Absolute Coordinates Make Motion Generation Easy

Anonymous Author(s)
Affiliation
Address
email

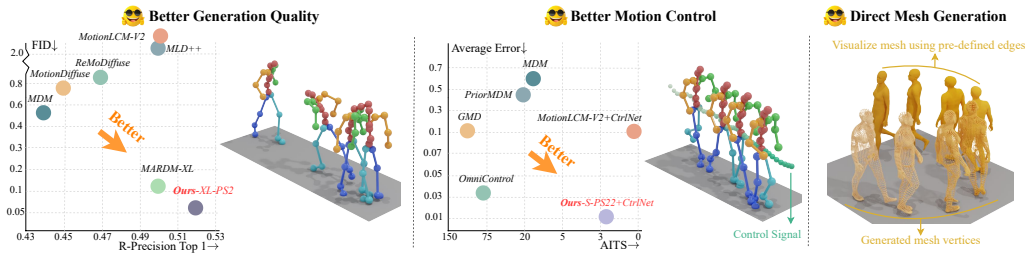

Figure 1: **Absolute coordinates make motion generation easy.** Here we show that our model produces motion of higher fidelity, has better controllability, and reports promising results of generating SMPL-H meshes directly.

Abstract

State-of-the-art text-to-motion generation models rely on the kinematic-aware, local-relative motion representation popularized by HumanML3D, which encodes motion relative to the pelvis and to the previous frame with built-in redundancy. While this design simplifies training for earlier generation models, it introduces critical limitations for diffusion models and hinders applicability to downstream tasks. In this work, we revisit the motion representation and propose a radically simplified and long-abandoned alternative for text-to-motion generation: absolute joint coordinates in global space. Through systematic analysis of design choices, we show that this formulation achieves significantly higher motion fidelity, improved text alignment, and strong scalability, even with a simple Transformer backbone and no auxiliary kinematic-aware losses. Moreover, our formulation naturally supports downstream tasks such as text-driven motion control and temporal/spatial editing without additional task-specific reengineering and costly classifier guidance generation from control signals. Finally, we demonstrate promising generalization to directly generate SMPL-H mesh vertices in motion from text, laying a strong foundation for future research and motion-related applications.

1 Introduction

Generating realistic human motion from textual descriptions has rapidly emerged as a significant research area. It has great potential for diverse applications, including virtual and augmented reality experiences, immersive metaverse environments, video game development, and robotics.

Recently, the introduction of the large-scale HumanML3D [25] dataset has catalyzed significant progress in text-to-motion generation by establishing a standardized, kinematic-aware motion representation. Earlier methods based on AutoEncoders [44, 4], GANs [22], or RNNs [82] attempted to model joints and kinematic rotations or joint and trajectory [1, 78, 5, 109, 55, 19], but struggled to

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

produce high-fidelity motion. HumanML3D instead proposes to encode motion relative to the pelvis and to the previous frame, enabling explicit modeling of intra-frame kinematics and inter-frame transitions. This local-relative, kinematic-aware representation, combined with built-in redundancy (non-animatable features such as incorrectly processed [101] relative rotations, local velocities, and foot contacts) as a form of data-level regularization [66], substantially simplifies training [12, 62, 66] and boosts the performance of these simple backbones. Recent diffusion-based methods [91, 119, 43] also adopt this representation for text-to-motion generation tasks as default, yielding state-of-the-art performances. While later works have explored architectural improvements [10, 90, 125, 33, 122], generation speedups [12, 15, 14], and retrieval-based enhancements [120], the underlying representation has been largely inherited from HumanML3D [25] without much careful study.

However, this *de facto* representation introduces several fundamental limitations. First, although this representation benefits earlier methods, the redundancy makes it difficult for diffusion models to learn [66], often leading to underperformance in generated motion quality. Second, its inherently relative nature is misaligned with the requirements of downstream tasks such as motion control and temporal/spatial editing [102, 42, 77]. These tasks demand motion generation that is not only semantically meaningful but also aware of absolute joint locations, which are usually provided by users, to enable precise control and intuitive motion editing. Attempts to inject absolute location information into the existing local-relative representation have often resulted in overly complex designs [52] and degraded generation fidelity [42, 102, 15, 14].

In this paper, we revisit the foundational question of motion representation for text-driven motion generative models. We begin by demonstrating that the redundant, local-relative, kinematic-aware formulation—commonly assumed to be essential—is not crucial for the performance of diffusion-based models. Instead, we adopt a much simpler and long-abandoned non-kinematic representation in text-to-motion methods: absolute joint coordinates in global space. Through careful analysis of key design choices, we show that even with a simple Transformer [93] model (*e.g.* without UNet [33, 10] or altered attentions [10, 120]) and without additional kinematic losses, this simple formulation can achieve significantly higher motion fidelity, improved text alignment, and strong scalability potential.

Furthermore, we show this simple representation naturally supports a range of downstream tasks, including motion control and temporal/spatial editing, without requiring task-specific reengineering. With inherent absolute location awareness, our formulation enables direct controllability by eliminating the need for relative-to-absolute post-processing, which often introduces errors, as well as removing reliance on time-consuming classifier guidance from control signals during generation.

By discarding the constraints of redundant, local-relative, kinematic-aware representation designs, our approach also opens the door to directly modeling motion from textual inputs beyond standard human joint skeletons. Our formulation shows potential to generalize to other subclasses of absolute coordinates, such as SMPL-H mesh vertices [59] in motion from text, which are largely neglected by existing approaches but crucial toward having vivid, animatable human avatars. This lays a foundation for future research in broader text-to-motion generation domains, enabling new applications across diverse motion-related domains.

In summary, our contributions are as follows:

- We propose a new formulation for text-to-motion diffusion models using absolute joint coordinates. Through systematic analysis of design choices, our method can achieve state-of-the-art performance with simple Transformer [93] backbones and no auxiliary losses.
- We demonstrate that this formulation naturally supports downstream motion tasks, including motion control and temporal/spatial editing, achieving better performance and enabling seamless integration without additional reengineering or time-costly guidance generation.
- We further show promising generalizes beyond joints to directly modeling other subclasses of absolute coordinates, such as mesh vertices. This flexibility marks an important step toward text-driven motion generation across broader domains and serves as a foundation for future research and broader real-world applications.

2 Related Works

Human Motion Generation. Early approaches in text-driven motion generation [1, 25, 70, 71, 89, 110] attempt to align the latent spaces of text and motion. However, these methods faced significant challenges in generating high-fidelity motion due to the difficulty of seamlessly aligning

two fundamentally distinct modalities. Inspired by the success of denoising diffusion models in image generation [29, 86], several pioneering works [91, 43, 119, 12, 115] introduced diffusion-based approaches for human motion generation. Subsequent works have primarily focused on architectural innovations [3, 116, 129, 10, 33, 14, 90, 125, 122, 101] or on improved training methodologies [51, 116, 2, 31, 95, 129, 61, 15, 120]. Other human motion generation works introduce Vector Quantization (VQ), enabling discrete motion token modeling [26, 117, 114, 77, 23, 8, 76, 127, 50, 37, 121, 63, 123] or explore autoregressive generation [124, 11, 85, 125, 90, 66, 101]. Recent works also diversified their focus, exploring human-scene/object interactions [69, 32, 45, 105, 74, 48, 9, 97, 21, 112, 107, 64, 13, 100, 38, 46, 58, 17, 126, 96, 108, 60, 35, 106], human-human interaction [36, 104, 99, 20, 53, 7, 113], stylized human motion generation [128, 24, 49], more datasets [103, 56], long-motion generation [131, 72], shape-aware motion generation [92, 54], fine-grained text controlled generation [132, 34, 111, 84, 39, 81, 88], leveraging 2D data [40, 73, 47], as well as investigating advanced architectures [122, 98]. In contrast, our work revisits the underlying text-to-motion representation itself. We show that adopting a simpler yet long-abandoned alternative: absolute joint coordinates, even with a simple Transformer backbone and no additional constraints, can significantly improve generation quality.

Controllable Text-to-Motion Generation. In addition to synthesizing motion purely from text prompts, recent work has explored controlling motion generation with auxiliary signals such as trajectories or editing constraints [42, 102, 15, 14, 75, 80, 94, 41]. Early approaches such as Prior-MDM [83] extended MDM [91] to support end-effector constraints. GMD [42] introduced spatial control by guiding the diffusion process on the root joint trajectory, but required a re-engineered motion representation specifically designed for the task. OmniControl and MotionLCM [102, 15] generalized control to arbitrary joints by leveraging ControlNet [118], but both still rely on relative motion representations. Moreover, OmniControl heavily depends on classifier guidance from control signals during generation; without it, motion quality degrades significantly. Input optimization-based approaches [41, 75, 14] proposed directly optimizing the inputs to meet control objectives, but suffer from high computational and time costs due to multi-round optimization and gradient accumulations, making real-time applications impractical. In this work, we show that our proposed absolute joint coordinate formulation enables superior performance without the need for task-specific reengineering and time-consuming classifier guidance or inference-time optimization.

