conditional distribution $p(y_0 \mid s)$.

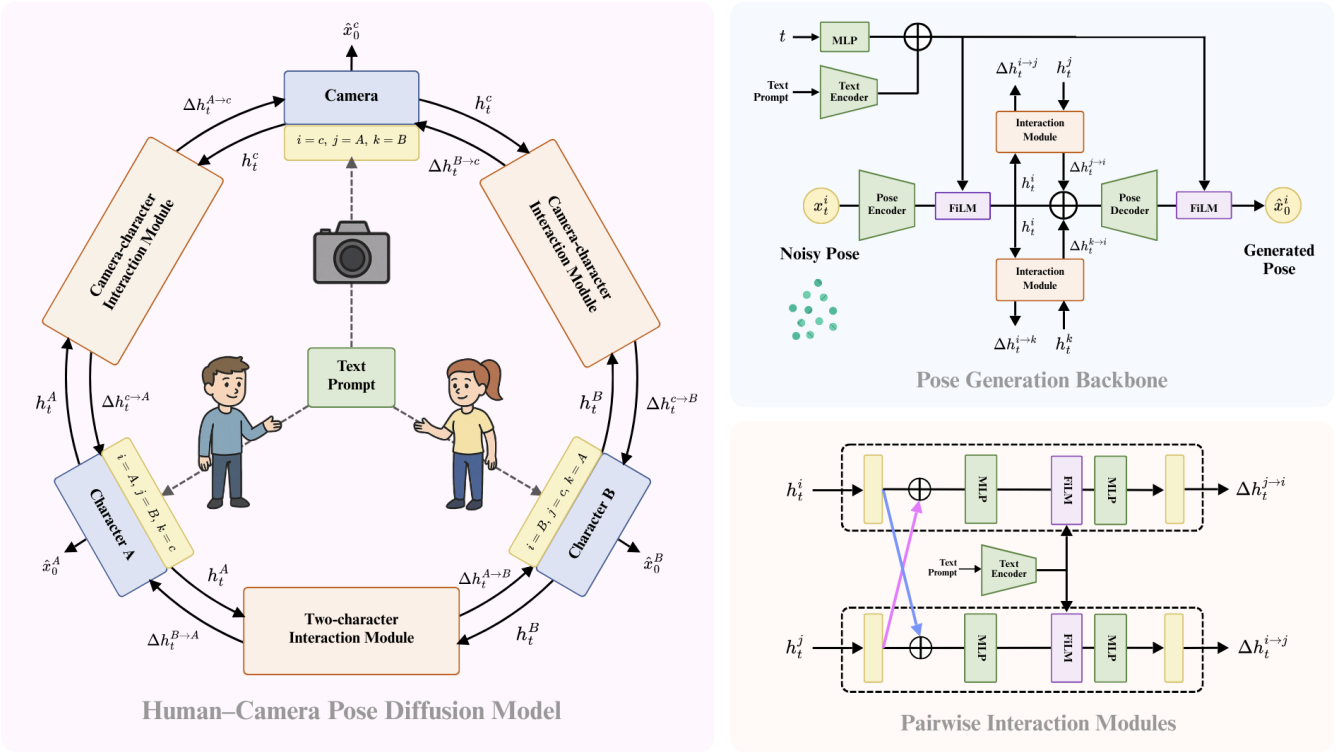


Fig. 2. Overview of our text-conditioned joint generation framework. Starting from Gaussian noise, the model jointly denoises a two-character pose pair and a camera state. Three parallel Transformer backbones encode intermediate embeddings for character A, character B, and the camera, while three pairwise interaction modules exchange residual messages between entity pairs (A↔B, A↔c, B↔c) to capture mutual dependencies under textual guidance.

**Forward process.** Starting from a clean sample $y_0 \sim q(y_0)$, the forward diffusion gradually adds Gaussian noise over $T$ steps:

$$q(y_{1:T} \mid y_0) = \prod_{t=1}^{T} q(y_t \mid y_{t-1}), \tag{3}$$

$$q(y_t \mid y_{t-1}) = \mathcal{N}\left(y_t; \sqrt{1-\beta_t}\, y_{t-1}, \beta_t \mathbf{I}\right), \tag{4}$$

where $\{\beta_t\}$ is a fixed variance schedule.

**Reverse process.** The reverse denoising process is parameterized by a neural network conditioned on the text prompt:

$$p_\theta(y_{0:T} \mid s) = p(y_T) \prod_{t=1}^{T} p_\theta(y_{t-1} \mid y_t, s). \tag{5}$$

We train a denoiser $f_\theta(y_t, t, s)$ to predict the clean state $y_0$ from $(y_t, t)$ under text conditioning using the objective

$$\mathcal{L} = \mathbb{E}_{y_0,t}\left[\|y_0 - f_\theta(y_t, t, s)\|_2^2\right]. \tag{6}$$

**Network architecture.** As shown in Fig. 2, our denoiser consists of three parallel branches that process the noisy states $(x_t^A, x_t^B, x_t^C)$. Each branch encodes its input into an intermediate embedding $(h_t^A, h_t^B, h_t^c)$, which is subsequently refined through pairwise interaction modules and decoded to predict $(\hat{x}_0^A, \hat{x}_0^B, \hat{x}_0^C)$. FiLM is used to inject both text and timestep conditions throughout the denoising network.

