MOTIONSPACEFLOW: REPRESENTATION-AWARE FLOW MATCHING IN DIRECT MOTION SPACE

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent advances in diffusion and flow models have substantially improved text-driven human motion generation. Yet most methods generate in low-dimensional, temporally downsampled latent spaces learned primarily for reconstruction, a bottleneck that can limit generation quality and preclude direct manipulation of individual frames and joints. We introduce MotionSpaceFlow (MSFLOW), a representation-aware flow-matching framework that predicts clean motion directly in continuous motion space without a learned encoder or decoder. To account for the anisotropic structure of direct motion representations, we propose representation-aware noise scaling and show how the initial Gaussian source scale governs the covariance of intermediate probability-path marginals. We further introduce a Representation-Aware Multimodal Diffusion Transformer (RA-MMDiT), which jointly updates token-level language and full-resolution motion features through joint attention while adapting temporal information flow to the motion representation: causal attention for incremental features defined by frame-to-frame changes, and bidirectional attention for global features such as absolute joint coordinates. Across different datasets and motion representations, MSFLOW achieves state-of-the-art text-to-motion performance. Its global representation variant additionally enables zero-shot, inference-time control over any joint or frame through projection sampling without control-conditioned training, delivering leading motion quality with exact constraint satisfaction.

1 INTRODUCTION

Synthesizing human motion from natural language requires translating a high-level description into a coherent physical trajectory. Diffusion and flow models (Ho et al., 2020; Song et al., 2021; Rombach et al., 2022; Lipman et al., 2023) address this one-to-many problem by transforming Gaussian noise into complete motion sequences. Most recent methods generate in low-dimensional, temporally downsampled spaces produced by variational autoencoders (VAEs) (Kingma & Welling, 2014; Chen et al., 2023; Dai et al., 2024; Hong et al., 2025), reducing sequence length and generation cost. Others discretize motion with a VQ-VAE (van den Oord et al., 2017) and apply transformer-based generative models to the resulting token sequence (Guo et al., 2022b; Jiang et al., 2023; Guo et al., 2024). Although effective, both continuous and discrete latent approaches learn representations primarily through reconstruction. Generation quality can therefore be limited by autoencoder capacity and reconstruction fidelity (Yao et al., 2025; Zheng et al., 2026), while latent tokens lose direct correspondence to individual frames and joints, complicating fine-grained spatial control.

As illustrated in Fig. 1, we instead generate full-resolution motion variables with a Diffusion Transformer (DiT)-only model. Removing the motion encoder and decoder eliminates the reconstruction bottleneck and preserves the frame–joint structure required for inference-time spatial control. Direct-space generation, however, is not achieved by simply removing the autoencoder. Coordinate-wise z-normalization does not eliminate correlations across joints, feature types, and frames because motion representations combine continuous 3D quantities (*e.g.*, root-relative joint positions), continuous 6D rotations (*e.g.*, joint orientations), and categorical variables (*e.g.*, foot-contact labels). The flow must therefore traverse an anisotropic data distribution, adapt temporal dependencies to the represented variables, and align a short prompt with full-resolution motion.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

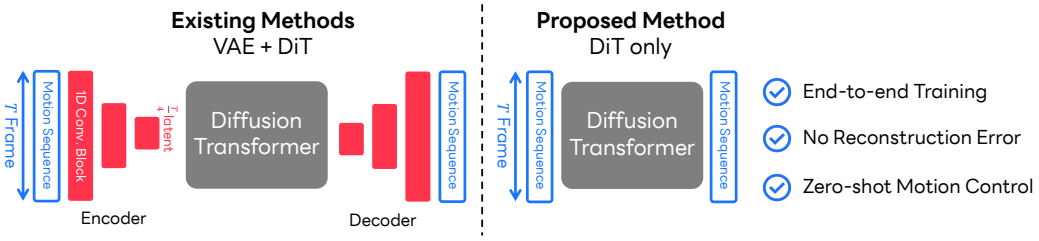

Figure 1: **Latent-space versus direct motion-space generation.** Existing methods encode a T-frame sequence into a shorter latent sequence, apply a diffusion transformer in the latent space, and decode the result back to motion. MSFLOW removes the motion encoder and decoder and applies a flow-matching transformer directly to the full sequence. This end-to-end design avoids autoencoder reconstruction error and preserves direct access to frames and joints for zero-shot spatial control.

To address these challenges, we introduce MotionSpaceFlow (MSFLOW), a representation-aware framework for flow matching in continuous motion space. Following JiT (Li & He, 2026), MSFLOW predicts the clean motion endpoint and analytically converts it into a velocity field for deterministic ODE sampling. Representation-aware noise scaling incorporates the Gaussian source scale into the probability path used for both training and sampling, treating it as a parameter of the probability path rather than merely as a control for inference-time randomness. Our analysis shows how this scale controls signal emergence and the anisotropy of intermediate path marginals.

Our Representation-Aware Multimodal DiT (RA-MMDiT) jointly updates token-level language and frame-level motion features to support fine-grained alignment across the full sequence. Crucially, its temporal attention graph is configured to match the underlying representation. The standard 263-dimensional HumanML3D representation (Guo et al., 2022a) includes root velocities whose accumulation recovers the global trajectory of the whole sequence. RA-MMDiT uses causal attention for these incremental features: all frames are updated in parallel at each ODE step, while each motion token attends only to its prefix and the complete text condition. For globally coupled absolute XYZ coordinates of all joints, it instead uses bidirectional attention, providing two-sided context and propagating spatial constraints before and after a controlled keyframe. This representation-aware attention lets RA-MMDiT handle both incremental and absolute motion representations.

