FlashMo: Geometric Interpolants and Frequency-Aware Sparsity for Scalable Efficient Motion Generation

Zeyu Zhang[1*] **Yiran Wang**[1*] **Danning Li**[2*] **Dong Gong**[3] **Ian Reid**[2] **Richard Hartley**[1]

[1]ANU [2]MBZUAI [3]UNSW

https://steve-zeyu-zhang.github.io/FlashMo

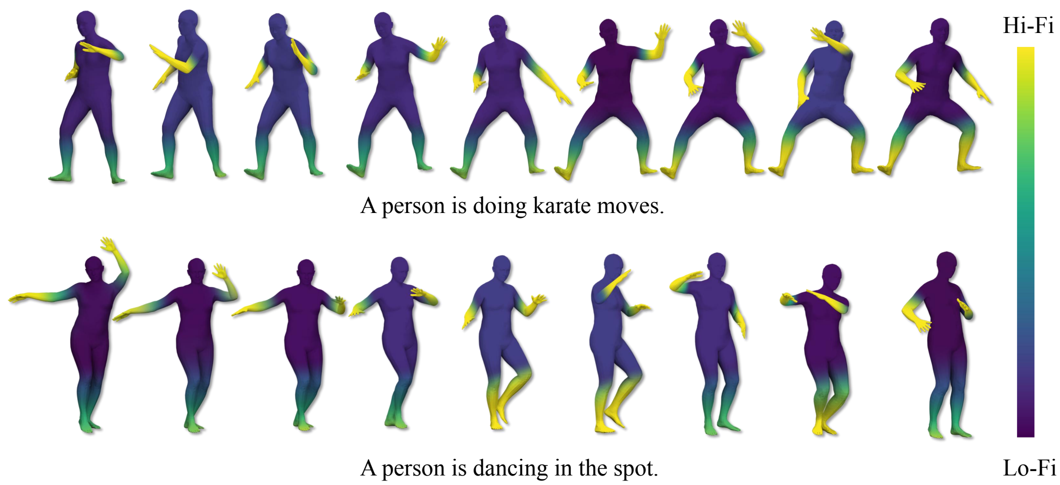


Figure 1: **Frequency of human motion.** The diagram illustrates that dynamic motions exhibit higher frequencies, while static motions correspond to lower frequencies. This observation provides key insights for the frequency-aware sparsification design of our FlashMo.

Abstract

Diffusion models have recently advanced 3D human motion generation by producing smoother and more realistic sequences from natural language. However, existing approaches face two major challenges: high computational cost during training and inference, and limited scalability due to reliance on U-Net inductive bias. To address these challenges, we propose **FlashMo**, a frequency-aware sparse motion diffusion model that prunes low-frequency tokens to enhance efficiency without custom kernel design. We further introduce *MotionSiT*, a scalable diffusion transformer based on a joint-temporal factorized interpolant with Lie group geodesics over $\mathrm{SO}(3)$ manifolds, enabling principled generation of joint rotations. Extensive experiments on the large-scale MotionHub V2 dataset and standard benchmarks including HumanML3D and KIT-ML demonstrate that our method significantly outperforms previous approaches in motion quality, efficiency, and scalability. Compared to the state-of-the-art 1-step distillation baseline, FlashMo reduces **12.9%** inference time and FID by **34.1%**.

1 Introduction

The conditional generation of 3D motions has recently garnered significant attention due to its broad applicability across various domains, including robot manipulation [65], urban planning [11], virtual

*Equal contribution.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

[83] and augmented reality [87], game development [21, 101], and video creation [5]. Notably, recent progress in text-to-motion generation, particularly with autoregressive [100, 60, 59, 30, 85, 39, 92] and diffusion models [70, 93, 7, 94, 44, 73], has enabled the synthesis of natural human motion from natural language. While VQ-VAE-based autoregressive methods achieve outstanding quantitative results, they generate less natural motion with jitters due to frame-wise noise arising from directly decoding discrete tokens, and fine-grained motion details are sometimes lost during token discretization [13]. In contrast, motion diffusion models generate smoother and more realistic human motion, showing a promising trend in human motion generation [73, 74, 95]. However, despite their strengths, diffusion-based approaches still face two significant challenges, collectively limiting their applicability in real-world scenarios.

(1) **Efficiency.** Motion diffusion suffers from high computational cost, as well as long training and inference times [41]. Existing efficient methods such as step distillation [17, 16, 41], step reduction [104], and linear models [99, 89] either require additional training of teacher models or involve complex scanning mechanisms, resulting in overengineering and low efficiency.

(2) **Scalability.** Recent works in image [24, 54, 48, 71] and video generation [82, 72] indicate that the U-Net [64] inductive bias is not critical for diffusion [24], while plain transformer denoisers show promising scalability. However, existing motion diffusion models are mostly restricted conventional U-Net architectures on smaller datasets [31, 61], resulting in limited and unexplored scalability.

To tackle the first challenge without additional engineering, we leverage intrinsic data characteristics. We observe that dynamic motion, which is more important [97, 100], exhibits higher frequency compared to static motion in generation process, as shown in Figure 1. Hence, we design a frequency-aware sparsification mechanism that dynamically prunes tokens corresponding to low frequency motion, enhancing efficiency while preserving high frequency motion at the attention head level. This structured sparsification allows seamless adaptation to hardware-efficient exact attention [19, 18, 67] without the need to customize kernels, which further enhances the efficiency. Furthermore, the unified training and inference sparsification strategy resolves the sparsity granularity gap, enabling a **2.25×** speedup over full attention while maintaining comparable performance.

To tackle the second challenge, we design MotionSiT, along with FlashMo, tailored for motion diffusion, which differs from the standard SiT that interpolates all dimensions together in Euclidean space. Instead, we perform temporal-spatial factorized interpolant with Lie group geodesics on the manifolds of joint rotations. This benefits our diffusion on the $\mathfrak{so}(3)$ representation of joint rotations [26, 37] and makes it more feasible to scale training on larger datasets [50]. Comprehensive analyses also validate the effectiveness of each component and their synergistic integration.

Furthermore, to demonstrate the efficiency and scalability of FlashMo, we pretrain on the large-scale open-source dataset MotionHub V2 [50] and evaluate our method on standard text-to-motion benchmarks, including HumanML3D [31] and KIT-ML [61]. The results in Figure 5 show our method significantly outperforms prior approaches in terms of motion quality, efficiency, and scalability.

The contributions of our paper can be summarized as follows:

- We present FlashMo, a frequency-aware sparse motion diffusion that prunes low-frequency tokens to improve efficiency without kernel customization. Moreover, our trainable sparsification strategy eliminates the sparsity granularity gap between training and inference, achieving a **2.25×** speedup over full attention without sacrificing performance.
- We introduce MotionSiT, a diffusion transformer with factorized Lie group interpolant that enables scalable motion diffusion on $\mathrm{SO}(3)$ manifolds of joint rotations.
- To evaluate our method's efficiency and scalability, we conduct comprehensive experiments on the pretraining dataset MotionHub V2 [50] and downstream benchmarks including HumanML3D [31] and KIT-ML [61]. The results show that our model outperforms previous methods and achieves outstanding balance between efficiency and performance. Compared to the state-of-the-art 1-step distillation baseline (MotionPCM), FlashMo reduces inference time by 12.9% and FID by 34.1%.

2 Related Work

Motion diffusion. Diffusion models have been widely adopted for human motion generation due to their strong ability to synthesize realistic and diverse sequences [70, 93, 7, 94, 44, 73, 95, 42, 80,

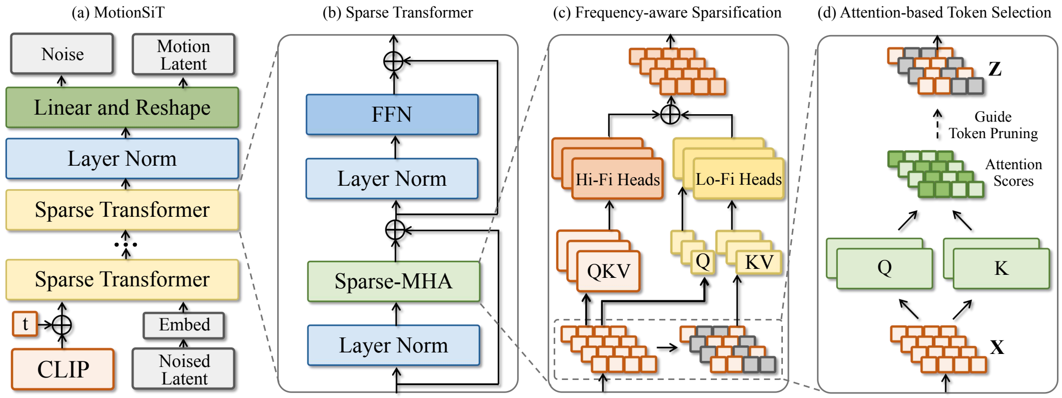


Figure 2: **FlashMo architecture.** FlashMo leverages frequency-aware sparsification and a MotionSiT backbone to efficiently and scalably generate motion from noised latent inputs.

12, 38, 8, 74, 43, 90, 76, 33]. However, the time-consuming problem has motivated recent efforts of acceleration. Existing approaches include step distillation [17, 16, 41], GAN-based discriminators [104], and lightweight linear models [99, 98, 97, 89]. However, these methods often either rely on multi-stage training with teacher models, require additional discriminators, or trade-off performance for efficiency. Moreover, recent motion diffusion methods have explored motion frequency for designing consistency losses [6] or decomposing hidden states in linear models [47]. However, they have not investigated its role at the attention-head level or its potential for further improving efficiency [57]. These methods have also been applied to downstream tasks such as long motion generation [66, 2] and motion editing [79, 62, 92, 39, 95, 14].

Sparse attention. Sparse attention manifests in two forms: token sparsification and activation sparsification. Depending on the granularity of sparsity, token sparsification can be categorized into unstructured, semi-structured, and structured schemes. Unstructured sparse attention [75, 55] applies token sparsification without a fixed attention pattern, resulting in hardware unfriendly and challenges in kernel design [22]. Semi-structured sparse attention performs token sparsification with: (1) pyramid [102], local [86, 3, 78, 88, 105, 51], or sliding windows [96, 34, 53, 84], (2) predefined attention mask [40, 77, 56], (3) introducing $N : M$ sparsity into attention weights [25, 23, 10], according to the observed attention patterns, (4) block-wise sparse attention with a pooling operation [81, 45, 27, 84]. However, it often requires custom computational kernels tailored to each sparsity pattern or dedicated hardware support. Structured sparse attention [4, 63, 1, 52, 103, 35] prunes tokens before the attention computation, enabling acceleration without the need for custom kernels. However, due to its coarse granularity, it sometimes underperforms and is therefore typically accompanied by distillation [49, 46], which introduces additional computation to pretrain the full attention teacher model. For activation sparsification, it reduce computational cost by zeroing insignificant activations [91, 9, 68, 29].

3 Method

3.1 Overview

Our FlashMo introduces innovations in both the formulation and architecture of motion diffusion, as shown in Figure 2. We propose a *geometric factorized interpolant* that respects the temporal-spatial structure of motion and preserves joint rotation consistency via manifold-based interpolation. This allows us to achieve superior performance and improved scalability. To improve efficiency, we design a *frequency-aware sparse attention* layer that prunes low-frequency tokens while preserving high-frequency ones at the head level. Token selection is adaptively guided by the attention distribution, enabling unified training and inference without the sparsity granularity mismatch. Our sparsification integrates seamlessly with hardware-efficient exact attention [19, 18, 67] without kernel customization, achieving high efficiency without sacrificing performance.

3.2 MotionSiT

MotionSiT is a latent generative model. Given a motion representation $\mathbf{M} \in \mathbb{M}^{L\times D}$ where L denotes the motion length and D the spatial feature dimension encoded from rotations, we apply a VAE to compress $\mathbf{M}$ along the temporal axis. The encoder maps $\mathbf{M}$ to a latent representation $\mathbf{X} \in \mathbb{R}^{T\times S}$, where $T = L/r$ is the reduced temporal resolution with compression ratio r, and S is the latent dimension in spatial direction. Compared to standard SiT as in *Appendix A.1*, which targets image generation, motion generation differs due to its temporal-spatial structure and the manifold geometry of motion representations. According to these characteristics, our MotionSiT introduces following geometric interpolant tailored for motion.

Temporal-spatial factorized interpolant. The original SiT assumes each input sample is a flat vector and defines a simple linear stochastic interpolant between a noise sample $\epsilon \sim \mathcal{N}(0, I)$ and the data sample x^* via:

$$x_t = \alpha(t)x^* + \sigma(t)\epsilon.$$

While effective for image data, this formulation overlooks the heterogeneous nature of motion representation, where temporal and spatial dimensions carry distinct physical meanings and roles in motion representation.

Hence, we introduce a *temporal-spatial factorized interpolant (FI)*, which applies independent noise schedules along the temporal and spatial dimensions. Specifically, we define:

$$x_t = \alpha_T(t) \odot \alpha_S(t) \odot x^* + \sigma(t)\epsilon,$$

where $\alpha_T(t) \in \mathbb{R}^T$ and $\alpha_S(t) \in \mathbb{R}^S$ are scalar schedules for temporal and spatial dimensions, respectively. The element-wise product $\odot$ is broadcasted to modulate x^* at each timestep and spatial location individually. This factorization allows flexible noise control across time and joints, reflecting motion's structural characteristics.

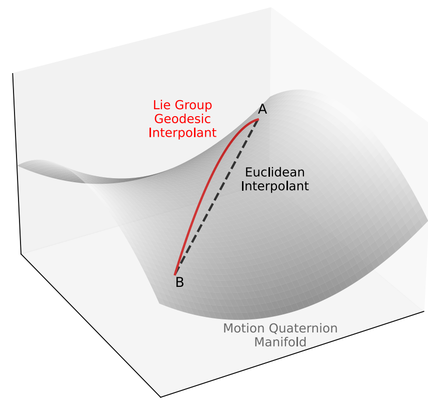


Figure 3: **Euclidean interpolant vs. Lie group geodesics interpolant.**

Lie group geodesics interpolant. Another limitation of interpolation in vanilla SiT lies in its assumption of Euclidean geometry, as shown in Figure 3 However, for motion representation, joint orientations are commonly expressed as rotation matrices in $\mathrm{SO}(3)$ [26, 37]. Applying such interpolation directly to the motion manifold leads to geometric inconsistency in joint rotations.

Hence, we introduce the *Lie group geodesic interpolant (LI)* to respect the underlying manifold structure through Lie group geodesics. Given a target motion sample $x^* \in \mathcal{G}$, where $\mathcal{G}$ denotes a Lie group, we first map x^* to its tangent space $\mathfrak{g}$ using the logarithmic map $\log : \mathcal{G} \to \mathfrak{g}$. We then interpolate within the tangent space before mapping back to the manifold with the exponential map:

$$x_t = \mathrm{Exp}\left(\alpha(t) \cdot \log(x^*) + \sigma(t) \cdot \epsilon\right),$$

where ϵ is Gaussian noise in the Lie algebra $\mathfrak{g}$. This approach ensures that the interpolated sample x_t always lies on the manifold and that noise is injected in a way that respects the local geometry of the motion space.