## 3.4 Interaction Modules

The interaction modules implement directed residual update passing between entity pairs to reflect real production scenarios, in which the poses of two characters and the camera framing are decided jointly. The relative positions and orientations of the characters constrain feasible camera placements to ensure visibility and clear composition. Conversely, the selected camera viewpoint influences how characters are oriented, spaced, and posed within the frame.

**Module design.** Each interaction module consists of two MLP layers with a FiLM modulation in between. Given a source–target embedding pair, the module produces a directed residual update that refines the target representation via residual addition. The FiLM parameters are generated from the text prompt $s$, enabling text-dependent interaction behavior.

**Character–character interaction.** A character–character module $\mathcal{I}_{HH}$ captures coordination and spatial relations between two people by predicting bidirectional residuals:

$$\Delta h_t^{A\to B} = \mathcal{I}_{HH}(h_t^A, h_t^B; s), \quad \Delta h_t^{B\to A} = \mathcal{I}_{HH}(h_t^B, h_t^A; s). \tag{7}$$

**Camera–character interaction.** To couple camera framing with character configurations, we apply a camera–character module $\mathcal{I}_{CH}$ to each character–camera pair and compute residuals in both directions:

$$\Delta h_t^{A\to c} = \mathcal{I}_{CH}(h_t^A, h^c; s), \quad \Delta h_t^{c\to A} = \mathcal{I}_{CH}(h_t^c, h^A; s), \tag{8}$$

$$\Delta h_t^{B\to c} = \mathcal{I}_{CH}(h_t^B, h_t^c; s), \quad \Delta h_t^{c\to B} = \mathcal{I}_{CH}(h_t^c, h_t^B; s). \tag{9}$$

**Embedding refinement.** All residuals are aggregated to update each entity embedding:

$$\hat{h}_t^A = h_t^A + \Delta h_t^{B\to A} + \Delta h_t^{c\to A}, \tag{10}$$

$$\hat{h}_t^B = h_t^B + \Delta h_t^{A\to B} + \Delta h_t^{c\to B}, \tag{11}$$

$$\hat{h}_t^c = h_t^c + \Delta h_t^{A\to c} + \Delta h_t^{B\to c}. \tag{12}$$

The refined embeddings $(\hat{h}_t^A, \hat{h}_t^B, \hat{h}_t^c)$ are passed to the decoders to produce the denoised predictions $(\hat{x}_0^A, \hat{x}_0^B, \hat{x}_0^C)$.

## References


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

Nikos Athanasiou, Mathis Petrovich, Michael J Black, and Gül Varol. 2023. SINC: Spatial composition of 3D human motions for simultaneous action generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9984–9995.

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

Ling-Hao Chen, Shunlin Lu, Wenxun Dai, Zhiyang Dou, Xuan Ju, Jingbo Wang, Taku Komura, and Lei Zhang. 2024a. Pay Attention and Move Better: Harnessing Attention for Interactive Motion Generation and Training-free Editing. *arXiv preprint arXiv:2410.18977* (2024).

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

Boyuan Cheng, Shang Ni, Jian Jun Zhang, and Xiaosong Yang. 2025. Automating visual narratives: Learning cinematic camera perspectives from 3D human interaction. *Computers & Graphics* (2025), 104484.

Seunggeun Chi, Hyung-gun Chi, Hengbo Ma, Nakul Agarwal, Faizan Siddiqui, Karthik Ramani, and Kwonjoon Lee. 2024. M2d2m: Multi-motion generation from text with discrete diffusion models. In *European Conference on Computer Vision*. Springer, 18–36.

Marc Christie and Jean-Marie Normand. 2005. A semantic space partitioning approach to virtual camera composition. In *Computer Graphics Forum*, Vol. 24. Amsterdam: North Holland, 1982-, 247–256.

Marc Christie, Patrick Olivier, and Jean-Marie Normand. 2008. Camera control in computer graphics. In *Computer graphics forum*, Vol. 27. Wiley Online Library, 2197–2218.

Robin Courant, Nicolas Dufour, Xi Wang, Marc Christie, and Vicky Kalogeiton. 2024. ET the Exceptional Trajectories: Text-to-camera-trajectory generation with character awareness. In *European Conference on Computer Vision*. Springer, 464–480.

CROSSBOW 2008. XBOW Sensor Motes Specifications. http://www.xbow.com.

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

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

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

Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. 2024. Momask: Generative masked modeling of 3d human motions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1900–1910.

Ziyan Guo, Zeyu Hu, Na Zhao, and De Wen Soh. 2025. MotionLab: Unified Human Motion Generation and Editing via the Motion-Condition-Motion Paradigm. *arXiv preprint arXiv:2502.02358* (2025).