Mesh-Level Text-Driven Human Motion Generation. Previous works rarely perform direct mesh vertex generation. Instead, prior methods [91, 12, 102, 42] typically predict HumanML3D representations and convert them to joint positions, followed by SMPL fitting [6]. Other efforts in related fields such as Human-Object Interaction (HOI), Human-Scene Interaction (HSI) and Dual-Person motion generation have attempted to directly model SMPL parameters [32, 45, 21, 64, 46, 100, 38, 58, 126, 96, 104] or joint rotations and translations [74, 48, 112], which are then applied to meshes through standard skinning and rigging techniques. However, SMPL fitting is time-consuming and prone to reconstruction errors, while directly modeling SMPL parameters or joint transformations remains challenging and often results in unsatisfactory mesh quality[69, 46]. Moreover, even small joint-level errors can be magnified when propagated to mesh vertices, degrading the visual fidelity of the synthesized motion. Direct mesh vertex generation from textual inputs remains largely unexplored, yet it is critical for achieving high-fidelity, visually realistic motion synthesis. In this work, we show that with our absolute coordinate formulation, we can naturally extend to directly generating mesh vertices from text and achieve strong performance.

Text-to-Human Motion Representation Early text-to-motion generation methods, often based on AutoEncoders [44, 4] or GANs [22], attempted to directly predict absolute joint positions [1], but struggled to produce realistic motions. Later approaches incorporated human kinematics by predicting joint rotations [78, 5, 109], combining joint positions with trajectory modeling [55, 19]. However, these designs remained limited in producing high-fidelity and semantically aligned motions. The HumanML3D [25] representation addressed these challenges by encoding motion relative to the pelvis and the previous frame, explicitly modeling intra-frame kinematics and inter-frame transitions. Its local-relative, kinematic-aware design, with built-in redundancy [66, 101] from features such as relative rotations, local velocity, and foot contacts, substantially simplified training [12, 62] and quickly became the dominant choice for subsequent text-to-motion generation methodologies. In this work, we demystify the significance of HumanML3D representation formulation and adopt a simpler, long-abandoned non-kinematic formulation: absolute joint coordinates in global space. We show that, with this design, our method achieves better performance using simple Transformer [93]

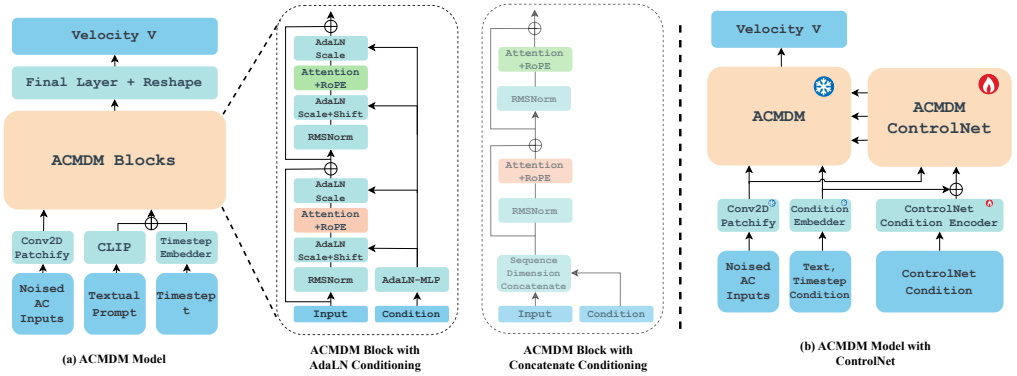

Figure 2: **Overview of our proposed ACMDM.** (a) Left: The raw/latent absolute coordinates representation is patchified and processed through a sequence of ACMDM blocks. Right: Details of ACMDM blocks, where we experiment with two conditioning variants: concatenation and AdaLN. (b) ControlNet-augmented ACMDM for controllable motion generation: Structured control signals are separately encoded and fused into the ACMDM generation process via additive residuals at each ACMDM block, enabling the model to follow both semantical and spatial controlling constraints.

backbones without auxiliary losses, and naturally extends to direct modeling other subclasses of absolute coordinates such as mesh vertices.

3 ACMDM: Absolute Coordinates Motion Diffusion Model

The majority of recent methods utilize the redundant, local-relative, and kinematic-aware motion representation popularized by HumanML3D [25]. However, this explicit inter-frame kinematic modeling around the pelvis makes the generation prone to accumulating global drift errors through frames, while the intra-frame relative formulation makes it difficult to incorporate absolute location controlling signals for downstream tasks. In contrast, we propose adopting a much simpler but long-abandoned alternative, absolute joint coordinates in global 3D space and show it makes human motion generation easy.

We first introduce our proposed ACMDM in Section 3.1 that we will systematically investigate and ablate in the experiments section. Next, in Section 3.2, we describe how to extend ACMDM to controllable motion generation through ControlNet integration without much task-specific engineering. Finally, we show how ACMDM generalizes to direct mesh vertex motion generation in Section 3.3.

3.1 Absolute Joint Coordinates for Text-to-Motion Diffusion

Absolute Coordinates Representation. We define absolute joint coordinates at each frame as $\mathbf{X}^i \in \mathbb{R}^{N_j \times 3}$, where N_j is the number of joints (*e.g.*, 22 for the HumanML3D dataset), and each joint is represented by its 3D global position (XYZ). This intuitive formulation naturally avoids pelvis drift accumulation and facilitates direct controllability over spatial control signals. Previous works generally avoided this representation due to concerns about generating unnatural, non-human-like motions [102]. It was widely believed [102, 12, 62] that kinematic features were essential for physically plausible motion synthesis. In the experiment section, we demonstrate that using redundant kinematic features actually degrades motion generation quality, and that absolute joint positions alone are sufficient to achieve high-fidelity and controllable motion generation.

Tokenizing Motion Representation. Absolute joint coordinates inherently preserve both spatial and temporal structure of the motion data. Given a motion sequence input of shape (L, N_j, d_{in}), where L is the motion sequence length and d_{in} is the input feature dimension (3 for raw absolute coordinates), we apply a 2D convolutional layer to transform this structured input into a sequence of T tokens similar to ViT [18], each with hidden dimension d. The number of tokens T is determined by the predefined patch size (P_T, P_S), where the convolution kernel size and stride are both set equal to (P_T, P_S), resulting in $T = \frac{L}{P_T} \times \frac{N_j}{P_S}$. Importantly, we perform tokenization only along the spatial

(joint) dimension while preserving the full temporal resolution (*i.e.*, we define $P_T = 1$), as temporal details are especially critical for motion modeling. In our design, we explore various patch sizes, including 1×22, 1×11, and 1×2, corresponding to different joint-wise granularities.

Motion Diffusion with Transformer. After tokenizing the absolute coordinate inputs, the resulting token sequence is fed directly into Transformer for diffusion-based motion generation. Note that the central goal of this work is not to advance model architecture for motion generation. Rather, we focus on investigating the absolute coordinates motion representation. Therefore, we simply adopt a simple Transformer similar to DiT [68] and found it works sufficiently well.

To incorporate conditioning signals, we follow prior works [91, 117, 23, 77, 66, 76, 12, 120] and use a pretrained CLIP-B/32 [79] text encoder to extract the textual embedding c, along with a timestep embedder to process diffusion timestep t. We explore two conditioning mechanisms within our ACMDM design: (1) Concatenation, a commonly used method in prior text-to-motion works [91, 117, 23, 77, 76, 12], where condition vectors are appended along the sequence dimension; and (2) AdaLN, where the text and timestep embeddings modulate each block via adaptive layer normalization, similar to image diffusion DiT [68]. An illustration of these variants are shown in Figure 2 (a). In line with recent best practices in Transformer models, we also adopt several modern architectural components: Rotary Positional Embedding (RoPE) [87] and QK Normalization [28] are applied in the attention layers, and SwiGLU activations[67] are used in the feed-forward networks (FFNs). We also investigate different denoising targets for training ACMDM, including predicting $\mathbf{x}_0$ [29] (the original motion), $\boldsymbol{\epsilon}$ [29] (the added noise), and velocity [57] $\mathbf{v}$ (under flow-matching formulations). In our experimental analysis, we show that $\mathbf{v}$ prediction consistently yields the best generation performance. All ACMDM variants are trained with a standard L_2 reconstruction loss on the diffusion objective. More details are provided in the supplemental material.

After processing through the motion diffusion Transformer, the output token sequence is linearly projected to match the original shape. Specifically, a linear layer is applied to transform each token from dimension d back to $d_{in} \times P_T \times P_S$. The output is then reshaped to recover the original 2D structure (*i.e.*, (L, N_j, d_{in})) of the absolute joint coordinates.