Geometric factorized interpolant. Building on the temporal-spatial factorization introduced earlier, we now incorporate the Lie group geodesic to define the *geometric factorized interpolant (GFI)* in MotionSiT. Specifically, for motion representation lies on a manifold such as $\mathrm{SO}(3)$, we perform geodesic interpolant on both temporal and spatial dimension:

$$x_t^{\tau,s} = \mathrm{Exp}\left(\alpha_T(t)_\tau \cdot \alpha_S(t)_s \cdot \log(x^{*\tau,s}) + \sigma(t) \cdot \epsilon^{\tau,s}\right),$$

where $x^{*\tau,s}$ denotes the motion state at time τ and spatial location s, and $\epsilon^{\tau,s}$ is Gaussian noise in the corresponding tangent space. This formulation combines the benefits of temporal-spatial control while preserves the geometric consistency of joint rotations during interpolation.

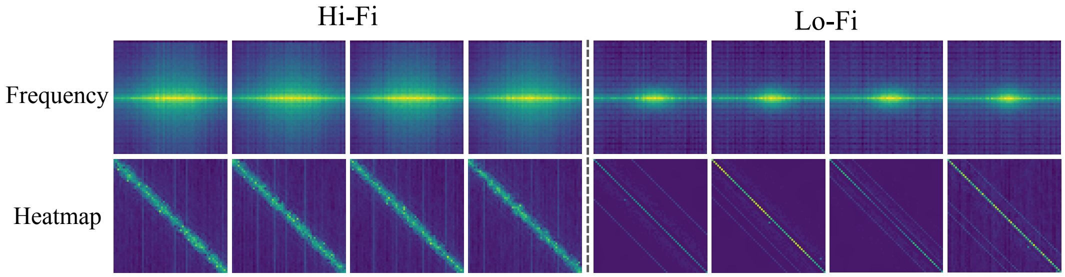


Figure 4: **Frequency magnitude vs. attention heatmap of attention heads.** The frequency magnitude is computed with the Fast Fourier Transform (FFT) and averaged over 100 latent motion features. Brighter colors indicate higher magnitudes. Pixels closer to the center represent lower frequencies. Both maps are visualized from the last layer of MotionSiT.

3.3 Frequency-Aware Sparsification

Through observation of the frequency of attention heads in the upper part of Figure 4, the heads can be categorized into Hi-Fi and Lo-Fi groups. Moreover, the lower part of Figure 4 shows that Hi-Fi and Lo-Fi heads exhibit clearly distinct patterns in the attention heatmap, providing strong empirical evidence for designing dynamic token sparsification.

Frequency-aware head partition. Given the number of head N_h, we first partition the N_h attention heads based on their frequency with an ratio $\beta \in [0, 1]$, where the first βN_h heads are low-frequency (Lo-Fi) heads $\mathcal{H}_{\text{Lo}}$ and the remaining $(1-\beta)N_h$ heads are high-frequency (Hi-Fi) heads $\mathcal{H}_{\text{Hi}}$.

$$\mathcal{H}_{\text{Lo}} \subset \{1, \ldots, N_h\}, \text{with } |\mathcal{H}_{\text{Lo}}| = \beta N_h, \ \text{ and } \mathcal{H}_{\text{Hi}} = \{1, \ldots, N_h\} \setminus \mathcal{H}_{\text{Lo}}$$

Full attention for Hi-Fi heads. Given a motion latent representation $\mathbf{X} \in \mathbb{R}^{T \times S}$, for each attention head $h \in \mathcal{H}_{\text{Hi}}$, we define projection matrices $\mathbf{W}_q^{(h)}, \mathbf{W}_k^{(h)}, \mathbf{W}_v^{(h)} \in \mathbb{R}^{S \times d_k}$, where d_k is the hidden dimensions for a head. We project $\mathbf{X}$ into $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ for each head:

$$\mathbf{Q}[\mathbf{X}]^{(h)} = \mathbf{X}\mathbf{W}_q^{(h)}, \ \ \mathbf{K}[\mathbf{X}]^{(h)} = \mathbf{X}\mathbf{W}_k^{(h)}, \ \ \mathbf{V}[\mathbf{X}]^{(h)} = \mathbf{X}\mathbf{W}_v^{(h)},$$

where $\mathbf{Q}[\mathbf{X}]^{(h)}, \mathbf{K}[\mathbf{X}]^{(h)}, \mathbf{V}[\mathbf{X}]^{(h)} \in \mathbb{R}^{T \times d_k}$. We then calculate full attention for each Hi-Fi head:

$$\text{Attention}(\mathbf{Q}[\mathbf{X}]^{(h)}, \mathbf{K}[\mathbf{X}]^{(h)}, \mathbf{V}[\mathbf{X}]^{(h)}), \ \ \text{where} \ \ h \in \mathcal{H}_{\text{Hi}}.$$

Sparse attention for Lo-Fi heads. We perform attention-guided sparsification on Lo-Fi heads, adaptively pruning less important low-frequency motion tokens while preserving key motions

Attention-based adaptive token selection. Previous works [100] have successfully incorporated attention-guidance for selecting key motion tokens. However, they are limited to attention masking and do not adapt the masking ratio across layers. In contrast, our sparsification is adaptive across layers based on their attention distribution.

Given a full attention score in each head $\mathbf{A} = \mathbf{Q}[\mathbf{X}]^{(h)}\mathbf{K}[\mathbf{X}]^{(h)^\top} \in \mathbb{R}^{T \times T}$, the cumulative attention score a_x for each token x is calculated by summing the corresponding column, and the normalized cumulative attention score $\tilde{a}_x$ by averaging over the non-zero entries in that column.

$$a_x = \sum_{c=1}^{T} \mathbf{A}_{c,x}, \ \ \tilde{a}_x = \frac{\sum_{c=1}^{T} \mathbf{A}_{c,x}}{\sum_{c=1}^{T} \mathbb{I}[\mathbf{A}_{c,x} \neq 0]}.$$

We then select the minimum number of tokens κ whose cumulative attention scores reach the threshold γ of the total attention mass:

$$\kappa = \min \left\{ \kappa \in \mathbb{Z} \ \middle| \ \sum_{i=1}^{\kappa} a_{\text{sorted}(i)} \geq \gamma \sum_{i=1}^{T} a_{\text{sorted}(i)} \right\},$$

where $a_{\text{sorted}(i)}$ denotes the i-th highest attention score, obtained by sorting cumulative attention scores a in descending order.

We then select the top κ tokens from $\mathbf{X} \in \mathbb{R}^{T \times S}$ according to their normalized cumulative attention scores $\tilde{a}_x$, and obtain the selected token matrix $\mathbf{Z} \in \mathbb{R}^{\kappa \times S}$, where $\kappa < T$. This allows κ to vary depending on the distribution of attention scores in each attention layer:

$$\mathcal{T}_\kappa = \text{Top-}\kappa(\tilde{a}_x), \quad \mathbf{Z} = \mathbf{X}[\mathcal{T}_\kappa, :], \quad \mathbf{Z} \in \mathbb{R}^{\kappa \times S}.$$

Sparse attention calculation. For each attention head $h \in \mathcal{H}_{\text{Lo}}$, we project $\mathbf{X}$ into $\mathbf{Q}$, and $\mathbf{Z}$ into $\mathbf{K}, \mathbf{V}$ for each head:

$$\mathbf{Q}[\mathbf{X}]^{(h)} = \mathbf{X}\mathbf{W}_q^{(h)}, \;\; \mathbf{K}[\mathbf{Z}]^{(h)} = \mathbf{Z}\mathbf{W}_k^{(h)}, \;\; \mathbf{V}[\mathbf{Z}]^{(h)} = \mathbf{Z}\mathbf{W}_v^{(h)},$$

where $\mathbf{Q}[\mathbf{X}]^{(h)} \in \mathbb{R}^{T \times d_k}$, and $\mathbf{K}[\mathbf{Z}]^{(h)}, \mathbf{V}[\mathbf{Z}]^{(h)} \in \mathbb{R}^{\kappa \times d_k}$. We then calculate sparse attention for each Lo-Fi head:

$$\text{Attention}(\mathbf{Q}[\mathbf{X}]^{(h)}, \mathbf{K}[\mathbf{Z}]^{(h)}, \mathbf{V}[\mathbf{Z}]^{(h)}), \;\; \text{where} \;\; h \in \mathcal{H}_{\text{Lo}}.$$

Sparse multi-head attention. Hence our frequency-aware sparse multi-head attention can be calculated as:

$$\text{Sparse-MHA}(\mathbf{X}) = \text{Concat}\left[\begin{cases} \text{Attention}(\mathbf{Q}[\mathbf{X}]^{(h)}, \mathbf{K}[\mathbf{X}]^{(h)}, \mathbf{V}[\mathbf{X}]^{(h)}), & h \in \mathcal{H}_{\text{Hi}} \\ \text{Attention}(\mathbf{Q}[\mathbf{X}]^{(h)}, \mathbf{K}[\mathbf{Z}]^{(h)}, \mathbf{V}[\mathbf{Z}]^{(h)}), & h \in \mathcal{H}_{\text{Lo}}, \end{cases}\right] \mathbf{W}_o$$

where $\mathbf{W}_o \in \mathbb{R}^{(N_h \cdot d_k) \times S}$ projects back to the original latent dimension.

4 Experiments

4.1 Datasets and Evaluation Metrics

Datasets. For pretraining, we utilize the recent large-scale open-source dataset MotionHub V2 [50], which contains 142,350 motion clips and 259,998 captions. For downstream evaluation, we conduct experiments on standard text-to-motion (T2M) datasets, including HumanML3D [31] and KIT-ML [61]. HumanML3D comprises 14,616 motion clips, each accompanied by three textual descriptions, resulting in a total of 44,970 captions. The KIT-ML dataset consists of 3,911 motions paired with 6,278 textual descriptions. For both datasets, we adopt the pose representation defined in T2M [31] to ensure consistency in motion representation across evaluations.

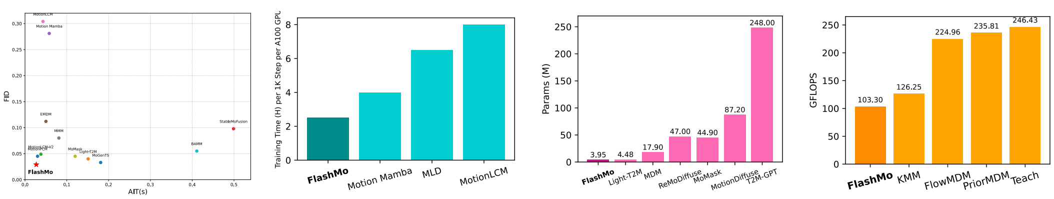
Evaluation metrics. We adopt standard evaluation metrics to assess different aspects of our experiments. We use FID and R-Precision to evaluate the realism and accuracy of generated motions, MultiModal Distance to measure motion-text alignment, and a diversity metric to quantify variation in motion features. Additionally, we employ the Multi-Modality (MModality) metric to assess diversity among motions generated from the same text description. Moreover, we calculate Average Inference Time (AIT) for showing efficiency.

4.2 Implementation Details

The encoder and decoder of the VAE consist of 4 layers with a compression rate $r = 4$. MotionSiT has a depth of 8, with a frequency ratio $\beta = 0.5$ and an attention threshold $\gamma = 0.95$. Both the VAE and MotionSiT use 4 attention heads with a latent dimension of 512. We employ a frozen text encoder from CLIP ViT-B/32. A constant learning rate of 1×10^{-4} is used, with a batch size of 256 and the AdamW optimizer. For fair comparison, each model is trained for 6K epochs during the VAE stage and 3K epochs during the diffusion stage. We adopt 1000 diffusion steps during training and 10 sampling steps during inference. All experiments are conducted on an Intel Xeon Platinum 8469C CPU at 2.60GHz, with a single NVIDIA H20 96G GPU and 32GB of RAM.

Table 1: Comparison of text-to-motion generation on HumanML3D [31] and KIT-ML [61] datasets. $\rightarrow$ indicates the closer to real data, the better. **Bold** and <u>underline</u> indicate best and second best results.