Rui He, Huaxin Wei, and Ying Cao. 2024. 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*. 1–15.

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

Qingqiu Huang, Yu Xiong, Anyi Rao, Jiaze Wang, and Dahua Lin. 2020. Movienet: A holistic dataset for movie understanding. In *European conference on computer vision*. Springer, 709–727.

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

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

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

Hongda Jiang, Xi Wang, Marc Christie, Libin Liu, and Baoquan Chen. 2024. Cinematographic camera diffusion model. In *Computer Graphics Forum*, Vol. 43. Wiley Online Library, e15055.

Kacper Kania, Marek Kowalski, and Tomasz Trzciński. 2021. Trajevae: Controllable human motion generation from trajectories. *arXiv preprint arXiv:2104.00351* (2021).

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

Ronghui Li, YuXiang Zhang, Yachao Zhang, Hongwen Zhang, Jie Guo, Yan Zhang, Yebin Liu, and Xiu Li. 2024b. 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.

Ronghui Li, Youliang Zhang, Yachao Zhang, Yuxiang Zhang, Mingyang Su, Jie Guo, Ziwei Liu, Yebin Liu, and Xiu Li. 2024c. InterDance: Reactive 3D Dance Generation with Realistic Duet Interactions. *arXiv preprint arXiv:2412.16982* (2024).

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

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

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. 2015. SMPL: A Skinned Multi-Person Linear Model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)* 34, 6 (Oct. 2015), 248:1–248:16.

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

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

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

Anyi Rao, Jiaze Wang, Linning Xu, Xuekun Jiang, Qingqiu Huang, Bolei Zhou, and Dahua Lin. 2020. A unified framework for shot type classification based on subject centric lens. In *European Conference on Computer Vision*. Springer, 17–34.

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

Mattia Savardi, András Bálint Kovács, Alberto Signoroni, and Sergio Benini. 2021. CineScale: A dataset of cinematic shot scale in movies. *Data in Brief* 36 (2021), 107002.

Mattia Savardi, András Bálint Kovács, Alberto Signoroni, and Sergio Benini. 2023. CineScale2: a dataset of cinematic camera features in movies. *Data in Brief* 51 (2023), 109627.

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

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

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

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

Congyi Wang. 2023. T2m-hifigpt: generating high quality human motion from textual descriptions with residual discrete representations. *arXiv preprint arXiv:2312.10628* (2023).

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

Zixuan Wang, Jiayi Li, Xiaoyu Qin, Shikun Sun, Songtao Zhou, Jia Jia, and Jiebo Luo. 2024b. DanceCamAnimator: Keyframe-Based Controllable 3D Dance Camera Synthesis. In *Proceedings of the 32nd ACM International Conference on Multimedia*. 10200–10209.

Xinyi Wu, Haohong Wang, and Aggelos K Katsaggelos. 2023. The secret of immersion: actor driven camera movement generation for auto-cinematography. *arXiv e-prints* (2023), arXiv–2303.

Kaijie Xiao, Yi Gao, Fu Li, Weifeng Xu, Pengzhi Chen, and Wei Dong. 2024. ChatCam: Embracing LLMs for Contextual Chatting-to-Camera with Interest-Oriented

Video Summarization. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 8, 4 (2024), 1–34.

Chun Xie, Isao Hemmi, Hidehiko Shishido, and Itaru Kitahara. 2023. Camera Motion Generation Method Based on Performer's Position for Performance Filming. In *2023 IEEE 12th Global Conference on Consumer Electronics (GCCE)*. IEEE, 957–960.

Yiming Xie, Varun Jampani, Lei Zhong, Deqing Sun, and Huaizu Jiang. 2024. OmniControl: Control Any Joint at Any Time for Human Motion Generation. arXiv:2310.08580 [cs.CV] https://arxiv.org/abs/2310.08580

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

Mingyuan Zhang, Huirong Li, Zhongang Cai, Jiawei Ren, Lei Yang, and Ziwei Liu. 2023a. Finemogen: Fine-grained spatio-temporal motion generation and editing. *Advances in Neural Information Processing Systems* 36 (2023), 13981–13992.

Yi Zhou, Connelly Barnes, Lu Jingwan, Yang Jimei, and Li Hao. 2019. On the Continuity of Rotation Representations in Neural Networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zixiang Zhou, Yu Wan, and Baoyuan Wang. 2024. 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*. 1357–1366.

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

## A Switching Times

In this appendix, we measure the channel switching time of Micaz [CROSSBOW 2008] sensor devices. In our experiments, one mote alternatingly switches between Channels 11 and 12. Every time after the node switches to a channel, it sends out a packet immediately and then changes to a new channel as soon as the transmission is finished. We measure the number of packets the test mote can send in 10 seconds, denoted as $N_1$. In contrast, we also measure the same value of the test mote without switching channels, denoted as $N_2$. We calculate the channel-switching time $s$ as

$$s = \frac{10}{N_1} - \frac{10}{N_2}.$$

By repeating the experiments 100 times, we get the average channel-switching time of Micaz motes: 24.3 $\mu$s.