Latent Motion Encoding with a Motion AutoEncoder. Optionally, we convert raw absolute coordinates into latents using a motion autoencoder (AE) and perform motion diffusion then, which leads to better generation fidelity as shown in the experiment section. Specifically, given a motion sequence $\mathbf{X}^{0:N} \in \mathbb{R}^{L \times N_j \times 3}$, a 2D ResNet-based encoder compresses it into a latent representation $\mathbf{x}^{0:n} \in \mathbb{R}^{l \times N_j \times d_j}$, where l denotes the downsampled motion sequence length and d_j is the dimension of the motion latent. We keep the number of joints N_j unchanged here. Tokenization is then performed over the latent representations (so $d_{in} = d_j$), whose output will be fed into the motion diffusion Transformer. A decoder later can reconstruct the motion sequence $\hat{\mathbf{X}}^{0:N} \in \mathbb{R}^{L \times N_j \times 3}$ via nearest-neighbor upsampling based on the diffusion output. We explore a causal AE (*i.e.*, convolution kernels can only access previous frames), a non-causal AE, a VAE-based variant, and direct modeling on raw absolute joint coordinates in the experimental section. All these motion AE variants are trained with a simple smooth L_1 reconstruction loss. More details of all the AE variants are provided in the supplemental material.

Scaling ACMDM. We scale the model capacity by increasing the motion diffusion Transformer layer's depth and width. Specifically, we follow a simple scaling strategy where the number of Transformer layers is set equal to the number of attention heads. We define four model sizes: ACMDM-S, ACMDM-B, ACMDM-L, and ACMDM-XL, corresponding to configurations with 8, 12, 16, and 20 layers and attention heads, respectively. This consistent scaling scheme enables systematic exploration of ACMDM's capacity and its effect on generation quality. In addition, we also vary the patch sizes for tokenization. We name different model variants according to their model and patch size (for tokenization); *e.g.*, ACMDM-XL-PS2 refers to the XL variant with a patch size of 1×2.

3.2 Adding Controls to Absolute Joint Coordinates Generation

Most prior methods face significant challenges in controllable motion generation due to their reliance on local-relative representations, which naturally misalign with user-provided absolute coordinates control signals. In contrast, our absolute coordinates representation removes this misalignment, enabling seamless integration of control without classifier guidance [16] and input optimization [41].

To enable controllable text-driven motion generation, such as trajectory conditioning and temporal/spatial editing with absolute joint coordinates, we follow prior works [102, 15, 14] and integrate a ControlNet [118]-style module into the ACMDM architecture. As shown in Figure 2 (b), the noised absolute coordinate latent is first tokenized via a 2D convolutional layer and then fed into both the main ACMDM and a parallel ControlNet module. At the same time, textual and timestep conditions are encoded and provided to both ACMDM and the ControlNet as conditioning embeddings. Separately, structured control signals (*e.g.*, joint trajectories or partial-body constraints) are processed through a dedicated ControlNet condition encoder. The ControlNet receives both the tokenized noised inputs as well as control-specific features in additive combination with the textual and timestep embeddings. These fused features generate residuals, which are injected into the main ACMDM backbone at each layers via additive fusion. This modulation enables the model to follow both semantic instructions and structural constraints. In addition to the standard L_2 reconstruction loss on the diffusion target, we also apply an L_2 loss between the model's prediction and the control signal. We also freeze the parameters of the main ACMDM and only train the ControlNet branch, which is initialized as copies of the main ACMDM blocks, similar to prior works [118, 102, 15, 14].

3.3 Generating Meshes with Absolute Coordinates Representation

Towards achieving vivid, animatable human avatars, joint representations are insufficient; when translated to meshes through fitting models, they often result in shaky body parts, unnatural hand motions, and missing flesh dynamics [91, 12, 15, 14]. Direct motion generation at the mesh level, however, largely falls behind joint counterparts, mainly due to the complexity of modeling mesh representations. Here, we show that our absolute, non-kinematic representation naturally extends to mesh vertices, which is seamlessly supported by ACMDM without major architectural changes.

In specific, we explore direct motion generation of SMPL-H [59] mesh vertices, where each frame is represented as a set of absolute 3D vertex coordinates with shape $(L, N_v, 3)$, where $N_v = 6890$ denotes the number of vertices. Unlike absolute joint coordinates, where the number of joints N_j is typically small, directly training diffusion models on full-resolution mesh data with $N_v = 6890$ is computationally prohibitive and unstable. To address this, we incorporate a 2D mesh autoencoder based on the Fully Convolutional Mesh Autoencoder [130]. The encoder spatially compresses the input mesh sequence $(L, N_v, 3)$ into a latent representation of shape (L, n_v, d_v), where we set $n_v = 28$ for diffusion modeling efficiency and reconstruction quality. Once mesh vertices are encoded, we reuse the ACMDM framework to perform motion diffusion in this latent mesh space. The resulting sequence is tokenized using patch sizes of 1×28 and processed with the same formulation as our joint-based ACMDM. In the experiment section, we show the flexibility and scalability of our approach for high-fidelity motion generation over mesh vertices as well in addition to human joints.

4 Experiment

4.1 Datasets, Training Setups, and Evaluation Protocols

Datasets. To fairly evaluate different ACMDM designs and compare against prior models, we adopt the widely used HumanML3D [25] benchmark for standard text-to-motion generation, downstream tasks such as text-driven trajectory-controlled generation and upper-body editing, and direct text-to-SMPL-H mesh motion generation. We also include text-to-motion evaluations on KIT-ML [78], reported in the Appendix. HumanML3D contains 14,616 motion sequences sourced from AMASS [65] and HumanAct12 [27], each paired with three textual descriptions (44,970 annotations in total). All motions are standardized to 20 FPS and capped at 10 seconds. It is augmented via mirroring and split into training, validation, and test sets using a standard 80%/15%/5% split.

Training Setups. All ACMDM variants are trained using the AdamW optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.99$. We use a batch size of 64 with a maximum sequence length of 196 frames. The learning rate is initialized at 2×10^{-4} and linearly warmed up over the first 2,000 steps. We apply a learning rate decay by a factor of 0.1 at 50,000 iterations during the training of 500 epochs. We also use an exponential moving average (EMA) of model weights to improve training stability and performance. During inference, we apply classifier-free guidance (CFG) [30] = 3 for text-to-motion generation and upper-body editing, 2.5 for trajectory control, and 4.5 for text-to-SMPL-H mesh motion generation.

Table 1: **Ablation study of the design choices of ACMDM on the HumanML3D dataset.** The results indicate that kinematic-aware redundancy is not necessary. Instead, absolute coordinates motion representation can achieve high-quality motion generation with AdaLN conditioning, the velocity diffusion objective ($\mathbf{v}$), and latent space modeling.

Motion Representation	Conditioning Mechanism	Motion AE	Diffusion Objective	FID ↓	R-Precision Top 1↑	R-Precision Top 2↑	R-Precision Top 3↑	Matching↓
Absolute+Redundancy	Concat	✗	$\mathbf{x}_0$	$0.771^{\pm.020}$	$0.441^{\pm.002}$	$0.633^{\pm.003}$	$\mathbf{0.738}^{\pm.002}$	$3.632^{\pm.009}$
			ϵ	$0.868^{\pm.030}$	$0.358^{\pm.003}$	$0.538^{\pm.005}$	$0.650^{\pm.004}$	$4.168^{\pm.025}$
			$\mathbf{v}$	$\mathbf{0.276}^{\pm.006}$	$\mathbf{0.445}^{\pm.002}$	$\mathbf{0.634}^{\pm.002}$	$\mathbf{0.738}^{\pm.002}$	$\mathbf{3.613}^{\pm.008}$
Absolute	Concat	✗	$\mathbf{x}_0$	$0.969^{\pm.029}$	$0.356^{\pm.003}$	$0.539^{\pm.004}$	$0.648^{\pm.003}$	$4.362^{\pm.013}$
			ϵ	$0.419^{\pm.013}$	$0.436^{\pm.002}$	$0.630^{\pm.003}$	$0.736^{\pm.003}$	$3.717^{\pm.013}$
			$\mathbf{v}$	$\mathbf{0.208}^{\pm.012}$	$\mathbf{0.451}^{\pm.003}$	$\mathbf{0.643}^{\pm.003}$	$\mathbf{0.751}^{\pm.002}$	$\mathbf{3.544}^{\pm.010}$
Absolute	AdaLN	✗	$\mathbf{x}_0$	$0.133^{\pm.004}$	$0.485^{\pm.002}$	$0.680^{\pm.002}$	$0.779^{\pm.002}$	$3.386^{\pm.012}$
			ϵ	$0.125^{\pm.007}$	$0.493^{\pm.002}$	$0.685^{\pm.003}$	$0.783^{\pm.002}$	$3.343^{\pm.009}$
			$\mathbf{v}$	$\mathbf{0.121}^{\pm.006}$	$\mathbf{0.502}^{\pm.002}$	$\mathbf{0.692}^{\pm.003}$	$\mathbf{0.789}^{\pm.003}$	$\mathbf{3.304}^{\pm.008}$
Absolute	AdaLN	Causal AE	$\mathbf{x}_0$	$0.137^{\pm.007}$	$0.473^{\pm.002}$	$0.670^{\pm.002}$	$0.772^{\pm.003}$	$3.451^{\pm.011}$
			ϵ	$0.188^{\pm.006}$	$0.475^{\pm.003}$	$0.670^{\pm.002}$	$0.775^{\pm.002}$	$3.393^{\pm.012}$
			$\mathbf{v}$	$\mathbf{0.109}^{\pm.005}$	$\mathbf{0.508}^{\pm.002}$	$\mathbf{0.701}^{\pm.003}$	$\mathbf{0.798}^{\pm.003}$	$\mathbf{3.253}^{\pm.010}$
Absolute	AdaLN	Non-Causal VAE	$\mathbf{v}$	$0.178^{\pm.006}$	$0.497^{\pm.002}$	$0.687^{\pm.003}$	$0.785^{\pm.004}$	$3.323^{\pm.010}$
		Non-Causal AE	$\mathbf{v}$	$0.150^{\pm.005}$	$0.502^{\pm.003}$	$0.693^{\pm.003}$	$0.787^{\pm.003}$	$3.296^{\pm.010}$
		Causal VAE	$\mathbf{v}$	$0.115^{\pm.005}$	$0.504^{\pm.002}$	$0.697^{\pm.002}$	$0.795^{\pm.003}$	$3.278^{\pm.011}$