Method	Venue	AIT(s) $\downarrow$	R-Precision $\uparrow$			FID $\downarrow$	MM Dist $\downarrow$	Diversity $\rightarrow$	MModality $\uparrow$
			Top-1	Top-2	Top-3				
HumanML3D [31]									
Real	-	-	0.511$^{\pm.003}$	0.703$^{\pm.003}$	0.797$^{\pm.002}$	0.002$^{\pm.000}$	2.974$^{\pm.008}$	9.503$^{\pm.065}$	-
MMM [60]	CVPR 2024	0.081	0.504$^{\pm.003}$	0.696$^{\pm.003}$	0.794$^{\pm.002}$	0.080$^{\pm.003}$	2.998$^{\pm.007}$	9.411$^{\pm.058}$	1.164$^{\pm.041}$
MoMask [30]	CVPR 2024	0.120	0.521$^{\pm.002}$	0.713$^{\pm.002}$	0.807$^{\pm.002}$	0.045$^{\pm.002}$	2.958$^{\pm.008}$	-	1.241$^{\pm.040}$
BAMM [59]	ECCV 2024	0.411	0.525$^{\pm.002}$	0.720$^{\pm.003}$	0.814$^{\pm.003}$	0.055$^{\pm.002}$	2.919$^{\pm.008}$	9.717$^{\pm.089}$	1.687$^{\pm.051}$
MoGenTS [85]	NeurIPS 2024	0.181	0.529$^{\pm.003}$	0.719$^{\pm.002}$	0.812$^{\pm.002}$	0.033$^{\pm.001}$	2.867$^{\pm.006}$	9.570$^{\pm.077}$	-
MotionLCM [17] (1-step)	ECCV 2024	<u>0.030</u>	0.502$^{\pm.003}$	0.701$^{\pm.002}$	0.803$^{\pm.002}$	0.467$^{\pm.012}$	3.022$^{\pm.009}$	9.631$^{\pm.066}$	2.172$^{\pm.082}$
MotionLCM [17] (2-step)	ECCV 2024	0.035	0.505$^{\pm.003}$	0.705$^{\pm.002}$	0.805$^{\pm.002}$	0.368$^{\pm.011}$	2.986$^{\pm.008}$	9.640$^{\pm.052}$	2.187$^{\pm.094}$
MotionLCM [17] (4-step)	ECCV 2024	0.043	0.502$^{\pm.003}$	0.698$^{\pm.002}$	0.798$^{\pm.002}$	0.304$^{\pm.012}$	3.012$^{\pm.007}$	9.607$^{\pm.066}$	2.259$^{\pm.092}$
EMDM [104]	ECCV 2024	0.050	0.498$^{\pm.007}$	0.684$^{\pm.006}$	0.786$^{\pm.006}$	0.112$^{\pm.019}$	3.110$^{\pm.027}$	**9.551**$^{\pm.078}$	1.641$^{\pm.078}$
Motion Mamba [99]	ECCV 2024	0.058	0.502$^{\pm.003}$	0.693$^{\pm.002}$	0.792$^{\pm.002}$	0.281$^{\pm.009}$	3.060$^{\pm.058}$	9.871$^{\pm.084}$	2.294$^{\pm.058}$
StableMoFusion [38]	MM 2024	0.499	0.553$^{\pm.003}$	0.748$^{\pm.002}$	0.841$^{\pm.002}$	0.098$^{\pm.003}$	-	9.748$^{\pm.092}$	1.774$^{\pm.051}$
MotionLCM-V2 [16] (1-step)	Preprint 2024	0.031	0.546$^{\pm.003}$	0.743$^{\pm.002}$	0.837$^{\pm.002}$	0.072$^{\pm.003}$	2.767$^{\pm.007}$	9.577$^{\pm.070}$	1.858$^{\pm.056}$
MotionLCM-V2 [16] (2-step)	Preprint 2024	0.038	0.551$^{\pm.003}$	0.745$^{\pm.002}$	0.836$^{\pm.002}$	0.049$^{\pm.003}$	2.765$^{\pm.008}$	9.584$^{\pm.066}$	1.833$^{\pm.052}$
MotionLCM-V2 [16] (4-step)	Preprint 2024	0.050	0.553$^{\pm.003}$	0.746$^{\pm.002}$	0.837$^{\pm.002}$	0.056$^{\pm.003}$	2.773$^{\pm.009}$	9.598$^{\pm.067}$	1.758$^{\pm.056}$
Light-T2M [89]	AAAI 2025	0.151	0.511$^{\pm.003}$	0.699$^{\pm.002}$	0.795$^{\pm.002}$	0.040$^{\pm.002}$	3.002$^{\pm.008}$	-	1.670$^{\pm.061}$
MotionPCM [41] (1-step)	Preprint 2025	0.031	0.560$^{\pm.002}$	0.752$^{\pm.003}$	0.844$^{\pm.002}$	0.044$^{\pm.003}$	<u>2.711</u>$^{\pm.008}$	<u>9.559</u>$^{\pm.081}$	1.772$^{\pm.067}$
MotionPCM [41] (2-step)	Preprint 2025	0.036	0.555$^{\pm.002}$	0.749$^{\pm.002}$	0.839$^{\pm.002}$	0.033$^{\pm.002}$	2.739$^{\pm.007}$	9.618$^{\pm.088}$	1.760$^{\pm.068}$
MotionPCM [41] (4-step)	Preprint 2025	0.045	0.559$^{\pm.003}$	0.752$^{\pm.003}$	0.842$^{\pm.002}$	<u>0.030</u>$^{\pm.002}$	2.716$^{\pm.008}$	9.575$^{\pm.082}$	1.714$^{\pm.062}$
FlashMo (Ours)	-	**0.027**	<u>0.562</u>$^{\pm.004}$	<u>0.754</u>$^{\pm.005}$	<u>0.847</u>$^{\pm.005}$	0.041$^{\pm.002}$	<u>2.711</u>$^{\pm.006}$	9.614$^{\pm.056}$	<u>2.812</u>$^{\pm.046}$
FlashMo w/ pretrain (Ours)	-	**0.027**	**0.568**$^{\pm.005}$	**0.761**$^{\pm.002}$	**0.851**$^{\pm.003}$	**0.029**$^{\pm.002}$	**2.703**$^{\pm.005}$	9.601$^{\pm.073}$	**2.851**$^{\pm.069}$
KIT-ML [61]									
Real	-	-	0.424$^{\pm.005}$	0.649$^{\pm.006}$	0.779$^{\pm.006}$	0.031$^{\pm.004}$	2.788$^{\pm.012}$	11.08$^{\pm.097}$	-
MMM [60]	CVPR 2024	-	0.404$^{\pm.005}$	0.621$^{\pm.005}$	0.744$^{\pm.004}$	0.316$^{\pm.028}$	2.977$^{\pm.019}$	10.91$^{\pm.101}$	1.232$^{\pm.039}$
MoMask [30]	CVPR 2024	-	0.433$^{\pm.007}$	0.656$^{\pm.005}$	0.781$^{\pm.005}$	0.204$^{\pm.011}$	2.779$^{\pm.022}$	-	1.131$^{\pm.043}$
BAMM [59]	ECCV 2024	-	0.438$^{\pm.009}$	0.661$^{\pm.009}$	0.788$^{\pm.005}$	0.183$^{\pm.013}$	2.723$^{\pm.026}$	<u>11.01</u>$^{\pm.094}$	1.609$^{\pm.065}$
MoGenTS [85]	NeurIPS 2024	-	0.445$^{\pm.006}$	<u>0.671</u>$^{\pm.006}$	0.797$^{\pm.005}$	<u>0.143</u>$^{\pm.004}$	2.711$^{\pm.024}$	10.92$^{\pm.090}$	-
EMDM [104]	ECCV 2024	-	0.443$^{\pm.006}$	0.660$^{\pm.006}$	0.780$^{\pm.005}$	0.261$^{\pm.014}$	2.874$^{\pm.015}$	10.96$^{\pm.093}$	1.343$^{\pm.089}$
Motion Mamba [99]	ECCV 2024	-	0.419$^{\pm.006}$	0.645$^{\pm.005}$	0.765$^{\pm.006}$	0.307$^{\pm.041}$	3.021$^{\pm.025}$	**11.02**$^{\pm.098}$	1.678$^{\pm.064}$
StableMoFusion [38]	MM 2024	-	0.445$^{\pm.006}$	0.660$^{\pm.005}$	0.782$^{\pm.004}$	0.258$^{\pm.029}$	-	10.94$^{\pm.077}$	1.362$^{\pm.062}$
Light-T2M [89]	AAAI 2025	-	0.444$^{\pm.006}$	0.670$^{\pm.007}$	0.794$^{\pm.005}$	0.161$^{\pm.009}$	2.746$^{\pm.016}$	-	1.005$^{\pm.036}$
MotionPCM [41] (1-step)	Preprint 2025	-	0.433$^{\pm.007}$	0.654$^{\pm.007}$	0.781$^{\pm.008}$	0.355$^{\pm.011}$	2.820$^{\pm.022}$	10.78$^{\pm.078}$	1.337$^{\pm.047}$
MotionPCM [41] (2-step)	Preprint 2025	-	0.437$^{\pm.005}$	0.664$^{\pm.005}$	0.787$^{\pm.006}$	0.294$^{\pm.011}$	2.844$^{\pm.018}$	10.83$^{\pm.094}$	1.254$^{\pm.050}$
MotionPCM [41] (4-step)	Preprint 2025	-	0.443$^{\pm.005}$	0.664$^{\pm.004}$	0.789$^{\pm.005}$	0.336$^{\pm.013}$	2.881$^{\pm.023}$	10.76$^{\pm.096}$	1.258$^{\pm.056}$
FlashMo (Ours)	-	-	<u>0.449</u>$^{\pm.002}$	0.670$^{\pm.004}$	<u>0.799</u>$^{\pm.002}$	0.152$^{\pm.004}$	<u>2.709</u>$^{\pm.005}$	10.64$^{\pm.074}$	<u>3.287</u>$^{\pm.042}$
FlashMo w/ pretrain (Ours)	-	-	**0.453**$^{\pm.001}$	**0.679**$^{\pm.004}$	**0.807**$^{\pm.003}$	**0.132**$^{\pm.005}$	**2.701**$^{\pm.005}$	10.79$^{\pm.093}$	**3.591**$^{\pm.070}$



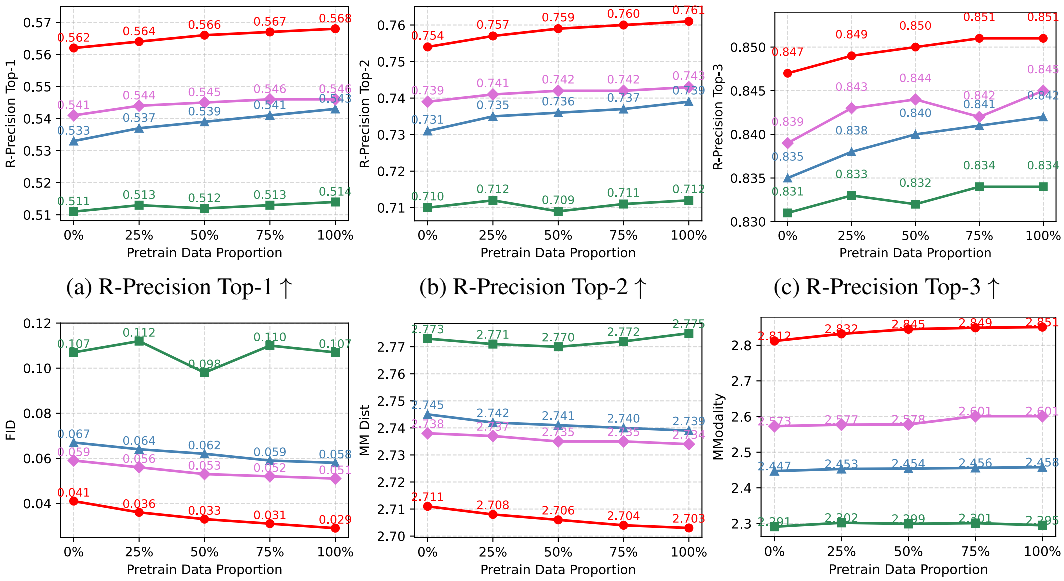
(a) FID vs. AIT(s) $\swarrow$ (b) Training Time $\downarrow$ (c) Parameters $\downarrow$ (d) GFLOPS $\downarrow$

Figure 5: **Efficiency comparison.** The figure demonstrates that FlashMo achieves the lowest inference time, training time, model size, and FLOPs while maintaining superior performance compared to other methods.

4.3 Comparative Study

We compare our method with recent efficient motion diffusion models and VQ-VAE-based models on both HumanML3D and KIT-ML. We include both our model trained from scratch and the version pretrained on MotionHub V2 [50]. And we follow T2M [31] and report the average over 20 runs with 95% confidence intervals. The results in Table 1 demonstrate that our method consistently outperforms other approaches across most quantitative performance metrics, achieving a 10% efficiency improvement compared to previous fastest 1-step distillation [17], without the need to train an additional teacher model. Please see the full comparison table in *Appendix D*.

Efficiency. We compare our method with other approaches in terms of Average Inference Time (AIT), training time, model parameters, and GFLOPs. For the inference setting, we follow [7] to calculate AIT, measured on the same Tesla V100 GPU. The results in Figure 5 and Table 1 demonstrate that our method not only achieves superior performance but also maintains the lowest inference time, training time, model size, and GFLOPs.



(d) FID ↓ (e) MM Dist ↓ (f) MModality ↑

Figure 6: **Scaling trend.** The figure demonstrates the scaling trends of different denoiser designs (U-Net, DiT, SiT, and **MotionSiT**) with varying proportions of pretraining data. The results show that our MotionSiT exhibits superior scalability and outperforms other methods.

Table 2: **Interpolants design.** The model is trained from scratch on HumanML3D [31]. The right arrow $\rightarrow$ means that the closer to the real motion, the better. **Bold** indicates the best result.

Method	R Precision ↑			FID↓	MM Dist↓	Diversity→	MModality↑
	Top 1	Top 2	Top 3				
Real	$0.511^{\pm.003}$	$0.703^{\pm.003}$	$0.797^{\pm.002}$	$0.002^{\pm.000}$	$2.974^{\pm.008}$	$9.503^{\pm.065}$	-
SBDM-VP	$0.503^{\pm.002}$	$0.705^{\pm.005}$	$0.791^{\pm.003}$	$0.094^{\pm.015}$	$2.774^{\pm.009}$	$9.642^{\pm.078}$	$2.209^{\pm.063}$
Linear	$0.535^{\pm.001}$	$0.733^{\pm.004}$	$0.837^{\pm.005}$	$0.065^{\pm.009}$	$2.744^{\pm.007}$	$9.595^{\pm.069}$	$2.449^{\pm.088}$
GVP	$0.542^{\pm.005}$	$0.741^{\pm.003}$	$0.840^{\pm.001}$	$0.058^{\pm.002}$	$2.736^{\pm.008}$	$\mathbf{9.588}^{\pm.043}$	$2.600^{\pm.039}$
FI (Linear)	$0.544^{\pm.001}$	$0.743^{\pm.003}$	$0.842^{\pm.005}$	$0.054^{\pm.002}$	$2.732^{\pm.005}$	$9.590^{\pm.077}$	$2.659^{\pm.064}$
FI (GVP)	$0.507^{\pm.005}$	$0.710^{\pm.002}$	$0.802^{\pm.006}$	$0.089^{\pm.008}$	$2.784^{\pm.001}$	$9.632^{\pm.057}$	$2.218^{\pm.067}$
LI (Linear)	$0.551^{\pm.005}$	$0.748^{\pm.001}$	$0.844^{\pm.003}$	$0.048^{\pm.002}$	$2.730^{\pm.014}$	$9.602^{\pm.047}$	$2.705^{\pm.029}$
LI (GVP)	$0.550^{\pm.001}$	$0.745^{\pm.001}$	$0.841^{\pm.003}$	$0.044^{\pm.005}$	$2.725^{\pm.012}$	$9.633^{\pm.072}$	$2.788^{\pm.075}$
GFI (Linear)	$\mathbf{0.562}^{\pm.004}$	$\mathbf{0.754}^{\pm.005}$	$\mathbf{0.847}^{\pm.005}$	$\mathbf{0.041}^{\pm.002}$	$\mathbf{2.711}^{\pm.006}$	$9.614^{\pm.056}$	$\mathbf{2.812}^{\pm.046}$

4.4 Ablation Study

Scalability. To showcase our method's scalability, we compare various denoiser architectures against our backbone using different proportions of pretraining data. As shown in Figure 6, our method consistently improves performance as data increases, demonstrating strong scalability and generalization capability. This scaling trend shows promising results for 3D human motion generation, as larger amounts of motion data can be acquired by HMR [28, 15] and MoCap [20, 36], highlighting a promising direction for foundational motion generative models.

Interpolants design. The interpolant approach is one of the key innovations in FlashMo. Since we propose the geometrically factorized interpolant, the choice of an appropriate interpolant function becomes particularly important. Both score-based [69] and velocity-field diffusion models [54] have explored different interpolant functions $\alpha(t)$ and variances $\sigma(t)$, as discussed in *Appendix A.1*. The results in Table 2 show that our geometric factorized interpolant with a linear interpolation function significantly outperforms other interpolants. Furthermore, when leveraging the factorized interpolant, the linear function exhibits a notable advantage compared to GVP, which contrasts with SiT [54] where GVP demonstrates better performance. This is mathematically sound because although factorization better aligns with the temporal-spatial structure of motion, combining it with

Table 3: **Sparsification parameters.** The model is trained from scratch on HumanML3D [31]. The right arrow $\rightarrow$ means that the closer to the real motion, the better. **Bold** indicates the best result.