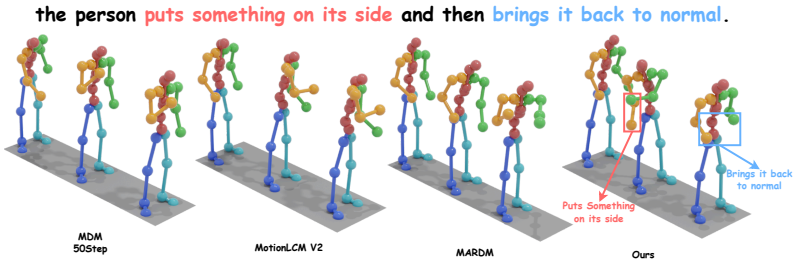

Figure 3: **Visual comparisons of generated motion between ACMDM and state-of-the-art methods.** ACMDM generates more realistic motion that accurately follows the textual condition.

Evaluation Metrics. We adopt the robust evaluation framework proposed by [66], focusing on essential, animatable motion features. Following [25, 66], we report: (1) R-Precision (Top-1/2/3) and Matching (semantic alignment with captions); (2) FID (distribution similarity); (3) MultiModality (motion diversity per prompt); and (4) CLIP-Score (cosine similarity between motion and caption embeddings). For trajectory-control evaluations [42], we additionally report Diversity (variability within generated motions), Foot Skating Ratio, Trajectory Error, Location Error, and Average Joint Error (accuracy of controlled joints at keyframes). Metrics are averaged over five levels of control intensity (1%, 2%, 5%, 25%, 100%). During training, control intensity levels are randomly sampled. For direct SMPL-H mesh generation, we also report Laplacian Surface Distance (LSD) to assess mesh structural preservation relative to the ground-truth T-pose. More metric details are in Appendix.

4.2 Ablating ACMDM Designs

Necessity of Kinematic-aware and Redundant Motion Representation. Prior attempts [102] of text-to-absolute-coordinate motion generation adopt InterGen [52]'s representation with heavy kinematic-aware redundancy and the $\mathbf{x}_0$ objective, but result in unrealistic motion. To systematically analyze this, in the top two sections of Table 1, we train an ACMDM-S-PS22 variant. We match the model size and flattened spatial embedding style used in prior works in two settings: one using absolute coordinates with kinematic-aware and redundant representation (*i.e.*, InterGen's representation), and another using plain absolute coordinates (our proposed). The results show that while the previously widely adopted $\mathbf{x}_0$-prediction diffusion benefits slightly from the redundancy, velocity prediction ($\mathbf{v}$) with plain absolute coordinates (our proposed) achieves better performance. Notably, by modeling plain absolute coordinates with $\mathbf{v}$ prediction, ACMDM achieves a FID that is **0.563 lower** and an R-Precision Top-3 score that is **0.013 higher** compared to redundant $\mathbf{x}_0$ prediction. These results demonstrate that with a more suitable diffusion objective ($\mathbf{v}$ prediction), and the previously assumed necessary kinematic-aware redundancy is not required for achieving high-quality motion generation. Therefore, for the rest of the paper, all ACMDM models will adopt the pure absolute coordinates representation without any kinematic-aware or redundant features.

Table 2: **Quantitative text-to-motion evaluation.** We repeat the evaluation 20 times and report the average with 95% confidence interval. We use **bold** face / underline to indicate the best/2nd results.

Methods	FID↓	R-Precision↑			Matching↓	MModality↑	CLIP-score↑
		Top 1	Top 2	Top 3			
Real	$0.000^{\pm.000}$	$0503^{\pm.002}$	$0.696^{\pm.001}$	$0.795^{\pm.002}$	$3.244^{\pm.005}$	-	$0.639^{\pm.001}$
MDM-50Step [91]	$0.518^{\pm.032}$	$0.440^{\pm.007}$	$0.636^{\pm.006}$	$0.742^{\pm.004}$	$3.640^{\pm.028}$	$\mathbf{3.604}^{\pm.031}$	$0.578^{\pm.003}$
MotionDiffuse [119]	$0.778^{\pm.005}$	$0.450^{\pm.006}$	$0.641^{\pm.005}$	$0.753^{\pm.005}$	$3.490^{\pm.023}$	$\underline{3.179}^{\pm.046}$	$0.606^{\pm.004}$
ReMoDiffuse [120]	$0.883^{\pm.021}$	$0.468^{\pm.003}$	$0.653^{\pm.003}$	$0.754^{\pm.005}$	$3.414^{\pm.020}$	$2.703^{\pm.154}$	$0.621^{\pm.003}$
MLD++ [14]	$2.027^{\pm.021}$	$0.500^{\pm.003}$	$0.691^{\pm.002}$	$0.789^{\pm.001}$	$3.220^{\pm.008}$	$1.924^{\pm.065}$	$0.639^{\pm.002}$
MotionLCM V2 [14]	$2.267^{\pm.023}$	$0.501^{\pm.002}$	$0.693^{\pm.002}$	$0.790^{\pm.002}$	$3.192^{\pm.009}$	$1.780^{\pm.062}$	$0.640^{\pm.003}$
MARDM [66]-ϵ	$0.116^{\pm.004}$	$0.492^{\pm.006}$	$0.690^{\pm.005}$	$0.790^{\pm.005}$	$3.349^{\pm.010}$	$2.470^{\pm.053}$	$0.637^{\pm.005}$
MARDM [66]-v	$0.114^{\pm.007}$	$0.500^{\pm.004}$	$0.695^{\pm.003}$	$0.795^{\pm.003}$	$3.270^{\pm.009}$	$2.231^{\pm.071}$	$0.642^{\pm.002}$
ACMDM-S-PS22	$\underline{0.109}^{\pm.005}$	$\underline{0.508}^{\pm.002}$	$\underline{0.701}^{\pm.003}$	$\underline{0.798}^{\pm.003}$	$\underline{3.253}^{\pm.010}$	$2.156^{\pm.061}$	$\underline{0.642}^{\pm.001}$
ACMDM-XL-PS2	$\mathbf{0.058}^{\pm.004}$	$\mathbf{0.522}^{\pm.002}$	$\mathbf{0.713}^{\pm.002}$	$\mathbf{0.807}^{\pm.002}$	$\mathbf{3.205}^{\pm.008}$	$2.077^{\pm.083}$	$\mathbf{0.652}^{\pm.001}$

Concatenation *vs.* AdaLN. In the third section of Table 1, we switch from the widely adopted concatenation-based conditioning to AdaLN conditioning with an ACMDM-S-PS22 variant with pure absolute coordinates. Our results show that across all diffusion objectives, better conditioning mechanism (AdaLN) lead to significant improvements. Notably, with $\mathbf{v}$ prediction, ACMDM achieves an FID of **0.121** and an R-Precision Top-3 score of **0.789**, substantially outperforming concatenation-based conditioning. These findings demonstrate that an effective conditioning mechanism is a key factor in achieving high-quality motion generation. Therefore, for all subsequent experiments, we adopt AdaLN-based conditioning mechanism across all ACMDM models.

Raw Absolute Coordinates *vs.* Latent Space. In the fourth section of Table 1, we switch from directly modeling raw absolute coordinates to a latent space. Our results show that latent space modeling further improves generation quality while also offering faster inference for $\mathbf{v}$ prediction, achieving the best FID of **0.109** and R-Precision Top-3 score of **0.798** We additionally compare different AutoEncoder variants: Causal-AE, Non-Causal-AE, and VAE in the last section of Table 1. Among them, Causal-AE achieves the best overall performance. Therefore, for all subsequent experiments, we adopt Causal-AE as our default setup. Since velocity ($\mathbf{v}$) prediction consistently yields the best performance across all settings, we also adopt it as the default diffusion objective.

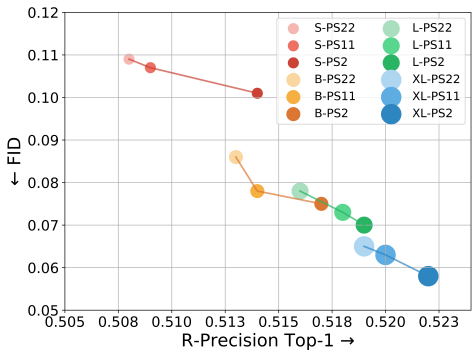

Figure 4: **Scaling of ACMDM with model capacity and decreasing patch size.** We use red for S, orange for B, green for L, and blue for XL, with color gradients indicating decreasing patch sizes. ACMDM exhibits strong scalability, with performance consistently improving as model size increases and patch size decreases.

Scaling Model and Decreasing Patch Sizes. In Figure 4, we train 12 ACMDM models over all model configs (S, B, L, XL) and patch sizes (1×22, 1×11, 1×2). In all cases, we find that increasing model size and decreasing patch size lead to improved text-to-motion generation performance both with and without CFG across all metrics. Notably, ACMDM-XL-PS2 achieves an FID of **0.058** and an R-Precision Top-1 score of **0.522**, outperforming the most recent state-of-the-art MARDM by **0.056** in FID and **0.022** in R-Precision Top-1. These findings demonstrate the effectiveness of scaling model capacity and decreasing patch sizes with absolute joint coordinates. We include detailed results in Appendix.

4.3 Comparison to State-of-the-Art Text-to-Motion Generation Methods

We present the quantitative comparison between our method and state-of-the-art text-to-motion generation baselines in Table 2, as well as qualitative comparison in Figure 3 and Appendix. As observed, our method achieves superior performance across multiple key metrics, including FID, R-Precision, Matching Score, and CLIP-Score. Compared to existing approaches, ACMDM demonstrates a significantly stronger ability to generate high-fidelity, semantically aligned motions that closely follow textual instructions. Notably, even for our smallest ACMDM variant, ACMDM-S-PS22, it outperforms all prior state-of-the-art methods. Larger ACMDM models, such as ACMDM-XL-PS2, further amplify the performance gains across all evaluation metrics.

Table 3: **Quantitative text-conditioned motion generation with spatial control signals and upper-body editing on HumanML3D.** In the first section, methods are trained and evaluated solely on pelvis controls. In the middle section, methods are trained on all joints and evaluated separately on each controlled joint. Only average results are reported for brevity. We include details in Appendix. Last section presents upper-body editing results. **bold** face / underline indicates the best/2nd results.

Controlling Joint	Methods	AITS↓	Classifier Guidance	FID↓	R-Precision Top 3	Diversity→	Foot Skating Ratio.↓	Traj. err.↓	Loc. err.↓	Avg. err.↓
	GT	–	-	0.000	0.795	10.455	-	0.000	0.000	0.000
Train On Pelvis	MDM [91]	16.34	✗	1.792	0.673	9.131	0.1019	0.4022	0.3076	0.5959
	PriorMDM [83]	20.19	✗	0.393	0.707	9.847	0.0897	0.3457	0.2132	0.4417
	GMD [42]	137.63	✓	0.238	0.763	10.011	0.1009	0.0931	0.0321	0.1439
	OmniContol [102]	81.00	✓	0.081	0.789	10.323	**0.0547**	0.0387	0.0096	0.0338
	MotionLCM V2+CtrlNet [14]	**0.066**	✗	3.978	0.738	9.249	0.0901	0.1080	0.0581	0.1386
	ACMDM-S-PS22+CtrlNet	2.51	✗	**0.067**	**0.805**	**10.481**	0.0591	**0.0075**	**0.0010**	**0.0100**
Train On All Joints (Average)	OmniContol [102]	81.00	✓	0.126	0.792	10.276	0.0608	0.0617	0.0107	0.0404
	MotionLCM V2+CtrlNet [14]	**0.066**	✗	4.504	0.715	9.230	0.1119	0.2740	0.1315	0.2464
	ACMDM-S-PS22+CtrlNet	2.51	✗	**0.070**	**0.803**	**10.526**	**0.0596**	**0.0117**	**0.0019**	**0.0197**

	Methods	AITS↓	Classifier Guidance	FID↓	R-Precision Top 1	R-Precision Top 2	R-Precision Top 3	Matching↓	Diversity→	-
UpperBody Edit	MDM [91]	16.34	✗	1.918	0.359	0.556	0.654	4.793	9.210	-
	OmniControl [119]	81.00	✓	0.909	0.428	0.614	0.722	3.694	10.207	-
	MotionLCM V2+CtrlNet [119]	**0.066**	✗	3.922	0.404	0.592	0.692	5.610	9.309	-
	ACMDM-S-PS22+CtrlNet	2.51	✗	**0.076**	**0.532**	**0.719**	**0.820**	**3.098**	**10.586**	-

Table 4: **Quantitative results** for direct text-to-SMPL-H mesh motion generation on HumanML3D.

Size	Transformer	FID ↓	R-Precision Top 1 ↑	R-Precision Top 2 ↑	R-Precision Top 3 ↑	Matching↓	CLIP-score↑	LSD↓
S	8 head 512 dim	$0.211^{\pm.005}$	$0.478^{\pm.004}$	$0.682^{\pm.003}$	$0.784^{\pm.003}$	$3.405^{\pm.011}$	$0.620^{\pm.002}$	$0.0026^{\pm.0002}$
B	12 head 768 dim	$0.181^{\pm.003}$	$0.490^{\pm.003}$	$0.691^{\pm.003}$	$0.783^{\pm.002}$	$3.345^{\pm.010}$	$0.631^{\pm.001}$	$\mathbf{0.0024^{\pm.0002}}$
L	16 head 1024 dim	$0.160^{\pm.004}$	$0.497^{\pm.003}$	$0.696^{\pm.002}$	$0.790^{\pm.002}$	$3.341^{\pm.009}$	$0.633^{\pm.0}$	$0.0025^{\pm.0001}$
XL	20 head 1280 dim	$\mathbf{0.139^{\pm.003}}$	$\mathbf{0.498^{\pm.003}}$	$\mathbf{0.704^{\pm.003}}$	$\mathbf{0.794^{\pm.003}}$	$\mathbf{3.309^{\pm.007}}$	$\mathbf{0.636^{\pm.001}}$	$0.0025^{\pm.0001}$

4.4 Comparison to State-of-the-Art Controllable Motion Generation Methods

We present quantitative comparisons between our method and state-of-the-art methods on text-driven trajectory control and upper-body editing in Table 3. For the trajectory control task, prior works [42, 102, 14] have shown that inference-time classifier guidance is crucial for achieving strong control performance. However, we show that even with our smallest ACMDM variant that matches to baseline model sizes and embedding formats, our absolute coordinate formulation achieves superior motion fidelity and control accuracy without the need for time-consuming classifier guidance from control signals. This results in significantly faster generation compared to guidance-dependent approaches (**2.51** *v.s.* **81.0** seconds). For the upper-body editing task, we follow the evaluation protocol proposed by [77, 75], where we fix the pelvis, left foot, and right foot joints and edit the upper body motion according to textual prompts. Our method achieves substantially better generation quality across all evaluation metrics, validating the effectiveness of our proposed approach.

4.5 Evaluations on Absolute Mesh Vertex Coordinates Motion Generation

We evaluate ACMDM on SMPL-H absolute mesh vertex coordinates motion generation in Table 4. We train and compare four ACMDM model sizes—S, B, L, and XL, with the patch size of 1×28. Despite the significantly increased complexity of modeling full mesh sequences compared to joint sequences, our ACMDM models still achieve strong performance. Notably, all variants achieve results competitive with the best text-to-joint generation models, while operating directly on high-dimensional vertex spaces. This highlights the effectiveness and flexibility of our absolute coordinates motion representation in handling broader motion generation tasks beyond human joints.

5 Conclusion

In conclusion, we presented ACMDM, a novel text-driven motion diffusion framework built on an absolute coordinates motion representation. We run extensive analysis to identify an optimal setting, including the velocity prediction diffusion objective, optimized conditioning mechanisms (AdaLN), and latent motion representation. Our model naturally supports downstream control tasks, which removes the misalignment between local motion representation and absolute controlling, and also generalizes to direct SMPL-H mesh vertices motion generation. Extensive experiments demonstrate that ACMDM achieves superior performance and scalability across text-to-motion benchmarks.

References

[1] Chaitanya Ahuja and Louis-Philippe Morency. Language2pose: Natural language grounded pose forecasting. In *2019 International conference on 3D vision (3DV)*, pages 719–728. IEEE, 2019.