Method	AIT(s)↓	R Precision ↑			FID↓	MM Dist↓	Diversity→	MModality↑
		Top 1	Top 2	Top 3				
Real	-	$0.511^{\pm.003}$	$0.703^{\pm.003}$	$0.797^{\pm.002}$	$0.002^{\pm.000}$	$2.974^{\pm.008}$	$9.503^{\pm.065}$	-
Full Attention	0.061	$0.565^{\pm.002}$	$0.759^{\pm.004}$	$0.850^{\pm.006}$	$0.033^{\pm.004}$	$2.708^{\pm.003}$	$9.598^{\pm.042}$	$2.833^{\pm.043}$
$\beta = 0.75$ $\gamma = 0.93$	**0.014**	$0.466^{\pm.003}$	$0.652^{\pm.005}$	$0.747^{\pm.006}$	$0.644^{\pm.003}$	$3.316^{\pm.007}$	$9.638^{\pm.071}$	$1.073^{\pm.042}$
$\beta = 0.75$ $\gamma = 0.95$	0.016	$0.517^{\pm.005}$	$0.694^{\pm.001}$	$0.798^{\pm.004}$	$0.305^{\pm.005}$	$3.035^{\pm.003}$	$9.745^{\pm.055}$	$2.653^{\pm.025}$
$\beta = 0.75$ $\gamma = 0.97$	0.019	$0.538^{\pm.001}$	$0.723^{\pm.004}$	$0.819^{\pm.002}$	$0.097^{\pm.004}$	$2.734^{\pm.005}$	$9.649^{\pm.053}$	$2.707^{\pm.052}$
$\beta = 0.50$ $\gamma = 0.93$	0.023	$0.515^{\pm.001}$	$0.690^{\pm.006}$	$0.792^{\pm.003}$	$0.065^{\pm.004}$	$2.752^{\pm.005}$	$9.598^{\pm.042}$	$2.619^{\pm.071}$
$\boldsymbol{\beta = 0.50}$ $\boldsymbol{\gamma = 0.95}$	0.027	**0.562**$^{\pm.004}$	**0.754**$^{\pm.005}$	**0.847**$^{\pm.005}$	$0.041^{\pm.002}$	**2.711**$^{\pm.006}$	$9.614^{\pm.056}$	$2.812^{\pm.046}$
$\beta = 0.50$ $\gamma = 0.97$	0.035	$0.559^{\pm.002}$	$0.752^{\pm.005}$	$0.844^{\pm.007}$	$0.040^{\pm.003}$	$2.713^{\pm.004}$	$9.442^{\pm.064}$	$2.820^{\pm.043}$
$\beta = 0.25$ $\gamma = 0.93$	0.042	$0.545^{\pm.005}$	$0.742^{\pm.002}$	$0.810^{\pm.006}$	$0.107^{\pm.003}$	$2.935^{\pm.004}$	**9.465**$^{\pm.031}$	$2.364^{\pm.045}$
$\beta = 0.25$ $\gamma = 0.95$	0.048	$0.560^{\pm.002}$	$0.753^{\pm.004}$	$0.846^{\pm.001}$	$0.040^{\pm.003}$	**2.711**$^{\pm.002}$	$9.657^{\pm.063}$	$2.819^{\pm.035}$
$\beta = 0.25$ $\gamma = 0.97$	0.053	**0.562**$^{\pm.005}$	$0.753^{\pm.002}$	$0.845^{\pm.003}$	**0.039**$^{\pm.004}$	$2.715^{\pm.002}$	$9.633^{\pm.053}$	**2.826**$^{\pm.063}$

Table 4: **Model configurations.** The model is trained from scratch on HumanML3D [31]. The right arrow $\rightarrow$ means that the closer to the real motion, the better. **Bold** and $\underline{\text{underline}}$ indicate best and second best results.

Method	AIT(s)↓	R Precision ↑			FID↓	MM Dist↓	Diversity→	MModality↑
		Top 1	Top 2	Top 3				
Real	-	$0.511^{\pm.003}$	$0.703^{\pm.003}$	$0.797^{\pm.002}$	$0.002^{\pm.000}$	$2.974^{\pm.008}$	$9.503^{\pm.065}$	-
$N_h = 2$	$\underline{0.024}$	$0.532^{\pm.004}$	$0.732^{\pm.002}$	$0.814^{\pm.002}$	$0.064^{\pm.003}$	$2.954^{\pm.005}$	$\underline{9.616}^{\pm.024}$	$2.426^{\pm.064}$
$N_h = 6$	0.032	$0.561^{\pm.001}$	$0.750^{\pm.006}$	$0.841^{\pm.002}$	$0.041^{\pm.004}$	$2.710^{\pm.002}$	$9.785^{\pm.053}$	$2.810^{\pm.061}$
$N_h = 8$	0.039	**0.566**$^{\pm.006}$	$\underline{0.752}^{\pm.003}$	**0.847**$^{\pm.002}$	$\underline{0.039}^{\pm.006}$	$\underline{2.709}^{\pm.003}$	$9.692^{\pm.047}$	$2.809^{\pm.083}$
$r = 2$	0.035	$0.542^{\pm.002}$	$0.745^{\pm.005}$	$0.839^{\pm.006}$	$0.047^{\pm.002}$	$2.722^{\pm.001}$	$9.684^{\pm.074}$	$2.793^{\pm.025}$
Depth = 4	**0.023**	$0.541^{\pm.006}$	$0.740^{\pm.001}$	$0.836^{\pm.002}$	$0.050^{\pm.004}$	$2.728^{\pm.001}$	$9.688^{\pm.074}$	$2.807^{\pm.047}$
Depth = 6	0.025	$0.553^{\pm.003}$	$0.749^{\pm.002}$	$0.840^{\pm.005}$	$0.045^{\pm.002}$	$2.719^{\pm.005}$	$9.704^{\pm.036}$	$2.809^{\pm.064}$
Depth = 10	0.033	$\underline{0.565}^{\pm.003}$	**0.754**$^{\pm.006}$	$\underline{0.846}^{\pm.001}$	**0.038**$^{\pm.001}$	**2.708**$^{\pm.007}$	$9.719^{\pm.064}$	$\underline{2.811}^{\pm.057}$
Ours	0.027	$0.562^{\pm.004}$	**0.754**$^{\pm.005}$	**0.847**$^{\pm.005}$	$0.041^{\pm.002}$	$2.711^{\pm.006}$	**9.614**$^{\pm.056}$	**2.812**$^{\pm.046}$

GVP breaks the variance preserving property, which is a theoretical guarantee in score-based and velocity-field diffusion training. In contrast, the linear interpolant does not have this issue.

Sparsification parameters. We conducted experiments with different head ratios β and attention thresholds γ, keeping all other settings the same. The results in Table 3 show that increasing the number of Lo-Fi heads and pruning more tokens improves speed but leads to a decrease in performance. In contrast, our sparsification parameters achieve an excellent balance between efficiency and sparsity, delivering performance comparable to full attention while achieving a **2.25×** speedup, enabled by unified training and inference without token granularity mismatch.

Model configurations. We investigate different model configuration including number of heads N_h and model depth of MotionSiT, compression ratio r of VAE, compare to our default settings ($N_h = 4$, $r = 4$, depth = 8). The results in Table 4 demonstrate the robustness of our model across different configurations. While increasing the model's depth and number of attention heads improves metrics such as FID and MM Dist, it sacrifices efficiency. Our chosen configuration balances performance and efficiency.

5 Conclusion

We present FlashMo, a frequency-aware sparse motion diffusion framework that addresses both efficiency and scalability challenges in 3D human motion generation. By introducing a unified training and inference sparsification strategy, FlashMo achieves a **2.25×** speedup over full attention without compromising motion quality. Our proposed MotionSiT architecture leverages a geometrically factorized interpolant with Lie group geodesics, enabling principled modeling of joint rotations on $\mathrm{SO}(3)$ manifolds. Extensive experiments on HumanML3D, and KIT-ML validate the effectiveness of FlashMo, demonstrating significant improvements over state-of-the-art methods in both performance and efficiency. Beyond empirical gains, FlashMo highlights the importance of respecting manifold geometry in diffusion processes, demonstrating that modeling motion trajectories along $\mathrm{SO}(3)$ geodesics preserves rotational consistency and yields smoother, physically coherent human motions compared to Euclidean approximations.

References

[1] Kazi Hasan Ibn Arif, JinYi Yoon, Dimitrios S Nikolopoulos, Hans Vandierendonck, Deepu John, and Bo Ji. Hired: Attention-guided token dropping for efficient inference of high-resolution vision-language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 1773–1781, 2025. 3

[2] German Barquero, Sergio Escalera, and Cristina Palmero. Seamless human motion composition with blended positional encodings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 457–469, 2024. 3

[3] Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020. 3

[4] Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models. In *European Conference on Computer Vision*, pages 19–35. Springer, 2024. 3

[5] Weiliang Chen, Fangfu Liu, Diankun Wu, Haowen Sun, Haixu Song, and Yueqi Duan. Dreamcinema: Cinematic transfer with free camera and 3d character. *arXiv preprint arXiv:2408.12601*, 2024. 2

[6] Wenshuo Chen, Haozhe Jia, Songning Lai, Keming Wu, Hongru Xiao, Lijie Hu, and Yutao Yue. Free-t2m: Frequency enhanced text-to-motion diffusion model with consistency loss. *arXiv preprint arXiv:2501.18232*, 2025. 3, 21, 24

[7] 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. 2, 7, 21, 24

[8] Xingyu Chen. Text-driven human motion generation with motion masked diffusion model. *arXiv preprint arXiv:2409.19686*, 2024. 3, 21, 24

[9] Xuanyao Chen, Zhijian Liu, Haotian Tang, Li Yi, Hang Zhao, and Song Han. Sparsevit: Revisiting activation sparsity for efficient high-resolution vision transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2061–2070, 2023. 3

[10] Zhaodong Chen, Zheng Qu, Yuying Quan, Liu Liu, Yufei Ding, and Yuan Xie. Dynamic n: M fine-grained structured sparse attention mechanism. In *Proceedings of the 28th ACM SIGPLAN Annual Symposium on Principles and Practice of Parallel Programming*, pages 369–379, 2023. 3

[11] Ziyu Chen, Jiawei Yang, Jiahui Huang, Riccardo de Lutio, Janick Martinez Esturo, Boris Ivanovic, Or Litany, Zan Gojcic, Sanja Fidler, Marco Pavone, et al. Omnire: Omni urban scene reconstruction. *arXiv preprint arXiv:2408.16760*, 2024. 1

[12] Seunggeun Chi, Hyung-gun Chi, Hengbo Ma, Nakul Agarwal, Faizan Siddiqui, Karthik Ramani, and Kwonjoon Lee. M2d2m: Multi-motion generation from text with discrete diffusion models. In *European Conference on Computer Vision*, pages 18–36. Springer, 2024. 3, 21, 24

[13] Jungbin Cho, Junwan Kim, Jisoo Kim, Minseo Kim, Mingu Kang, Sungeun Hong, Tae-Hyun Oh, and Youngjae Yu. Discord: Discrete tokens to continuous motion via rectified flow decoding. *arXiv preprint arXiv:2411.19527*, 2024. 2

[14] Setareh Cohan, Guy Tevet, Daniele Reda, Xue Bin Peng, and Michiel van de Panne. Flexible motion in-betweening with diffusion models. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–9, 2024. 3

[15] Lin Cong, Philipp Ruppel, Yizhou Wang, Xiang Pan, Norman Hendrich, and Jianwei Zhang. Efficient human motion reconstruction from monocular videos with physical consistency loss. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–9, 2023. 8

[16] 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. 2, 3, 7, 21

[17] Wenxun Dai, Ling-Hao Chen, Jingbo Wang, Jinpeng Liu, Bo Dai, and Yansong Tang. Motionlcm: Real-time controllable motion generation via latent consistency model. In *European Conference on Computer Vision*, pages 390–408. Springer, 2024. 2, 3, 7, 20, 21, 23

[18] Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*, 2023. 2, 3

[19] Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in neural information processing systems*, 35:16344–16359, 2022. 2, 3

[20] Fernando De la Torre, Jessica Hodgins, Adam Bargteil, Xavier Martin, Justin Macey, Alex Collado, and Pep Beltran. Guide to the carnegie mellon university multimodal activity (cmu-mmac) database. 2009. 8

[21] César Roberto De Souza, Adrien Gaidon, Yohann Cabon, Naila Murray, and Antonio Manuel López. Generating human action videos by coupling 3d game engines and probabilistic graphical models. *International Journal of Computer Vision*, 128(5):1505–1536, 2020. 2

[22] Juechu Dong, Boyuan Feng, Driss Guessous, Yanbo Liang, and Horace He. Flex attention: A programming model for generating optimized attention kernels. *arXiv preprint arXiv:2412.05496*, 2024. 3

[23] Gongfan Fang, Hongxu Yin, Saurav Muralidharan, Greg Heinrich, Jeff Pool, Jan Kautz, Pavlo Molchanov, and Xinchao Wang. Maskllm: Learnable semi-structured sparsity for large language models. *arXiv preprint arXiv:2409.17481*, 2024. 3

[24] Zhengcong Fei, Mingyuan Fan, Changqian Yu, and Junshi Huang. Scalable diffusion models with state space backbone. *arXiv preprint arXiv:2402.05608*, 2024. 2

[25] Elias Frantar and Dan Alistarh. Sparsegpt: Massive language models can be accurately pruned in one-shot. In *International Conference on Machine Learning*, pages 10323–10337. PMLR, 2023. 3

[26] Eduardo Gallo. The so (3) and se (3) lie algebras of rigid body rotations and motions and their application to discrete integration, gradient descent optimization, and state estimation. *arXiv preprint arXiv:2205.12572*, 2022. 2, 4

[27] Yizhao Gao, Zhichen Zeng, Dayou Du, Shijie Cao, Peiyuan Zhou, Jiaxing Qi, Junjie Lai, Hayden Kwok-Hay So, Ting Cao, Fan Yang, et al. Seerattention: Learning intrinsic sparse attention in your llms. *arXiv preprint arXiv:2410.13276*, 2024. 3

[28] Shubham Goel, Georgios Pavlakos, Jathushan Rajasegaran, Angjoo Kanazawa, and Jitendra Malik. Humans in 4d: Reconstructing and tracking humans with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14783–14794, 2023. 8

[29] Nils Graef and Andrew Wasielewski. Slim attention: cut your context memory in half without loss of accuracy–k-cache is all you need for mha. *arXiv preprint arXiv:2503.05840*, 2025. 3

[30] Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. Momask: Generative masked modeling of 3d human motions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1900–1910, 2024. 2, 7, 21, 24

[31] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5152–5161, 2022. 2, 6, 7, 8, 9, 20, 21, 23

[32] 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. 21, 24

[33] Bo Han, Hao Peng, Minjing Dong, Yi Ren, Yixuan Shen, and Chang Xu. Amd: Autoregressive motion diffusion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 2022–2030, 2024. 3

[34] Ali Hassani, Steven Walton, Jiachen Li, Shen Li, and Humphrey Shi. Neighborhood attention transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6185–6194, 2023. 3

[35] Yefei He, Feng Chen, Jing Liu, Wenqi Shao, Hong Zhou, Kaipeng Zhang, and Bohan Zhuang. Zipvl: Efficient large vision-language models with dynamic token sparsification and kv cache compression. 2024. 3

[36] Daniel Holden. Robust solving of optical motion capture data by denoising. *ACM Transactions on Graphics (TOG)*, 37(4):1–12, 2018. 8

[37] Stefan Holzinger, Martin Arnold, and Johannes Gerstmayr. Evaluation and implementation of lie group integration methods for rigid multibody systems. *Multibody System Dynamics*, pages 1–34, 2024. 2, 4

[38] 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. 3, 7, 21, 24

[39] 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, 2024. 2, 3, 21, 24

[40] Huiqiang Jiang, Yucheng Li, Chengruidong Zhang, Qianhui Wu, Xufang Luo, Surin Ahn, Zhenhua Han, Amir Abdi, Dongsheng Li, Chin-Yew Lin, et al. Minference 1.0: Accelerating pre-filling for long-context llms via dynamic sparse attention. *Advances in Neural Information Processing Systems*, 37:52481–52515, 2024. 3

[41] Lei Jiang, Ye Wei, and Hao Ni. Motionpcm: Real-time motion synthesis with phased consistency model. *arXiv preprint arXiv:2501.19083*, 2025. 2, 3, 7, 21, 24

[42] 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*, 36:15497–15518, 2023. 2, 21, 24

[43] Zixi Kang, Xinghan Wang, and Yadong Mu. Biomodiffuse: Physics-guided biomechanical diffusion for controllable and authentic human motion synthesis. *arXiv preprint arXiv:2503.06151*, 2025. 3, 21, 24