[2] Nefeli Andreou, Xi Wang, Victoria Fernández Abrevaya, Marie-Paule Cani, Yiorgos Chrysanthou, and Vicky Kalogeiton. Lead: Latent realignment for human motion diffusion. *arXiv preprint arXiv:2410.14508*, 2024.

[3] Samaneh Azadi, Akbar Shah, Thomas Hayes, Devi Parikh, and Sonal Gupta. Make-an-animation: Large-scale text-conditional 3d human motion generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15039–15048, 2023.

[4] Dor Bank, Noam Koenigstein, and Raja Giryes. Autoencoders. *Machine learning for data science handbook: data mining and knowledge discovery handbook*, pages 353–374, 2023.

[5] Uttaran Bhattacharya, Nicholas Rewkowski, Abhishek Banerjee, Pooja Guhan, Aniket Bera, and Dinesh Manocha. Text2gestures: A transformer-based network for generating emotive body gestures for virtual agents. In *2021 IEEE virtual reality and 3D user interfaces (VR)*, pages 1–10. IEEE, 2021.

[6] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*, pages 561–578. Springer, 2016.

[7] Zhi Cen, Huaijin Pi, Sida Peng, Qing Shuai, Yujun Shen, Hujun Bao, Xiaowei Zhou, and Ruizhen Hu. Ready-to-react: Online reaction policy for two-character interaction generation. In *The Thirteenth International Conference on Learning Representations*, 2025.

[8] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11315–11325, 2022.

[9] Jianqi Chen, Panwen Hu, Xiaojun Chang, Zhenwei Shi, Michael Christian Kampffmeyer, and Xiaodan Liang. Sitcom-crafter: A plot-driven human motion generation system in 3d scenes. *arXiv preprint arXiv:2410.10790*, 2024.

[10] 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.

[11] 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.

[12] Xin Chen, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, and Gang Yu. Executing your commands via motion diffusion in latent space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18000–18010, 2023.

[13] Peishan Cong, Ziyi Wang, Zhiyang Dou, Yiming Ren, Wei Yin, Kai Cheng, Yujing Sun, Xiaoxiao Long, Xinge Zhu, and Yuexin Ma. Laserhuman: Language-guided scene-aware human motion generation in free environment. *arXiv preprint arXiv:2403.13307*, 2024.

[14] Wenxun Dai, Ling-Hao Chen, Yufei Huo, Jingbo Wang, Jinpeng Liu, Bo Dai, and Yansong Tang. Real-time controllable motion generation via latent consistency model. *arXiv preprint*, 2024.

[15] 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 *ECCV*, pages 390–408, 2025.

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

[17] Christian Diller and Angela Dai. Cg-hoi: Contact-guided 3d human-object interaction generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19888–19901, 2024.

[18] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[19] Anindita Ghosh, Noshaba Cheema, Cennet Oguz, Christian Theobalt, and Philipp Slusallek. Synthesis of compositional animations from textual descriptions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1396–1406, 2021.

[20] Anindita Ghosh, Rishabh Dabral, Vladislav Golyanik, Christian Theobalt, and Philipp Slusallek. Remos: Reactive 3d motion synthesis for two-person interactions. *arXiv preprint arXiv:2311.17057*, 2023.

[21] Jingyu Gong, Chong Zhang, Fengqi Liu, Ke Fan, Qianyu Zhou, Xin Tan, Zhizhong Zhang, Yuan Xie, and Lizhuang Ma. Diffusion implicit policy for unpaired scene-aware motion synthesis. *arXiv preprint arXiv:2412.02261*, 2024.

[22] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[23] 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.

[24] Chuan Guo, Yuxuan Mu, Xinxin Zuo, Peng Dai, Youliang Yan, Juwei Lu, and Li Cheng. Generative human motion stylization in latent space. *arXiv preprint arXiv:2401.13505*, 2024.

[25] 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 (CVPR)*, pages 5152–5161, June 2022.

[26] Chuan Guo, Xinxin Zuo, Sen Wang, and Li Cheng. Tm2t: Stochastic and tokenized modeling for the reciprocal generation of 3d human motions and texts. In *European Conference on Computer Vision*, pages 580–597. Springer, 2022.

[27] Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2motion: Conditioned generation of 3d human motions. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2021–2029, 2020.

[28] Alex Henry, Prudhvi Raj Dachapally, Shubham Pawar, and Yuxuan Chen. Query-key normalization for transformers. *arXiv preprint arXiv:2010.04245*, 2020.

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

[30] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[31] Vincent Tao Hu, Wenzhe Yin, Pingchuan Ma, Yunlu Chen, Basura Fernando, Yuki M Asano, Efstratios Gavves, Pascal Mettes, Bjorn Ommer, and Cees GM Snoek. Motion flow matching for human motion synthesis and editing. *arXiv preprint arXiv:2312.08895*, 2023.

[32] Siyuan Huang, Zan Wang, Puhao Li, Baoxiong Jia, Tengyu Liu, Yixin Zhu, Wei Liang, and Song-Chun Zhu. Diffusion-based generation, optimization, and planning in 3d scenes. In *CVPR*, 2023.

[33] 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.

[34] Yiming Huang, Weilin Wan, Yue Yang, Chris Callison-Burch, Mark Yatskar, and Lingjie Liu. Como: Controllable motion generation through language guided pose code editing. In *European Conference on Computer Vision*, pages 180–196. Springer, 2025.

[35] Inwoo Hwang, Bing Zhou, Young Min Kim, Jian Wang, and Chuan Guo. Scenemi: Motion in-betweening for modeling human-scene interactions. *arXiv preprint arXiv:2503.16289*, 2025.

[36] Muhammad Gohar Javed, Chuan Guo, Li Cheng, and Xingyu Li. Intermask: 3d human interaction generation via collaborative masked modelling. *arXiv preprint arXiv:2410.10010*, 2024.

[37] Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. Motiongpt: Human motion as a foreign language. *Advances in Neural Information Processing Systems*, 36:20067–20079, 2023.

[38] Nan Jiang, Zhiyuan Zhang, Hongjie Li, Xiaoxuan Ma, Zan Wang, Yixin Chen, Tengyu Liu, Yixin Zhu, and Siyuan Huang. Scaling up dynamic human-scene interaction modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1737–1747, 2024.

[39] Peng Jin, Yang Wu, Yanbo Fan, Zhongqian Sun, Wei Yang, and Li Yuan. Act as you wish: Fine-grained control of motion diffusion model with hierarchical semantic graphs. *Advances in Neural Information Processing Systems*, 36, 2024.

[40] Roy Kapon, Guy Tevet, Daniel Cohen-Or, and Amit H Bermano. Mas: Multi-view ancestral sampling for 3d motion generation using 2d diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1965–1974, 2024.

[41] Korrawe Karunratanakul, Konpat Preechakul, Emre Aksan, Thabo Beeler, Supasorn Suwajanakorn, and Siyu Tang. Optimizing diffusion noise can serve as universal motion priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1334–1345, 2024.

[42] Korrawe Karunratanakul, Konpat Preechakul, Supasorn Suwajanakorn, and Siyu Tang. Guided motion diffusion for controllable human motion synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2151–2162, 2023.

[43] Jihoon Kim, Jiseob Kim, and Sungjoon Choi. Flame: Free-form language-based motion synthesis & editing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 8255–8263, 2023.

[44] Diederik P Kingma, Max Welling, et al. Auto-encoding variational bayes, 2013.

[45] Nilesh Kulkarni, Davis Rempe, Kyle Genova, Abhijit Kundu, Justin Johnson, David Fouhey, and Leonidas Guibas. Nifty: Neural object interaction fields for guided human motion synthesis. In *CVPR*, 2024.

[46] Jiaman Li, Alexander Clegg, Roozbeh Mottaghi, Jiajun Wu, Xavier Puig, and C Karen Liu. Controllable human-object interaction synthesis. In *European Conference on Computer Vision*, pages 54–72. Springer, 2025.

[47] Jiaman Li, C Karen Liu, and Jiajun Wu. Lifting motion to the 3d world via 2d diffusion. *arXiv preprint arXiv:2411.18808*, 2024.

[48] Jiaman Li, Jiajun Wu, and C Karen Liu. Object motion guided human motion synthesis. *ACM Transactions on Graphics (TOG)*, 42(6):1–11, 2023.

[49] Zhe Li, Yisheng He, Lei Zhong, Weichao Shen, Qi Zuo, Lingteng Qiu, Zilong Dong, Laurence Tianruo Yang, and Weihao Yuan. Mulsmo: Multimodal stylized motion generation by bidirectional control flow. In *arXiv 2412.09901*, 2024.

[50] Zhe Li, Weihao Yuan, Yisheng He, Lingteng Qiu, Shenhao Zhu, Xiaodong Gu, Weichao Shen, Yuan Dong, Zilong Dong, and Laurence T. Yang. Lamp: Language-motion pretraining for motion generation, retrieval, and captioning. In *arXiv 2410.07093*, 2024.

[51] Han Liang, Jiacheng Bao, Ruichi Zhang, Sihan Ren, Yuecheng Xu, Sibei Yang, Xin Chen, Jingyi Yu, and Lan Xu. Omg: Towards open-vocabulary motion generation via mixture of controllers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 482–493, 2024.

[52] 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*, pages 1–21, 2024.

[53] 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*, pages 1–21, 2024.

[54] Ting-Hsuan Liao, Yi Zhou, Yu Shen, Chun-Hao Paul Huang, Saayan Mitra, Jia-Bin Huang, and Uttaran Bhattacharya. Shape my moves: Text-driven shape-aware synthesis of human motions. *arXiv preprint arXiv:2504.03639*, 2025.

[55] Angela S. Lin, Lemeng Wu, Rodolfo Corona, Kevin W. H. Tai, Qi-Xing Huang, and Raymond J. Mooney. Generating animated videos of human activities from natural language descriptions. *arXiv preprint*, 2018.

[56] Jing Lin, Ailing Zeng, Shunlin Lu, Yuanhao Cai, Ruimao Zhang, Haoqian Wang, and Lei Zhang. Motion-x: A large-scale 3d expressive whole-body human motion dataset. *Advances in Neural Information Processing Systems*, 2023.

[57] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

[58] Xinpeng Liu, Haowen Hou, Yanchao Yang, Yong-Lu Li, and Cewu Lu. Revisit human-scene interaction via space occupancy. In *European Conference on Computer Vision*, pages 1–19. Springer, 2025.

[59] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. Smpl: a skinned multi-person linear model. *ACM Trans. Graph.*, 34(6), October 2015.

[60] Yuke Lou, Yiming Wang, Zhen Wu, Rui Zhao, Wenjia Wang, Mingyi Shi, and Taku Komura. Zero-shot human-object interaction synthesis with multimodal priors. *arXiv preprint arXiv:2503.20118*, 2025.

[61] Yunhong Lou, Linchao Zhu, Yaxiong Wang, Xiaohan Wang, and Yi Yang. Diversemotion: Towards diverse human motion generation via discrete diffusion. *arXiv preprint arXiv:2309.01372*, 2023.

[62] Shunlin Lu, Jingbo Wang, Zeyu Lu, Ling-Hao Chen, Wenxun Dai, Junting Dong, Zhiyang Dou, Bo Dai, and Ruimao Zhang. Scamo: Exploring the scaling law in autoregressive motion generation model. *arXiv preprint arXiv:2412.14559*, 2024.

[63] Shunlin Lu, Jingbo Wang, Zeyu Lu, Ling-Hao Chen, Wenxun Dai, Junting Dong, Zhiyang Dou, Bo Dai, and Ruimao Zhang. Scamo: Exploring the scaling law in autoregressive motion generation model. *arXiv preprint arXiv:2412.14559*, 2024.

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

[65] Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, and Michael J Black. Amass: Archive of motion capture as surface shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5442–5451, 2019.

[66] Zichong Meng, Yiming Xie, Xiaogang Peng, Zeyu Han, and Huaizu Jiang. Rethinking diffusion for text-driven human motion generation. *arXiv preprint arXiv:2411.16575*, 2024.

[67] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[68] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

[69] Xiaogang Peng, Yiming Xie, Zizhao Wu, Varun Jampani, Deqing Sun, and Huaizu Jiang. Hoi-diff: Text-driven synthesis of 3d human-object interactions using diffusion models. *arXiv preprint arXiv:2312.06553*, 2023.

[70] Mathis Petrovich, Michael J Black, and Gül Varol. Temos: Generating diverse human motions from textual descriptions. In *ECCV*, 2022.

[71] Mathis Petrovich, Michael J. Black, and Gül Varol. TMR: Text-to-motion retrieval using contrastive 3D human motion synthesis. In *ICCV*, 2023.

[72] Mathis Petrovich, Or Litany, Umar Iqbal, Michael J Black, Gul Varol, Xue Bin Peng, and Davis Rempe. Multi-track timeline control for text-driven 3d human motion generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1911–1921, 2024.

[73] Huaijin Pi, Ruoxi Guo, Zehong Shen, Qing Shuai, Zechen Hu, Zhumei Wang, Yajiao Dong, Ruizhen Hu, Taku Komura, Sida Peng, et al. Motion-2-to-3: Leveraging 2d motion data to boost 3d motion generation. *arXiv preprint arXiv:2412.13111*, 2024.

[74] Huaijin Pi, Sida Peng, Minghui Yang, Xiaowei Zhou, and Hujun Bao. Hierarchical generation of human-object interactions with diffusion probabilistic models. In *ICCV*, 2023.

[75] Ekkasit Pinyoanuntapong, Muhammad Usama Saleem, Korrawe Karunratanakul, Pu Wang, Hongfei Xue, Chen Chen, Chuan Guo, Junli Cao, Jian Ren, and Sergey Tulyakov. Controlmm: Controllable masked motion generation. *arXiv preprint arXiv:2410.10780*, 2024.

[76] Ekkasit Pinyoanuntapong, Muhammad Usama Saleem, Pu Wang, Minwoo Lee, Srijan Das, and Chen Chen. Bamm: Bidirectional autoregressive motion model. *arXiv preprint arXiv:2403.19435*, 2024.

[77] Ekkasit Pinyoanuntapong, Pu Wang, Minwoo Lee, and Chen Chen. Mmm: Generative masked motion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1546–1555, 2024.

[78] Matthias Plappert, Christian Mandery, and Tamim Asfour. The kit motion-language dataset. *Big data*, 4(4):236–252, 2016.

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

[80] Davis Rempe, Zhengyi Luo, Xue Bin Peng, Ye Yuan, Kris Kitani, Karsten Kreis, Sanja Fidler, and Or Litany. Trace and pace: Controllable pedestrian animation via guided trajectory diffusion. In *CVPR*, 2023.

[81] Pablo Ruiz-Ponce, German Barquero, Cristina Palmero, Sergio Escalera, and José García-Rodríguez. Mixermdm: Learnable composition of human motion diffusion models. *arXiv preprint arXiv:2504.01019*, 2025.

[82] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning internal representations by error propagation. In *Sematic Scholar*, 1986.

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

[84] Xu Shi, Chuanchen Luo, Junran Peng, Hongwen Zhang, and Yunlian Sun. Generating fine-grained human motions using chatgpt-refined descriptions. *arXiv preprint arXiv:2312.02772*, 2023.

[85] Yi Shi, Jingbo Wang, Xuekun Jiang, Bingkun Lin, Bo Dai, and Xue Bin Peng. Interactive character control with auto-regressive motion diffusion models. *ACM Transactions on Graphics (TOG)*, 43(4):1–14, 2024.

[86] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[87] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[88] Shanlin Sun, Gabriel De Araujo, Jiaqi Xu, Shenghan Zhou, Hanwen Zhang, Ziheng Huang, Chenyu You, and Xiaohui Xie. Coma: Compositional human motion generation with multi-modal agents. *arXiv preprint arXiv:2412.07320*, 2024.

[89] 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.

[90] Guy Tevet, Sigal Raab, Setareh Cohan, Daniele Reda, Zhengyi Luo, Xue Bin Peng, Amit H Bermano, and Michiel van de Panne. Closd: Closing the loop between simulation and diffusion for multi-task character control. *arXiv preprint arXiv:2410.03441*, 2024.

[91] Guy Tevet, Sigal Raab, Brian Gordon, Yoni Shafir, Daniel Cohen-or, and Amit Haim Bermano. Human motion diffusion model. In *The Eleventh International Conference on Learning Representations*, 2023.

[92] Shashank Tripathi, Omid Taheri, Christoph Lassner, Michael Black, Daniel Holden, and Carsten Stoll. Humos: Human motion model conditioned on body shape. In *European Conference on Computer Vision*, pages 133–152. Springer, 2025.

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

[94] Weilin Wan, Zhiyang Dou, Taku Komura, Wenping Wang, Dinesh Jayaraman, and Lingjie Liu. Tlcontrol: Trajectory and language control for human motion synthesis. *arXiv preprint arXiv:2311.17135*, 2023.

[95] Weilin Wan, Yiming Huang, Shutong Wu, Taku Komura, Wenping Wang, Dinesh Jayaraman, and Lingjie Liu. Diffusionphase: Motion diffusion in frequency domain. *arXiv preprint arXiv:2312.04036*, 2023.

[96] Jingbo Wang, Yu Rong, Jingyuan Liu, Sijie Yan, Dahua Lin, and Bo Dai. Towards diverse and natural scene-aware 3d human motion synthesis. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, page 20428–20437. IEEE, June 2022.

[97] Wenjia Wang, Liang Pan, Zhiyang Dou, Zhouyingcheng Liao, Yuke Lou, Lei Yang, Jingbo Wang, and Taku Komura. Sims: Simulating human-scene interactions with real world script planning. *arXiv preprint arXiv:2411.19921*, 2024.