[44] Hanyang Kong, Kehong Gong, Dongze Lian, Michael Bi Mi, and Xinchao Wang. Priority-centric human motion generation in discrete latent space. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14806–14816, 2023. 2, 21, 24

[45] Xunhao Lai, Jianqiao Lu, Yao Luo, Yiyuan Ma, and Xun Zhou. Flexprefill: A context-aware sparse attention mechanism for efficient long-sequence inference. *arXiv preprint arXiv:2502.20766*, 2025. 3

[46] Heejun Lee, Jina Kim, Jeffrey Willette, and Sung Ju Hwang. Sea: Sparse linear attention with estimated attention mask. *arXiv preprint arXiv:2310.01777*, 2023. 3

[47] Chengjian Li, Xiangbo Shu, Qiongjie Cui, Yazhou Yao, and Jinhui Tang. Ftmomamba: Motion generation with frequency and text state space models. *arXiv preprint arXiv:2411.17532*, 2024. 3, 21

[48] Hao Li, Shamit Lal, Zhiheng Li, Yusheng Xie, Ying Wang, Yang Zou, Orchid Majumder, R Manmatha, Zhuowen Tu, Stefano Ermon, et al. Efficient scaling of diffusion transformers for text-to-image generation. *arXiv preprint arXiv:2412.12391*, 2024. 2

[49] Ling Li, David Thorsley, and Joseph Hassoun. Sait: Sparse vision transformers through adaptive token pruning. *arXiv preprint arXiv:2210.05832*, 2022. 3

[50] Zeyu Ling, Bo Han, Shiyang Li, Hongdeng Shen, Jikang Cheng, and Changqing Zou. Motionllama: A unified framework for motion synthesis and comprehension. *arXiv preprint arXiv:2411.17335*, 2024. 2, 6, 7

[51] Songhua Liu, Zhenxiong Tan, and Xinchao Wang. Clear: Conv-like linearization revs pre-trained diffusion transformers up. *arXiv preprint arXiv:2412.16112*, 2024. 3

[52] Ting Liu, Liangtao Shi, Richang Hong, Yue Hu, Quanjun Yin, and Linfeng Zhang. Multi-stage vision token dropping: Towards efficient multimodal large language model. *arXiv preprint arXiv:2411.10803*, 2024. 3

[53] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 3

[54] Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision*, pages 23–40. Springer, 2024. 2, 8, 17

[55] J Pablo Muoz, Jinjie Yuan, and Nilesh Jain. Shears: Unstructured sparsity with neural low-rank adapter search. *arXiv preprint arXiv:2404.10934*, 2024. 3

[56] Matteo Pagliardini, Daniele Paliotta, Martin Jaggi, and François Fleuret. Fast attention over long sequences with dynamic sparse flash attention. *Advances in Neural Information Processing Systems*, 36:59808–59831, 2023. 3

[57] Zizheng Pan, Jianfei Cai, and Bohan Zhuang. Fast vision transformers with hilo attention. *Advances in Neural Information Processing Systems*, 35:14541–14554, 2022. 3

[58] 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. 17

[59] Ekkasit Pinyoanuntapong, Muhammad Usama Saleem, Pu Wang, Minwoo Lee, Srijan Das, and Chen Chen. Bamm: bidirectional autoregressive motion model. In *European Conference on Computer Vision*, pages 172–190. Springer, 2024. 2, 7, 21, 24

[60] 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. 2, 7, 21, 24

[61] Matthias Plappert, Christian Mandery, and Tamim Asfour. The kit motion-language dataset. *Big data*, 4(4):236–252, 2016. 2, 6, 7, 20, 24

[62] Sigal Raab, Inbal Leibovitch, Guy Tevet, Moab Arar, Amit Haim Bermano, and Daniel Cohen-Or. Single motion diffusion. In *The Twelfth International Conference on Learning Representations*. 3

[63] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *Advances in neural information processing systems*, 34:13937–13949, 2021. 3

[64] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2

[65] Agon Serifi, Ruben Grandia, Espen Knoop, Markus Gross, and Moritz Bächer. Robot motion diffusion model: Motion generation for robotic characters. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–9, 2024. 1

[66] Yoni Shafir, Guy Tevet, Roy Kapon, and Amit Haim Bermano. Human motion diffusion as a generative prior. In *The Twelfth International Conference on Learning Representations*. 3

[67] Jay Shah, Ganesh Bikshandi, Ying Zhang, Vijay Thakkar, Pradeep Ramani, and Tri Dao. Flashattention-3: Fast and accurate attention with asynchrony and low-precision. *Advances in Neural Information Processing Systems*, 37:68658–68685, 2024. 2, 3

[68] Jifeng Song, Kai Huang, Xiangyu Yin, Boyuan Yang, and Wei Gao. Achieving sparse activation in small language models. *arXiv preprint arXiv:2406.06562*, 2024. 3

[69] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 8

[70] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022. 2, 21, 24

[71] Yuchuan Tian, Zhijun Tu, Hanting Chen, Jie Hu, Chao Xu, and Yunhe Wang. U-dits: Downsample tokens in u-shaped diffusion transformers. *arXiv preprint arXiv:2405.02730*, 2024. 2

[72] Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. 2

[73] Yin Wang, Zhiying Leng, Frederick WB Li, Shun-Cheng Wu, and Xiaohui Liang. Fg-t2m: Fine-grained text-driven human motion generation via diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22035–22044, 2023. 2, 21, 24

[74] Yin Wang, Mu Li, Jiapeng Liu, Zhiying Leng, Frederick WB Li, Ziyao Zhang, and Xiaohui Liang. Fg-t2m++: Llms-augmented fine-grained text driven human motion generation. *International Journal of Computer Vision*, pages 1–17, 2025. 2, 3, 21, 24

[75] Cong Wei, Brendan Duke, Ruowei Jiang, Parham Aarabi, Graham W Taylor, and Florian Shkurti. Sparsifiner: Learning sparse instance-dependent attention for efficient vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22680–22689, 2023. 3

[76] Mingjie Wei, Xuemei Xie, and Guangming Shi. Acmo: Attribute controllable motion generation. *arXiv preprint arXiv:2503.11038*, 2025. 3, 21

[77] Haocheng Xi, Shuo Yang, Yilong Zhao, Chenfeng Xu, Muyang Li, Xiuyu Li, Yujun Lin, Han Cai, Jintao Zhang, Dacheng Li, et al. Sparse videogen: Accelerating video diffusion transformers with spatial-temporal sparsity. *arXiv preprint arXiv:2502.01776*, 2025. 3

[78] Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453*, 2023. 3

[79] Yiming Xie, Varun Jampani, Lei Zhong, Deqing Sun, and Huaizu Jiang. Omnicontrol: Control any joint at any time for human motion generation. *arXiv preprint arXiv:2310.08580*, 2023. 3

[80] Zhenyu Xie, Yang Wu, Xuehao Gao, Zhongqian Sun, Wei Yang, and Xiaodan Liang. Towards detailed text-to-motion synthesis via basic-to-advanced hierarchical diffusion model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 6252–6260, 2024. 2, 21, 24

[81] Ruyi Xu, Guangxuan Xiao, Haofeng Huang, Junxian Guo, and Song Han. Xattention: Block sparse attention with antidiagonal scoring. *arXiv preprint arXiv:2503.16428*, 2025. 3

[82] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. 2

[83] Yongjing Ye, Libin Liu, Lei Hu, and Shihong Xia. Neural3points: Learning to generate physically realistic full-body motion for virtual reality users. In *Computer Graphics Forum*, volume 41, pages 183–194. Wiley Online Library, 2022. 2

[84] Jingyang Yuan, Huazuo Gao, Damai Dai, Junyu Luo, Liang Zhao, Zhengyan Zhang, Zhenda Xie, YX Wei, Lean Wang, Zhiping Xiao, et al. Native sparse attention: Hardware-aligned and natively trainable sparse attention. *arXiv preprint arXiv:2502.11089*, 2025. 3

[85] Weihao Yuan, Yisheng He, Weichao Shen, Yuan Dong, Xiaodong Gu, Zilong Dong, Liefeng Bo, and Qixing Huang. Mogents: Motion generation based on spatial-temporal joint modeling. *Advances in Neural Information Processing Systems*, 37:130739–130763, 2024. 2, 7, 20, 21, 23, 24

[86] Zhihang Yuan, Hanling Zhang, Lu Pu, Xuefei Ning, Linfeng Zhang, Tianchen Zhao, Shengen Yan, Guohao Dai, and Yu Wang. Ditfastattn: Attention compression for diffusion transformer models. *Advances in Neural Information Processing Systems*, 37:1196–1219, 2024. 3

[87] Shaojun Yue. Human motion tracking and positioning for augmented reality. *Journal of Real-Time Image Processing*, 18(2):357–368, 2021. 2

[88] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297, 2020. 3

[89] Ling-An Zeng, Guohong Huang, Gaojie Wu, and Wei-Shi Zheng. Light-t2m: A lightweight and fast model for text-to-motion generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 9797–9805, 2025. 2, 3, 7, 21, 24

[90] Xingzu Zhan, Chen Xie, Haoran Sun, and Xiaochun Mai. Histf mamba: Hierarchical spatiotemporal fusion with multi-granular body-spatial modeling for high-fidelity text-to-motion generation. *arXiv preprint arXiv:2503.06897*, 2025. 3, 21, 24

[91] Biao Zhang, Ivan Titov, and Rico Sennrich. Sparse attention with linear units. *arXiv preprint arXiv:2104.07012*, 2021. 3

[92] Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Yong Zhang, Hongwei Zhao, Hongtao Lu, Xi Shen, and Ying Shan. Generating human motion from textual descriptions with discrete representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14730–14740, 2023. 2, 3, 21, 24

[93] Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. Motiondiffuse: Text-driven human motion generation with diffusion model. *IEEE transactions on pattern analysis and machine intelligence*, 46(6):4115–4128, 2024. 2, 21, 24

[94] 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. 2, 21, 24

[95] Mingyuan Zhang, Huirong Li, Zhongang Cai, Jiawei Ren, Lei Yang, and Ziwei Liu. Finemogen: Fine-grained spatio-temporal motion generation and editing. *Advances in Neural Information Processing Systems*, 36:13981–13992, 2023. 2, 3, 21, 24

[96] Peiyuan Zhang, Yongqi Chen, Runlong Su, Hangliang Ding, Ion Stoica, Zhenghong Liu, and Hao Zhang. Fast video generation with sliding tile attention. *arXiv preprint arXiv:2502.04507*, 2025. 3

[97] Zeyu Zhang, Hang Gao, Akide Liu, Qi Chen, Feng Chen, Yiran Wang, Danning Li, and Hao Tang. Kmm: Key frame mask mamba for extended motion generation. *arXiv preprint arXiv:2411.06481*, 2024. 2, 3

[98] Zeyu Zhang, Akide Liu, Qi Chen, Feng Chen, Ian Reid, Richard Hartley, Bohan Zhuang, and Hao Tang. Infinimotion: Mamba boosts memory in transformer for arbitrary long motion generation. *arXiv preprint arXiv:2407.10061*, 2024. 3

[99] 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. 2, 3, 7, 21, 24

[100] 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. 2, 5

[101] Zeyu Zhang, Yiran Wang, Biao Wu, Shuo Chen, Zhiyuan Zhang, Shiya Huang, Wenbo Zhang, Meng Fang, Ling Chen, and Yang Zhao. Motion avatar: Generate human and animal avatars with arbitrary motion. *arXiv preprint arXiv:2405.11286*, 2024. 2

[102] Xuanlei Zhao, Xiaolong Jin, Kai Wang, and Yang You. Real-time video generation with pyramid attention broadcast. *arXiv preprint arXiv:2408.12588*, 2024. 3

[103] Qihui Zhou, Peiqi Yin, Pengfei Zuo, and James Cheng. Progressive sparse attention: Algorithm and system co-design for efficient attention in llm serving. *arXiv preprint arXiv:2503.00392*, 2025. 3

[104] 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, 2024. 2, 3, 7, 21, 24

[105] Qianchao Zhu, Jiangfei Duan, Chang Chen, Siran Liu, Xiuhong Li, Guanyu Feng, Xin Lv, Huanqi Cao, Xiao Chuanfu, Xingcheng Zhang, et al. Sampleattention: Near-lossless acceleration of long context llm inference with adaptive structured sparse attention. *arXiv preprint arXiv:2406.15486*, 2024. 3

Appendix

A Preliminaries

A.1 Scalable Interpolant Transformer (SiT)

Scalable Interpolant Transformers (SiT) [54] is a diffusion transformer [58] based on *stochastic interpolants*, which defines a class of time-indexed stochastic processes that map a noise sample $\mathbf{x}_0 \sim p_0$ to an intermediate point $\mathbf{x}(t)$ over $t \in [0, 1]$. These interpolants are defined without requiring a target sample $\mathbf{x}_1$ (as in standard DDPMs), and instead use structured noise perturbation with learned dynamics to generate samples.

The interpolant is defined as:

$$\mathbf{x}(t) = \alpha(t)\mathbf{x}_0 + \sigma(t)\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, I),$$

where $\alpha(t)$ and $\sigma(t)$ are time-dependent scalar functions, $\mathbf{x}_0$ is a noise sample (analogous to initial state in forward diffusion). It holds that $\mathbf{x}(0) = \mathbf{x}_0$, and $\mathbf{x}(1)$ ideally follows p_1, the data distribution.

These interpolants allow generation and learning without explicitly simulating a forward diffusion trajectory from $\mathbf{x}_1$ to $\mathbf{x}_0$ as in traditional diffusion models.

Forward SDE. The forward generative process in diffusion models is typically described using a stochastic differential equation of the form:

$$d\mathbf{x} = f(\mathbf{x}, t)\, dt + g(t)\, d\mathbf{w}_t,$$

where $f(\mathbf{x}, t)$ is a drift function, $g(t)$ is a diffusion coefficient, $\mathbf{w}_t$ is standard Brownian motion.

Reverse-time SDE. The reverse-time dynamics of this process can be derived from the theory of time-reversal of stochastic processes. The reverse time SDE is:

$$d\mathbf{x} = \left[f(\mathbf{x}, t) - g(t)^2 \nabla_{\mathbf{x}} \log p_t(\mathbf{x})\right] dt + g(t) d\bar{\mathbf{w}}_t,$$

where $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$ is the *score function*, which denotes the gradient of the log-density at time t, $d\bar{\mathbf{w}}_t$ is a reverse-time Brownian motion.

This reverse-time SDE shows that, to sample from the data distribution starting from noise, we need to access the time-dependent score function of intermediate states. This is typically learned using score matching in diffusion models.

Probability flow ODE. An alternative, deterministic formulation of the same marginal distributions is given by the probability flow ODE:

$$\frac{d\mathbf{x}}{dt} = f(\mathbf{x}, t) - \frac{1}{2} g(t)^2 \nabla_{\mathbf{x}} \log p_t(\mathbf{x}).$$

Unlike the reverse SDE, this ODE yields a deterministic mapping from $\mathbf{x}_0$ to $\mathbf{x}_1$. Critically, both the reverse-time SDE and the probability flow ODE share the same marginal distribution $p_t(\mathbf{x})$ for each t.

This connection allows one to model generation either stochastically (via sampling the reverse SDE) or deterministically (via integrating the ODE). In SiT, the idea is to sidestep direct score estimation and instead predict the time-derivative of the interpolant path.

Velocity fields and learning objectives. Rather than explicitly learning the score function $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$, SiT models the trajectory of the interpolant $\mathbf{x}(t)$ by learning its *velocity field*:

$$\mathbf{v}(\mathbf{x}(t), t) := \frac{d\mathbf{x}(t)}{dt}.$$

Given the analytic form of the interpolant:

$$\mathbf{x}(t) = \alpha(t)\mathbf{x}_0 + \sigma(t)\boldsymbol{\epsilon},$$

its time derivative is:

$$\frac{d\mathbf{x}(t)}{dt} = \dot{\alpha}(t)\mathbf{x}_0 + \dot{\sigma}(t)\boldsymbol{\epsilon}.$$

Since $\mathbf{x}_0$ and $\boldsymbol{\epsilon}$ are both known (sampled during training), this velocity is analytically computable.

The SiT model learns a velocity estimator $\mathbf{v}_\theta(\mathbf{x}(t), t)$ by minimizing the expected squared error between the predicted and true velocity:

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\int_0^1 \left\|\mathbf{v}_\theta(\mathbf{x}(t), t) - \frac{d\mathbf{x}(t)}{dt}\right\|^2 dt\right].$$

This learning objective removes the need for score estimation or denoising objectives and allows SiT to scale better.

Interpolants design. The choice of interpolant functions $\alpha(t)$, the deterministic scaling of the input noise $\mathbf{x}_0$ and $\sigma(t)$, the time-dependent standard deviation controlling the magnitude of the stochastic perturbation directly determines the geometry of the stochastic path $\mathbf{x}(t) = \alpha(t)\mathbf{x}_0 + \sigma(t)\boldsymbol{\epsilon}$. This, in turn, influences training dynamics, sample quality, and generalization. Below, we describe two common interpolant designs that capture different trade-offs

The linear interpolant is defined as:

$$\alpha(t) = 1 - t, \quad \sigma(t) = \sqrt{t}.$$

This design induces a *linear* interpolation in the input $\mathbf{x}_0$ and a *square-root* scaling of the noise. At $t = 0$, we have $\mathbf{x}(0) = \mathbf{x}_0$; at $t = 1$, $\alpha(1) = 0$ and $\sigma(1) = 1$, hence $\mathbf{x}(1) = \boldsymbol{\epsilon}$, a pure noise sample.

This interpolant is simple and intuitive, but its marginal distribution $p_t(\mathbf{x})$ varies in both mean and variance across time. Specifically:

$$\mathbb{E}[\mathbf{x}(t)] = \alpha(t)\mathbb{E}[\mathbf{x}_0] = 0, \quad \mathrm{Var}[\mathbf{x}(t)] = \alpha(t)^2 + \sigma(t)^2 = (1-t)^2 + t.$$

Thus, the total variance is time-varying.

To address the variance inconsistency, the GVP interpolant is constructed such that the total variance remains constant over time:

$$\alpha(t)^2 + \sigma(t)^2 = 1.$$

A canoncial choice under this constraint is:

$$\alpha(t) = \cos\left(\frac{\pi}{2}t\right), \quad \sigma(t) = \sin\left(\frac{\pi}{2}t\right).$$

This design ensures that:

$$\mathbf{x}(t) \sim \mathcal{N}(0, I), \quad \forall t \in [0, 1].$$

That is, the marginal distribution of $\mathbf{x}(t)$ stays isotropic Gaussian throughout the path. This simplifies score estimation and enhances training stability. Moreover, the smooth transition from $\mathbf{x}_0$ to noise is nonlinear, leading to smoother gradients and more coherent sample trajectories.

The choice between them depends on downstream task requirements and model capacity.

A.2 Lie Groups for Rigid Body Rotations and Motions

Rigid body rotations and transformations in three-dimensional space are not elements of Euclidean space, but instead belong to structured non-Euclidean manifolds with group structures, specifically Lie groups. This geometric structure is critical for ensuring mathematically consistent operations such as interpolation, averaging, and noise perturbation, which are frequently needed in motion analysis and generation tasks.

The Special Orthogonal Group $\mathrm{SO}(3)$**.** The space of all 3D rotation matrices forms a Lie group known as the special orthogonal group:

$$\mathrm{SO}(3) = \left\{R \in \mathbb{R}^{3\times 3} \mid R^\top R = I,\ \det(R) = 1\right\},$$

which is a compact, connected, non-commutative Lie group of dimension 3. Each element of $\mathrm{SO}(3)$ represents a proper rotation in $\mathbb{R}^3$, and the group operation is matrix multiplication. The non-Euclidean nature of $\mathrm{SO}(3)$ implies that standard linear operations, such as averaging two rotation matrices or interpolating between them, may lead to results that no longer lie on the manifold.

The Special Euclidean Group $\mathrm{SE}(3)$**.** For full rigid body transformations, including both rotation and translation, the appropriate Lie group is $\mathrm{SE}(3)$:

$$\mathrm{SE}(3) = \left\{ \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \in \mathbb{R}^{4\times 4} \;\middle|\; R \in \mathrm{SO}(3),\ t \in \mathbb{R}^3 \right\},$$

which is a 6-dimensional, non-compact, non-commutative Lie group. The group operation is again matrix multiplication, and $\mathrm{SE}(3)$ encapsulates both orientation and position of a rigid body in space.

Lie Algebras and Local Coordinates. Associated with each Lie group $\mathcal{G}$ is a Lie algebra $\mathfrak{g}$, which serves as the tangent space at the identity element and provides a local, linear coordinate system for the manifold. For $\mathrm{SO}(3)$, the Lie algebra $\mathfrak{so}(3)$ consists of all 3D skew-symmetric matrices:

$$\mathfrak{so}(3) = \left\{ A \in \mathbb{R}^{3\times 3} \mid A^\top = -A \right\}.$$

A standard isomorphism between $\mathbb{R}^3$ and $\mathfrak{so}(3)$ is provided by the **hat operator**:

$$[\omega]_\times = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}, \quad \omega \in \mathbb{R}^3,$$

which maps a vector to its corresponding skew-symmetric matrix. The inverse operation is the **vee operator**, mapping from $\mathfrak{so}(3)$ to $\mathbb{R}^3$.

Exponential and Logarithmic Maps. The **exponential map** $\exp : \mathfrak{g} \to \mathcal{G}$ and its inverse, the **logarithmic map** $\log : \mathcal{G} \to \mathfrak{g}$, provide the tools to move between the manifold and its tangent space. For $\mathrm{SO}(3)$, the exponential map is given in closed form by Rodrigues' formula:

$$\exp([\omega]_\times) = I + \frac{\sin\theta}{\theta}[\omega]_\times + \frac{1-\cos\theta}{\theta^2}[\omega]_\times^2, \quad \theta = \|\omega\|.$$

This constructs a rotation matrix corresponding to a rotation of angle θ around axis $\omega/\|\omega\|$. The logarithmic map inverts this operation, computing the minimal-axis rotation vector ω corresponding to a given rotation matrix R:

$$\log(R) = \frac{\theta}{2\sin\theta}(R - R^\top), \quad \theta = \cos^{-1}\left(\frac{\mathrm{Tr}(R)-1}{2}\right).$$

Interpolation and Perturbation on Lie Groups. A major benefit of the Lie group structure is that interpolation and noise perturbation can be carried out in the tangent space, ensuring that the results lie back on the manifold after mapping. Given two rotations $R_1, R_2 \in \mathrm{SO}(3)$, a geodesic interpolation can be defined via:

$$R(t) = R_1 \cdot \exp\left(t \cdot \log(R_1^\top R_2)\right), \quad t \in [0,1],$$

which traces the shortest path on the manifold between R_1 and R_2. More generally, any operation of the form:

$$R = \exp(\omega), \quad \omega \sim \mathcal{N}(0, \Sigma),$$

defines a distribution on $\mathrm{SO}(3)$ by sampling from a Gaussian in the tangent space $\mathbb{R}^3$ and mapping to the manifold via the exponential map. Such constructions are widely used in manifold-aware generative models and motion synthesis.

Extensions to $\mathrm{SE}(3)$**.** For rigid body motion in $\mathrm{SE}(3)$, the corresponding Lie algebra $\mathfrak{se}(3)$ is a 6-dimensional space encoding translational and rotational velocities. Elements of $\mathfrak{se}(3)$ are typically expressed using twist coordinates $(v, \omega) \in \mathbb{R}^6$, and the exponential or logarithmic maps generalize accordingly. The Baker-Campbell-Hausdorff formula governs the non-linear composition of transformations, and matrix representations of $\exp$ and $\log$ are available via the theory of screw motions.

This Lie group machinery forms the mathematical foundation for handling rotation and transformation data in a consistent, geometry-aware manner, and is indispensable in domains such as robotics, graphics, and motion modeling.

B User Study

This study conducts a comprehensive user evaluation of our method compared with MotionLCM [17] and MoGenTS [85]. We assess the real-world applicability of motion sequences generated by Motion Anything and baseline models using a Google Forms survey completed by 50 participants. As shown in Figure 7, the user interface presents 3–4 motion clips (Videos 1–3/4) generated by the same model, followed by a comparative set (Videos A–C) from different models. Participants evaluate each animation based on motion accuracy and overall user experience, using a 3-point scale (1 = low, 3 = high). In the comparison section, users select the model they perceive as most realistic and engaging. This evaluation is designed to measure both the fidelity of the generated motion to real-world human movement and the overall effectiveness of each model in delivering visually compelling results.

Results:

- Our method achieved a motion quality rating of **2.92**, with **94%** of participants agreeing that it produces high-quality motion with minimal jitter, sliding, or unrealistic artifacts.
- For motion diversity, we received a rating of **2.86**, with **90%** of participants indicating that our method generates complex and varied motion sequences.
- In terms of text-motion alignment, our model scored **2.80**, and **82%** of users reported that the generated motions were well-aligned with the given text descriptions.
- Notably, **92%** of participants preferred our method over competing approaches.

C Qualitative Results

To qualitatively evaluate our performance in text-to-motion generation, we compare the visualizations generated by our method with those produced by both state-of-the-art diffusion and VQ-VAE based methods specializing in text-to-motion generation, including MotionLCM [17] and MoGenTS [85]. The text prompts are customized based on the HumanML3D [31] test set. As shown in Figure 8 and video demos, our method generates motions with superior quality, greater diversity, and better alignment between text and motion compared to the previous state-of-the-art methods.

D Full Comparison Tables

To comprehensively evaluate our method on text-to-motion generation, we report full comparisons with prior approaches in Table 5 and 6. Our method consistently achieves state-of-the-art performance on both HumanML3D [31] and KIT-ML [61], outperforming existing baselines across all metrics.

E Broader Impacts

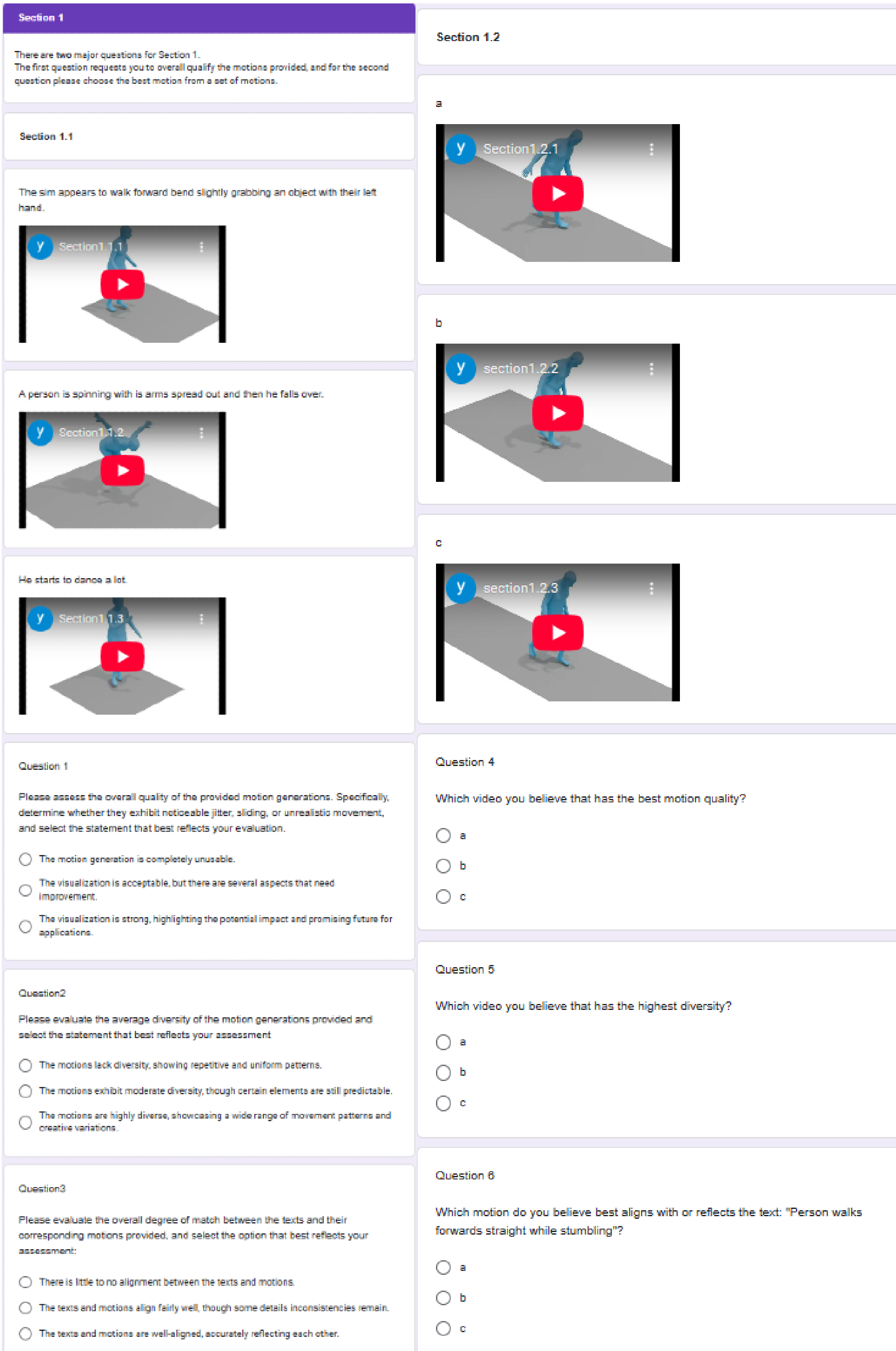
Our work advances the field of 3D human motion generation by addressing key limitations in efficiency and scalability that hinder real-world deployment. By introducing frequency-aware sparsification and a scalable transformer architecture with principled geometric modeling, FlashMo offers a more practical solution for generating realistic human motion at scale. This has broad implications for downstream applications such as human-robot interaction, AR/VR environments, animation, and digital avatars. By reducing computational demands without sacrificing quality, our approach lowers the barrier to adoption in resource-constrained settings and paves the way for real-time, interactive, and embodied AI systems.

F Limitations and Future Work

Currently, our design has not yet explored other interpolant functions, and we still rely on learning latent diffusion models on the tangent space. In future work, we will investigate how different interpolant forms benefit temporal-spatial factorization and Lie group geometric interpolation for improved motion modeling. Moreover, we will continue to investigate manifold geometry in motion generation from an optimization perspective.

Table 5: Comparison of text-to-motion generation on HumanML3D [31] dataset. → indicates the closer to real data, the better. **Bold** and underline indicate best and second best results. Efficient motion diffusion models are highlighted in blue.