[98] Xinghan Wang, Zixi Kang, and Yadong Mu. Text-controlled motion mamba: Text-instructed temporal grounding of human motion. *arXiv preprint arXiv:2404.11375*, 2024.

[99] Zhenzhi Wang, Jingbo Wang, Dahua Lin, and Bo Dai. Intercontrol: Generate human motion interactions by controlling every joint. *arXiv preprint arXiv:2311.15864*, 2023.

[100] Qianyang Wu, Ye Shi, Xiaoshui Huang, Jingyi Yu, Lan Xu, and Jingya Wang. Thor: Text to human-object interaction diffusion via relation intervention. *arXiv preprint arXiv:2403.11208*, 2024.

[101] Lixing Xiao, Shunlin Lu, Huaijin Pi, Ke Fan, Liang Pan, Yueer Zhou, Ziyong Feng, Xiaowei Zhou, Sida Peng, and Jingbo Wang. Motionstreamer: Streaming motion generation via diffusion-based autoregressive model in causal latent space. *arXiv preprint arXiv:2503.15451*, 2025.

[102] Yiming Xie, Varun Jampani, Lei Zhong, Deqing Sun, and Huaizu Jiang. Omnicontrol: Control any joint at any time for human motion generation. In *The Twelfth International Conference on Learning Representations*, 2024.

[103] Liang Xu, Shaoyang Hua, Zili Lin, Yifan Liu, Feipeng Ma, Yichao Yan, Xin Jin, Xiaokang Yang, and Wenjun Zeng. Motionbank: A large-scale video motion benchmark with disentangled rule-based annotations. *arXiv preprint arXiv:2410.13790*, 2024.

[104] 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.

[105] Sirui Xu, Zhengyuan Li, Yu-Xiong Wang, and Liang-Yan Gui. Interdiff: Generating 3d human-object interactions with physics-informed diffusion. In *ICCV*, 2023.

[106] Sirui Xu, Hung Yu Ling, Yu-Xiong Wang, and Liang-Yan Gui. Intermimic: Towards universal whole-body control for physics-based human-object interactions. *arXiv preprint arXiv:2502.20390*, 2025.

[107] Sirui Xu, Ziyin Wang, Yu-Xiong Wang, and Liang-Yan Gui. Interdreamer: Zero-shot text to 3d dynamic human-object interaction. *arXiv preprint arXiv:2403.19652*, 2024.

[108] Mengqing Xue, Yifei Liu, Ling Guo, Shaoli Huang, and Changxing Ding. Guiding human-object interactions with rich geometry and relations. *arXiv preprint arXiv:2503.20172*, 2025.

[109] Tatsuro Yamada, Hiroyuki Matsunaga, and Tetsuya Ogata. Paired recurrent autoencoders for bidirectional translation between robot actions and linguistic descriptions. *IEEE Robotics and Automation Letters*, 3(4):3441–3448, 2018.

[110] Sheng Yan, Yang Liu, Haoqiang Wang, Xin Du, Mengyuan Liu, and Hong Liu. Cross-modal retrieval for motion and text via droptriple loss. In *Proceedings of the 5th ACM International Conference on Multimedia in Asia*, pages 1–7, 2023.

[111] Payam Jome Yazdian, Eric Liu, Rachel Lagasse, Hamid Mohammadi, Li Cheng, and Angelica Lim. Motionscript: Natural language descriptions for expressive 3d human motions. *arXiv preprint arXiv:2312.12634*, 2023.

[112] Hongwei Yi, Justus Thies, Michael J Black, Xue Bin Peng, and Davis Rempe. Generating human interaction motions in scenes with text control. In *European Conference on Computer Vision*, pages 246–263. Springer, 2025.

[113] Heng Yu, Juze Zhang, Changan Chen, Tiange Xiang, Yusu Fang, Juan Carlos Niebles, and Ehsan Adeli. Socialgen: Modeling multi-human social interaction with language models. *arXiv preprint arXiv:2503.22906*, 2025.

[114] Weihao Yuan, Weichao Shen, Yisheng He, Yuan Dong, Xiaodong Gu, Zilong Dong, Liefeng Bo, and Qixing Huang. Mogents: Motion generation based on spatial-temporal joint modeling. *arXiv preprint arXiv:2409.17686*, 2024.

[115] Ye Yuan, Jiaming Song, Umar Iqbal, Arash Vahdat, and Jan Kautz. Physdiff: Physics-guided human motion diffusion model. In *ICCV*, 2023.

[116] Jianrong Zhang, Hehe Fan, and Yi Yang. Energymogen: Compositional human motion generation with energy-based diffusion model in latent space. *arXiv preprint arXiv:2412.14706*, 2024.

[117] Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. T2m-gpt: Generating human motion from textual descriptions with discrete representations. *arXiv preprint arXiv:2301.06052*, 2023.

[118] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3836–3847, 2023.

[119] Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. Motiondiffuse: Text-driven human motion generation with diffusion model. *arXiv preprint arXiv:2208.15001*, 2022.

[120] Mingyuan Zhang, Xinying Guo, Liang Pan, Zhongang Cai, Fangzhou Hong, Huirong Li, Lei Yang, and Ziwei Liu. Remodiffuse: Retrieval-augmented motion diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 364–373, 2023.

[121] Yaqi Zhang, Di Huang, Bin Liu, Shixiang Tang, Yan Lu, Lu Chen, Lei Bai, Qi Chu, Nenghai Yu, and Wanli Ouyang. Motiongpt: Finetuned llms are general-purpose motion generators. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.

[122] Zeyu Zhang, Akide Liu, Ian Reid, Richard Hartley, Bohan Zhuang, and Hao Tang. Motion mamba: Efficient and long sequence motion generation. In *European Conference on Computer Vision*, pages 265–282. Springer, 2024.

[123] Zeyu Zhang, Yiran Wang, Wei Mao, Danning Li, Rui Zhao, Biao Wu, Zirui Song, Bohan Zhuang, Ian Reid, and Richard Hartley. Motion anything: Any to motion generation. *arXiv preprint arXiv:2503.06955*, 2025.

[124] Zihan Zhang, Richard Liu, Rana Hanocka, and Kfir Aberman. Tedi: Temporally-entangled diffusion for long-term motion synthesis. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024.

[125] Kaifeng Zhao, Gen Li, and Siyu Tang. Dart: A diffusion-based autoregressive motion model for real-time text-driven motion control. *arXiv preprint arXiv:2410.05260*, 2024.

[126] Kaifeng Zhao, Yan Zhang, Shaofei Wang, Thabo Beeler, and Siyu Tang. Synthesizing diverse human motions in 3d indoor scenes. In *International conference on computer vision (ICCV)*, 2023.

[127] Chongyang Zhong, Lei Hu, Zihao Zhang, and Shihong Xia. Attt2m: Text-driven human motion generation with multi-perspective attention mechanism. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 509–519, 2023.

[128] Lei Zhong, Yiming Xie, Varun Jampani, Deqing Sun, and Huaizu Jiang. Smoodi: Stylized motion diffusion model. In *European Conference on Computer Vision*, pages 405–421. Springer, 2025.

[129] Wenyang Zhou, Zhiyang Dou, Zeyu Cao, Zhouyingcheng Liao, Jingbo Wang, Wenjia Wang, Yuan Liu, Taku Komura, Wenping Wang, and Lingjie Liu. Emdm: Efficient motion diffusion model for fast and high-quality motion generation. In *European Conference on Computer Vision*, pages 18–38. Springer, 2025.

[130] Yi Zhou, Chenglei Wu, Zimo Li, Chen Cao, Yuting Ye, Jason Saragih, Hao Li, and Yaser Sheikh. Fully convolutional mesh autoencoder using efficient spatially varying kernels. *Advances in neural information processing systems*, 33:9251–9262, 2020.

[131] Wenjie Zhuo, Fan Ma, and Hehe Fan. Infinidreamer: Arbitrarily long human motion generation via segment score distillation. *arXiv preprint arXiv:2411.18303*, 2024.

[132] Qiran Zou, Shangyuan Yuan, Shian Du, Yu Wang, Chang Liu, Yi Xu, Jie Chen, and Xiangyang Ji. Parco: Part-coordinating text-to-motion synthesis. In *European Conference on Computer Vision*, pages 126–143. Springer, 2025.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our abstract and introduction clearly reflected the paper's contribution and scope.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We include limitation of our work in Appendix.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: Our paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: This paper fully discloses all the information needed to reproduce experimental results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The datasets we used are open-source, and we will include open-sourced access to code in the camera-ready version of the paper.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: In Section 4 and the Appendix, we disclose all details for training and testing necessary to understand the results

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We report average results of multiple runs in our experimental section. Our paper does not report error bars.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide experiments compute resources in Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Research is conducted in the paper conforms with NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: Our paper is not highly related to societal impacts.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Our paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: CC-BY 4.0. And we referenced the works that we used to implement our code.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: Documentation of new assets is not applicable in our paper.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: Core method development in this research does not involve LLMs as any important, original, or non-standard components in our paper.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.