Method	Venue	AIT(s) ↓	R-Precision ↑			FID ↓	MM Dist ↓	Diversity →	MModality ↑
			Top-1	Top-2	Top-3				
Real	-	-	$0.511^{\pm.003}$	$0.703^{\pm.003}$	$0.797^{\pm.002}$	$0.002^{\pm.000}$	$2.974^{\pm.008}$	$9.503^{\pm.065}$	-
TM2T [32]	ECCV 2022	0.760	$0.424^{\pm.003}$	$0.618^{\pm.003}$	$0.729^{\pm.002}$	$1.501^{\pm.017}$	$3.467^{\pm.011}$	$8.589^{\pm.076}$	$2.424^{\pm.093}$
T2M-GPT [92]	CVPR 2023	0.380	$0.492^{\pm.003}$	$0.679^{\pm.002}$	$0.775^{\pm.002}$	$0.141^{\pm.005}$	$3.121^{\pm.009}$	$9.722^{\pm.082}$	$1.831^{\pm.048}$
CoMo [39]	ECCV 2024	0.620	$0.502^{\pm.002}$	$0.692^{\pm.007}$	$0.790^{\pm.002}$	$0.262^{\pm.004}$	$3.032^{\pm.015}$	$9.936^{\pm.066}$	$1.013^{\pm.046}$
MMM [60]	CVPR 2024	0.081	$0.504^{\pm.003}$	$0.696^{\pm.003}$	$0.794^{\pm.002}$	$0.080^{\pm.003}$	$2.998^{\pm.007}$	$9.411^{\pm.058}$	$1.164^{\pm.041}$
MoMask [30]	CVPR 2024	0.120	$0.521^{\pm.002}$	$0.713^{\pm.002}$	$0.807^{\pm.002}$	$0.045^{\pm.002}$	$2.958^{\pm.008}$	-	$1.241^{\pm.040}$
BAMM [59]	ECCV 2024	0.411	$0.525^{\pm.002}$	$0.720^{\pm.003}$	$0.814^{\pm.003}$	$0.055^{\pm.002}$	$2.919^{\pm.008}$	$9.717^{\pm.089}$	$1.687^{\pm.051}$
MoGenTS [85]	NeurIPS 2024	0.181	$0.529^{\pm.003}$	$0.719^{\pm.002}$	$0.812^{\pm.002}$	$0.033^{\pm.001}$	$2.867^{\pm.006}$	$9.570^{\pm.077}$	-
MDM [70]	ICLR 2023	24.74	$0.320^{\pm.005}$	$0.498^{\pm.004}$	$0.611^{\pm.007}$	$0.544^{\pm.044}$	$5.566^{\pm.027}$	$9.559^{\pm.086}$	$2.799^{\pm.072}$
MotionDiffuse [93]	TPAMI 2024	14.74	$0.491^{\pm.001}$	$0.681^{\pm.001}$	$0.782^{\pm.001}$	$0.630^{\pm.001}$	$3.113^{\pm.001}$	$9.410^{\pm.049}$	$1.553^{\pm.042}$
MLD [7]	CVPR 2023	0.217	$0.481^{\pm.003}$	$0.673^{\pm.003}$	$0.772^{\pm.002}$	$0.473^{\pm.013}$	$3.196^{\pm.010}$	$9.724^{\pm.082}$	$2.413^{\pm.079}$
ReMoDiffuse [94]	ICCV 2023	0.624	$0.510^{\pm.005}$	$0.698^{\pm.006}$	$0.795^{\pm.004}$	$0.103^{\pm.004}$	$2.974^{\pm.016}$	$9.018^{\pm.075}$	$1.795^{\pm.043}$
M2DM [44]	ICCV 2023	-	$0.497^{\pm.003}$	$0.682^{\pm.002}$	$0.763^{\pm.003}$	$0.352^{\pm.005}$	$3.134^{\pm.010}$	$9.926^{\pm.073}$	$\mathbf{3.587}^{\pm.072}$
Fg-T2M [73]	ICCV 2023	-	$0.492^{\pm.002}$	$0.683^{\pm.003}$	$0.783^{\pm.002}$	$0.243^{\pm.019}$	$3.109^{\pm.007}$	$9.278^{\pm.072}$	$1.614^{\pm.049}$
FineMoGen [95]	NeurIPS 2023	-	$0.504^{\pm.002}$	$0.690^{\pm.002}$	$0.784^{\pm.002}$	$0.151^{\pm.008}$	$2.998^{\pm.008}$	$9.263^{\pm.094}$	$2.696^{\pm.079}$
GraphMotion [42] (50-step)	NeurIPS 2023	0.776	$0.496^{\pm.003}$	$0.686^{\pm.003}$	$0.778^{\pm.002}$	$0.118^{\pm.008}$	$3.143^{\pm.009}$	$9.796^{\pm.069}$	$2.603^{\pm.095}$
GraphMotion [42] (150-step)	NeurIPS 2023	2.552	$0.504^{\pm.003}$	$0.699^{\pm.002}$	$0.785^{\pm.002}$	$0.116^{\pm.007}$	$3.070^{\pm.008}$	$9.692^{\pm.067}$	$2.766^{\pm.096}$
B2A-HDM [80]	AAAI 2024	-	$0.511^{\pm.002}$	$0.699^{\pm.002}$	$0.791^{\pm.002}$	$0.084^{\pm.004}$	$3.020^{\pm.010}$	$\mathbf{9.526}^{\pm.080}$	$1.914^{\pm.078}$
M2D2M [12]	ECCV 2024	-	-	-	$0.799^{\pm.002}$	$0.087^{\pm.004}$	$3.018^{\pm.008}$	$9.672^{\pm.086}$	$2.115^{\pm.079}$
MotionLCM [17] (1-step)	ECCV 2024	$\underline{0.030}$	$0.502^{\pm.003}$	$0.701^{\pm.002}$	$0.803^{\pm.002}$	$0.467^{\pm.012}$	$3.022^{\pm.009}$	$9.631^{\pm.066}$	$2.172^{\pm.082}$
MotionLCM [17] (2-step)	ECCV 2024	0.035	$0.505^{\pm.003}$	$0.705^{\pm.002}$	$0.805^{\pm.002}$	$0.368^{\pm.011}$	$2.986^{\pm.008}$	$9.640^{\pm.052}$	$2.187^{\pm.094}$
MotionLCM [17] (4-step)	ECCV 2024	0.043	$0.502^{\pm.003}$	$0.698^{\pm.002}$	$0.798^{\pm.002}$	$0.304^{\pm.012}$	$3.012^{\pm.007}$	$9.607^{\pm.066}$	$2.259^{\pm.092}$
EMDM [104]	ECCV 2024	0.050	$0.498^{\pm.007}$	$0.684^{\pm.006}$	$0.786^{\pm.006}$	$0.112^{\pm.019}$	$3.110^{\pm.027}$	$\underline{9.551}^{\pm.078}$	$1.641^{\pm.078}$
Motion Mamba [99]	ECCV 2024	0.058	$0.502^{\pm.003}$	$0.693^{\pm.002}$	$0.792^{\pm.002}$	$0.281^{\pm.009}$	$3.060^{\pm.058}$	$9.871^{\pm.084}$	$2.294^{\pm.058}$
StableMoFusion [38]	MM 2024	0.499	$0.553^{\pm.003}$	$0.748^{\pm.002}$	$0.841^{\pm.002}$	$0.098^{\pm.003}$	-	$9.748^{\pm.092}$	$1.774^{\pm.051}$
MotionLCM-V2 [16] (1-step)	Preprint 2024	0.031	$0.546^{\pm.003}$	$0.743^{\pm.002}$	$0.837^{\pm.002}$	$0.072^{\pm.003}$	$2.767^{\pm.007}$	$9.577^{\pm.070}$	$1.858^{\pm.056}$
MotionLCM-V2 [16] (2-step)	Preprint 2024	0.038	$0.551^{\pm.003}$	$0.745^{\pm.002}$	$0.836^{\pm.002}$	$0.049^{\pm.003}$	$2.765^{\pm.008}$	$9.584^{\pm.066}$	$1.833^{\pm.052}$
MotionLCM-V2 [16] (4-step)	Preprint 2024	0.050	$0.553^{\pm.003}$	$0.746^{\pm.002}$	$0.837^{\pm.002}$	$0.056^{\pm.003}$	$2.773^{\pm.009}$	$9.598^{\pm.067}$	$1.758^{\pm.056}$
MMDM-t [8]	Preprint 2024	-	$0.464^{\pm.006}$	$0.654^{\pm.007}$	$0.754^{\pm.005}$	$0.319^{\pm.026}$	$3.288^{\pm.023}$	$9.299^{\pm.064}$	$2.741^{\pm.112}$
MMDM-b [8]	Preprint 2024	-	$0.435^{\pm.006}$	$0.627^{\pm.006}$	$0.733^{\pm.007}$	$0.285^{\pm.032}$	$3.363^{\pm.029}$	$9.398^{\pm.088}$	$2.701^{\pm.083}$
FTMoMamba [47]	Preprint 2024	-	$0.489^{\pm.003}$	$0.680^{\pm.002}$	$0.777^{\pm.002}$	$0.181^{\pm.009}$	$3.151^{\pm.009}$	$9.789^{\pm.085}$	$2.277^{\pm.099}$
Light-T2M [89]	AAAI 2025	0.151	$0.511^{\pm.003}$	$0.699^{\pm.002}$	$0.795^{\pm.002}$	$0.040^{\pm.002}$	$3.002^{\pm.008}$	-	$1.670^{\pm.061}$
Free-MDM [6]	Preprint 2025	0.045	$0.466^{\pm.008}$	$0.657^{\pm.007}$	$0.757^{\pm.005}$	$0.256^{\pm.045}$	-	$9.666^{\pm.080}$	-
Free-StableMoFusion [6]	Preprint 2025	0.036	$0.520^{\pm.013}$	$0.707^{\pm.003}$	$0.803^{\pm.006}$	$0.051^{\pm.002}$	-	$\mathbf{9.480}^{\pm.005}$	-
MotionPCM [41] (1-step)	Preprint 2025	0.031	$0.560^{\pm.002}$	$0.752^{\pm.003}$	$0.844^{\pm.002}$	$0.044^{\pm.003}$	$\underline{2.711}^{\pm.008}$	$9.559^{\pm.081}$	$1.772^{\pm.067}$
MotionPCM [41] (2-step)	Preprint 2025	0.036	$0.555^{\pm.002}$	$0.749^{\pm.002}$	$0.839^{\pm.002}$	$0.033^{\pm.002}$	$2.739^{\pm.007}$	$9.618^{\pm.088}$	$1.760^{\pm.068}$
MotionPCM [41] (4-step)	Preprint 2025	0.045	$0.559^{\pm.003}$	$0.752^{\pm.003}$	$0.842^{\pm.002}$	$\underline{0.030}^{\pm.002}$	$2.716^{\pm.008}$	$9.575^{\pm.082}$	$1.714^{\pm.062}$
Fg-T2M++ [74]	IJCV 2025	-	$0.513^{\pm.002}$	$0.702^{\pm.002}$	$0.801^{\pm.003}$	$0.089^{\pm.004}$	$2.925^{\pm.007}$	$9.223^{\pm.114}$	$2.625^{\pm.084}$
BioMoDiffuse [43]	Preprint 2025	-	$0.547^{\pm.003}$	$0.743^{\pm.002}$	$0.835^{\pm.002}$	$0.071^{\pm.003}$	$2.784^{\pm.008}$	$9.567^{\pm.086}$	$1.919^{\pm.063}$
HiSTF Mamba [90] (10-step)	Preprint 2025	0.690	$0.488^{\pm.005}$	$0.685^{\pm.004}$	$0.784^{\pm.005}$	$0.189^{\pm.018}$	$3.101^{\pm.022}$	$9.712^{\pm.090}$	$2.529^{\pm.044}$
HiSTF Mamba [90] (15-step)	Preprint 2025	-	$0.504^{\pm.005}$	$0.699^{\pm.005}$	$0.798^{\pm.005}$	$0.249^{\pm.023}$	$3.053^{\pm.022}$	$9.383^{\pm.091}$	$2.276^{\pm.036}$
ACMo [76]	Preprint 2025	-	$0.493^{\pm.002}$	$0.698^{\pm.003}$	$0.795^{\pm.002}$	$0.102^{\pm.003}$	$2.973^{\pm.006}$	$9.749^{\pm.082}$	$2.614^{\pm.100}$
FlashMo (Ours)	-	**0.027**	$\underline{0.562}^{\pm.004}$	$\underline{0.754}^{\pm.005}$	$\underline{0.847}^{\pm.005}$	$0.041^{\pm.002}$	$\underline{2.711}^{\pm.006}$	$9.614^{\pm.056}$	$2.812^{\pm.046}$
FlashMo w/ pretrain (Ours)	-	**0.027**	$\mathbf{0.568}^{\pm.005}$	$\mathbf{0.761}^{\pm.002}$	$\mathbf{0.851}^{\pm.003}$	$\mathbf{0.029}^{\pm.002}$	$\mathbf{2.703}^{\pm.005}$	$9.601^{\pm.073}$	$\underline{2.851}^{\pm.069}$



Section 1

There are two major questions for Section 1.
The first question requests you to overall qualify the motions provided, and for the second question please choose the best motion from a set of motions.

Section 1.1

The sim appears to walk forward bend slightly grabbing an object with their left hand.

Section1.1.1

A person is spinning with is arms spread out and then he falls over.

Section1.1.2

He starts to dance a lot.

Section1.1.3

Question 1

Please assess the overall quality of the provided motion generations. Specifically, determine whether they exhibit noticeable jitter, sliding, or unrealistic movement, and select the statement that best reflects your evaluation.

- The motion generation is completely unusable.
- The visualization is acceptable, but there are several aspects that need improvement.
- The visualization is strong, highlighting the potential impact and promising future for applications.

Question2

Please evaluate the average diversity of the motion generations provided and select the statement that best reflects your assessment

- The motions lack diversity, showing repetitive and uniform patterns.
- The motions exhibit moderate diversity, though certain elements are still predictable.
- The motions are highly diverse, showcasing a wide range of movement patterns and creative variations.

Question3

Please evaluate the overall degree of match between the texts and their corresponding motions provided, and select the option that best reflects your assessment:

- There is little to no alignment between the texts and motions.
- The texts and motions align fairly well, though some details inconsistencies remain.
- The texts and motions are well-aligned, accurately reflecting each other.

Section 1.2

a

Section1.2.1

b

section1.2.2

c

section1.2.3

Question 4

Which video you believe that has the best motion quality?

- a
- b
- c

Question 5

Which video you believe that has the highest diversity?

- a
- b
- c

Question 6

Which motion do you believe best aligns with or reflects the text: "Person walks forwards straight while stumbling"?

- a
- b
- c

Figure 7: **User study Google Forms.** The User Interface (UI) used in our user study.

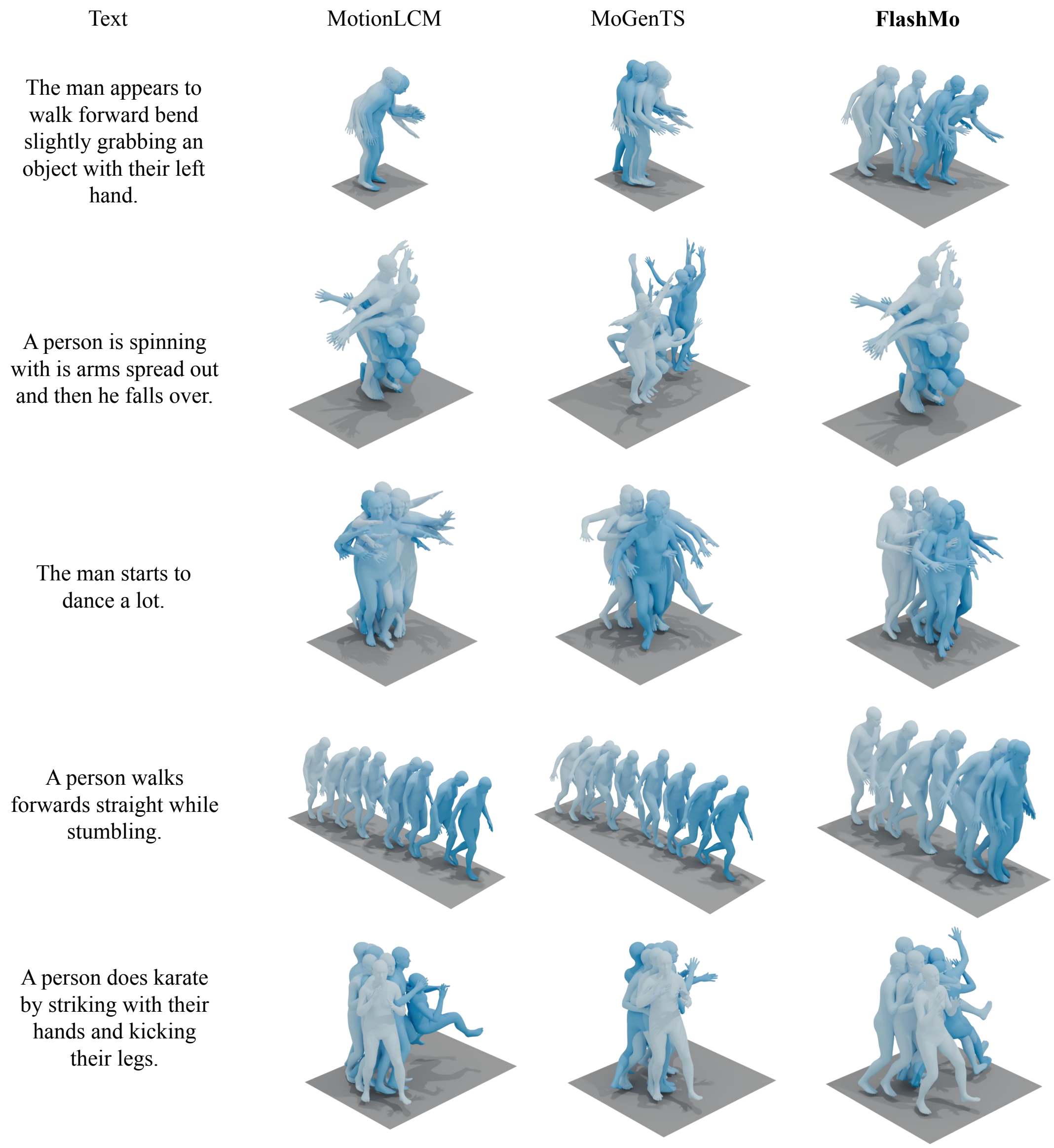


Figure 8: **Qualitative evaluation on HumanML3D [31] test set.** We qualitatively compared the visualizations generated by our method with those produced by MotionLCM [17] and MoGenTS [85].

Table 6: Comparison of text-to-motion generation on KIT-ML [61] dataset. $\rightarrow$ indicates the closer to real data, the better. **Bold** and $\underline{\text{underline}}$ indicate best and second best results. Efficient motion diffusion models are highlighted in blue.

Method	Venue	R-Precision ↑			FID ↓	MM Dist ↓	Diversity →	MModality ↑
		Top-1	Top-2	Top-3				
Real	-	$0.424^{\pm.005}$	$0.649^{\pm.006}$	$0.779^{\pm.006}$	$0.031^{\pm.004}$	$2.788^{\pm.012}$	$11.08^{\pm.097}$	-
TM2T [32]	ECCV 2022	$0.280^{\pm.005}$	$0.463^{\pm.006}$	$0.587^{\pm.005}$	$3.599^{\pm.153}$	$4.591^{\pm.026}$	$9.473^{\pm.117}$	$3.292^{\pm.081}$
T2M-GPT [92]	CVPR 2023	$0.416^{\pm.006}$	$0.627^{\pm.006}$	$0.745^{\pm.006}$	$0.514^{\pm.029}$	$3.007^{\pm.023}$	$10.92^{\pm.108}$	$1.570^{\pm.039}$
CoMo [39]	ECCV 2024	$0.422^{\pm.009}$	$0.638^{\pm.007}$	$0.765^{\pm.011}$	$0.332^{\pm.045}$	$2.873^{\pm.021}$	$10.95^{\pm.196}$	$1.249^{\pm.008}$
MMM [60]	CVPR 2024	$0.404^{\pm.005}$	$0.621^{\pm.005}$	$0.744^{\pm.004}$	$0.316^{\pm.028}$	$2.977^{\pm.019}$	$10.91^{\pm.101}$	$1.232^{\pm.039}$
MoMask [30]	CVPR 2024	$0.433^{\pm.007}$	$0.656^{\pm.005}$	$0.781^{\pm.005}$	$0.204^{\pm.011}$	$2.779^{\pm.022}$	-	$1.131^{\pm.043}$
BAMM [59]	ECCV 2024	$0.438^{\pm.009}$	$0.661^{\pm.009}$	$0.788^{\pm.005}$	$0.183^{\pm.013}$	$2.723^{\pm.026}$	$11.01^{\pm.094}$	$1.609^{\pm.065}$
MoGenTS [85]	NeurIPS 2024	$0.445^{\pm.006}$	$\underline{0.671}^{\pm.006}$	$0.797^{\pm.005}$	$0.143^{\pm.004}$	$2.711^{\pm.024}$	$10.92^{\pm.090}$	-
MDM [70]	ICLR 2023	$0.164^{\pm.004}$	$0.291^{\pm.004}$	$0.396^{\pm.004}$	$0.497^{\pm.021}$	$9.191^{\pm.022}$	$10.85^{\pm.109}$	$1.907^{\pm.214}$
MotionDiffuse [93]	TPAMI 2024	$0.417^{\pm.004}$	$0.621^{\pm.004}$	$0.739^{\pm.004}$	$1.954^{\pm.062}$	$2.958^{\pm.005}$	$\mathbf{11.10}^{\pm.143}$	$0.730^{\pm.013}$
MLD [7]	CVPR 2023	$0.390^{\pm.008}$	$0.609^{\pm.008}$	$0.734^{\pm.007}$	$0.404^{\pm.027}$	$3.204^{\pm.027}$	$10.80^{\pm.117}$	$2.192^{\pm.071}$
ReMoDiffuse [94]	ICCV 2023	$0.427^{\pm.014}$	$0.641^{\pm.004}$	$0.765^{\pm.055}$	$0.155^{\pm.006}$	$2.814^{\pm.012}$	$10.80^{\pm.105}$	$1.239^{\pm.028}$
M2DM [44]	ICCV 2023	$0.405^{\pm.003}$	$0.629^{\pm.005}$	$0.739^{\pm.004}$	$0.502^{\pm.049}$	$3.012^{\pm.015}$	$11.38^{\pm.079}$	$3.273^{\pm.045}$
Fg-T2M [73]	ICCV 2023	$0.418^{\pm.005}$	$0.626^{\pm.004}$	$0.745^{\pm.004}$	$0.571^{\pm.047}$	$3.114^{\pm.015}$	$10.93^{\pm.083}$	$1.019^{\pm.029}$
FineMoGen [95]	NeurIPS 2023	$0.432^{\pm.006}$	$0.649^{\pm.005}$	$0.772^{\pm.006}$	$0.178^{\pm.007}$	$2.869^{\pm.014}$	$10.85^{\pm.115}$	$1.877^{\pm.093}$
GraphMotion [42] (50-step)	NeurIPS 2023	$0.417^{\pm.008}$	$0.635^{\pm.006}$	$0.755^{\pm.004}$	$0.262^{\pm.021}$	$3.085^{\pm.031}$	$11.21^{\pm.106}$	$3.568^{\pm.132}$
GraphMotion [42] (150-step)	NeurIPS 2023	$0.429^{\pm.007}$	$0.648^{\pm.006}$	$0.769^{\pm.006}$	$0.313^{\pm.013}$	$3.076^{\pm.022}$	$11.12^{\pm.135}$	$\mathbf{3.627}^{\pm.113}$
B2A-HDM [80]	AAAI 2024	$0.436^{\pm.006}$	$0.653^{\pm.006}$	$0.773^{\pm.005}$	$0.367^{\pm.020}$	$2.946^{\pm.024}$	$10.86^{\pm.124}$	$1.291^{\pm.047}$
M2D2M [12]	ECCV 2024	-	-	$0.753^{\pm.006}$	$0.378^{\pm.023}$	$3.012^{\pm.021}$	$10.71^{\pm.121}$	$2.061^{\pm.067}$
EMDM [104]	ECCV 2024	$0.443^{\pm.006}$	$0.660^{\pm.006}$	$0.780^{\pm.005}$	$0.261^{\pm.014}$	$2.874^{\pm.015}$	$10.96^{\pm.093}$	$1.343^{\pm.089}$
Motion Mamba [99]	ECCV 2024	$0.419^{\pm.006}$	$0.645^{\pm.005}$	$0.765^{\pm.006}$	$0.307^{\pm.041}$	$3.021^{\pm.025}$	$11.02^{\pm.098}$	$1.678^{\pm.064}$
StableMoFusion [38]	MM 2024	$0.445^{\pm.006}$	$0.660^{\pm.005}$	$0.782^{\pm.004}$	$0.258^{\pm.029}$	-	$10.94^{\pm.077}$	$1.362^{\pm.062}$
MMDM-t [8]	Preprint 2024	$0.432^{\pm.006}$	$0.643^{\pm.007}$	$0.760^{\pm.006}$	$0.237^{\pm.013}$	$2.938^{\pm.025}$	$10.84^{\pm.125}$	$1.457^{\pm.129}$
MMDM-b [8]	Preprint 2024	$0.386^{\pm.007}$	$0.603^{\pm.006}$	$0.729^{\pm.006}$	$0.408^{\pm.022}$	$3.215^{\pm.026}$	$10.53^{\pm.100}$	$2.261^{\pm.144}$
Light-T2M [89]	AAAI 2025	$0.444^{\pm.006}$	$0.670^{\pm.007}$	$0.794^{\pm.005}$	$0.161^{\pm.009}$	$2.746^{\pm.016}$	-	$1.005^{\pm.036}$
Free-MDM [6]	Preprint 2025	$0.382^{\pm.006}$	$0.587^{\pm.006}$	$0.707^{\pm.007}$	$0.401^{\pm.033}$	-	$10.73^{\pm.102}$	-
Free-StableMoFusion [6]	Preprint 2025	$0.431^{\pm.003}$	$\underline{0.671}^{\pm.001}$	$0.789^{\pm.002}$	$0.155^{\pm.079}$	-	$10.90^{\pm.045}$	-
MotionPCM [41] (1-step)	Preprint 2025	$0.433^{\pm.007}$	$0.654^{\pm.007}$	$0.781^{\pm.008}$	$0.355^{\pm.011}$	$2.820^{\pm.022}$	$10.78^{\pm.078}$	$1.337^{\pm.047}$
MotionPCM [41] (2-step)	Preprint 2025	$0.437^{\pm.005}$	$0.664^{\pm.005}$	$0.787^{\pm.006}$	$0.294^{\pm.011}$	$2.844^{\pm.018}$	$10.83^{\pm.094}$	$1.254^{\pm.050}$
MotionPCM [41] (4-step)	Preprint 2025	$0.443^{\pm.005}$	$0.664^{\pm.004}$	$0.789^{\pm.005}$	$0.336^{\pm.013}$	$2.881^{\pm.023}$	$10.76^{\pm.096}$	$1.258^{\pm.056}$
Fg-T2M++ [74]	IJCV 2025	$0.442^{\pm.006}$	$0.657^{\pm.005}$	$0.781^{\pm.004}$	$\underline{0.135}^{\pm.004}$	$2.696^{\pm.011}$	$10.99^{\pm.105}$	$1.255^{\pm.078}$
BioMoDiffuse [43]	Preprint 2025	$0.448^{\pm.008}$	$0.666^{\pm.005}$	$0.788^{\pm.005}$	$0.211^{\pm.101}$	$2.772^{\pm.017}$	$\underline{11.11}^{\pm.094}$	$1.380^{\pm.050}$
HiSTF Mamba [90] (10-step)	Preprint 2025	$0.437^{\pm.006}$	$0.651^{\pm.006}$	$0.772^{\pm.006}$	$0.289^{\pm.021}$	$2.846^{\pm.018}$	$10.92^{\pm.096}$	$1.512^{\pm.088}$
HiSTF Mamba [90] (15-step)	Preprint 2025	$0.440^{\pm.006}$	$0.657^{\pm.006}$	$0.774^{\pm.006}$	$0.293^{\pm.017}$	$2.819^{\pm.015}$	$10.93^{\pm.099}$	$1.347^{\pm.056}$
FlashMo (Ours)	-	$\underline{0.449}^{\pm.002}$	$0.670^{\pm.004}$	$\underline{0.799}^{\pm.002}$	$0.152^{\pm.004}$	$\underline{2.709}^{\pm.005}$	$10.64^{\pm.074}$	$3.287^{\pm.042}$
FlashMo w/ pretrain (Ours)	-	$\mathbf{0.453}^{\pm.001}$	$\mathbf{0.679}^{\pm.004}$	$\mathbf{0.807}^{\pm.003}$	$\mathbf{0.132}^{\pm.005}$	$\mathbf{2.701}^{\pm.005}$	$10.79^{\pm.093}$	$\underline{3.591}^{\pm.070}$

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: Yes — the abstract and introduction clearly reflect the paper's contributions in efficiency, scalability, and interpolant design, which are consistently supported throughout the paper.

   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: Yes, in last section of main paper.

   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: While not a purely theoretical paper, it builds on established theoretical foundations from prior work, and all relevant assumptions are supported or cited accordingly.

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: Yes — the paper provides sufficient implementation details, model configurations, and evaluation settings to reproduce the main experimental results that support its core claims.

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: [No]

   Justification: While the method is easy to implement and the authors intend to release code and data after acceptance, they are not yet publicly available and require institutional approval for open-sourcing code and model weights.

   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: Yes — the paper specifies all necessary training and testing details, including hyperparameters, optimizer settings, and experimental configurations.

   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: Yes — the paper reports error bars based on multiple runs, providing appropriate statistical information to support the experimental results.

   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: Yes — the paper provides sufficient information on computational resources, including GPU type, memory, and execution settings, to reproduce the experiments.

   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: Yes — the research fully conforms to the NeurIPS Code of Ethics, with no ethical concerns identified in methodology, data usage, or reporting.

   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: [Yes]

    Justification: Yes — the paper discusses both positive and negative societal impacts, including the benefits for digital avatars and related applications, as outlined in the appendix.

    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: The paper does not involve high-risk models or data, and thus no specific safeguards are necessary.

    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: Yes — all external assets used in the paper are properly credited, with licenses and terms of use clearly respected.

    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: [Yes]

    Justification: Yes — the new assets introduced in the paper are well documented, including anomalous visualizations provided in the supplemental ZIP file.

    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 any crowdsourcing or human subjects, so this requirement does not apply.

    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 any crowdsourcing or human subjects, so this requirement does not apply.

    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: The paper does not use LLMs as a core component of the methodology; they are only used for grammar checking, which does not require declaration.

    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.