On HumanML3D, the causal 263D variant achieves state-of-the-art text-to-motion performance, and our results reveal a clear contrast between the optimal attention designs for incremental 263D and absolute XYZ representations. The XYZ variant further enables any-joint, any-frame projection sampling without control-conditioned training, delivering leading motion quality and exact constraint satisfaction under the OmniControl (Xie et al., 2024) protocol. We also investigate the broader applicability of MSFLOW on SnapMoGen dataset (Guo et al., 2025). Our main contributions are:

- **Direct motion-space flow matching.** We introduce MSFLOW, which generates continuous motion through clean-endpoint prediction and ODE sampling without a learned motion encoder or decoder, and achieves state-of-the-art text-to-motion results.
- **Representation-aware noise scaling.** We characterize how Gaussian source scale controls signal emergence and the covariance conditioning of intermediate direct-space path marginals.
- **Representation-aware temporal modeling.** We introduce RA-MMDiT, which jointly updates token-level language and full-resolution motion features, and show that causal attention favors incremental representations whereas bidirectional attention is crucial for absolute coordinates.
- **Training-free spatial control.** We derive projection sampling for any-joint, any-frame constraints using a model trained only for text-to-motion generation, with exact constraint satisfaction.

2 Related Work

2.1 Motion Representations and Temporal Modeling for Motion Generation

Early neural approaches generate motion with recurrent or convolutional architectures (Yan et al., 2019; Zhao et al., 2020; Guo et al., 2020; Ghosh et al., 2021). Recent text-conditioned models operate on discrete tokens (Guo et al., 2022b; Zhang et al., 2023a; Guo et al., 2024; Pinyoanuntapong et al., 2024; Zou et al., 2024), continuous motion features (Tevet et al., 2023; Zhang et al., 2024;

Dabral et al., 2023), or autoencoded latent sequences (Petrovich et al., 2022; Chen et al., 2023; Dai et al., 2024; Hong et al., 2025). Discrete and autoencoded representations shorten the modeled sequence, but temporal downsampling limits information capacity and decoded fidelity remains bounded by autoencoder reconstruction quality. MSFLOW instead learns transport in continuous motion space, preserving frame- and joint-level access.

Autoregressive models enforce temporal causality by predicting future tokens or segments from past context. Discrete autoregressive methods (Zhang et al., 2023a; Jiang et al., 2023) can suffer from exposure bias (Schmidt, 2019), whereas diffusion-based variants (Meng et al., 2025b; Zhao et al., 2025; Xiao et al., 2025; Yu et al., 2026) combine iterative refinement with causal backbones. Bidirectional denoisers instead use complete temporal context. We compare matched causal and bidirectional variants to test whether the preferred dependency graph depends on the representation.

2.2 FLOW MATCHING AND SPATIAL CONTROL

Flow matching (Albergo & Vanden-Eijnden, 2023; Albergo et al., 2023; Lipman et al., 2023) learn vector fields that transport a simple source distribution to data, and scalable interpolant and Diffusion Transformers (Peebles & Xie, 2023; Ma et al., 2024) provide effective parameterizations of such fields. We adopt clean-endpoint prediction from JiT (Li & He, 2026), which allows the method to operate effectively in high-dimensional spaces, and study its interaction with Gaussian source scale and temporal attention in anisotropic motion representations. Some prior work selects or optimizes the initial noise of a fixed pretrained diffusion process to control generation (Mao et al., 2024; Wang et al., 2025; Zhou et al., 2025; Harrington et al., 2026; Ota et al., 2026).

In motion generation, prior methods impose spatial constraints or edits through control-conditioned training, guidance, test-time optimization, or large-scale controllable modeling (Karunratanakul et al., 2023; Athanasiou et al., 2024; Huang et al., 2024b; Petrovich et al., 2024; Xie et al., 2024; Liu et al., 2024; Rempe et al., 2026). ProjFlow (Watanabe et al., 2026) instead projects clean endpoint estimates during sampling without control-conditioned training. MSFLOW applies endpoint projection directly to XYZ motion, enabling arbitrary frame–joint–axis constraints.

2.3 MOTION–LANGUAGE CONDITIONING

Unlike motion–language alignment models (Tevet et al., 2022; Petrovich et al., 2023; Yu et al., 2024; Fujiwara et al., 2024; Yu et al., 2025), which jointly update motion and text features through contrastive learning, text-to-motion generators commonly condition on a frozen language encoder. In direct motion-space generation, preserving textual semantics is significantly more challenging because no temporally downsampled latent representation is available to bridge the modality gap. MSFLOW therefore refines token-level DistilBERT features (Sanh et al., 2019) as a function of flow time and couples them with motion tokens through multimodal attention, without an auxiliary motion–language alignment objective.

3 METHOD

MSFLOW generates motion directly in the data representation, without a learned motion encoder or decoder. Fig. 2 shows that the method consists of two components. First, a clean motion sequence, represented using either incremental 263D features or global XYZ coordinates, is interpolated with a scaled Gaussian source to form a noised motion state. Second, a Representation-Aware Multimodal Diffusion Transformer (RA-MMDiT) combines this state with token-level text features to predict the clean motion endpoint. The representation informs both the source scale and the temporal attention graph: causal for incremental features and bidirectional for absolute coordinates. The predicted endpoint is converted into an ODE velocity for the training objective and inference-time sampling. For XYZ motion, the endpoint can also be projected to satisfy the spatial constraints during inference.

3.1 PROBLEM SETTING AND DIRECT MOTION REPRESENTATIONS

Let $\mathbf{x}_1 \in \mathbb{R}^{T \times D}$ be a normalized motion sequence, where T is the number of frames and D is the feature dimension, and let $\mathbf{c}$ denote a text prompt, such as a natural language description of

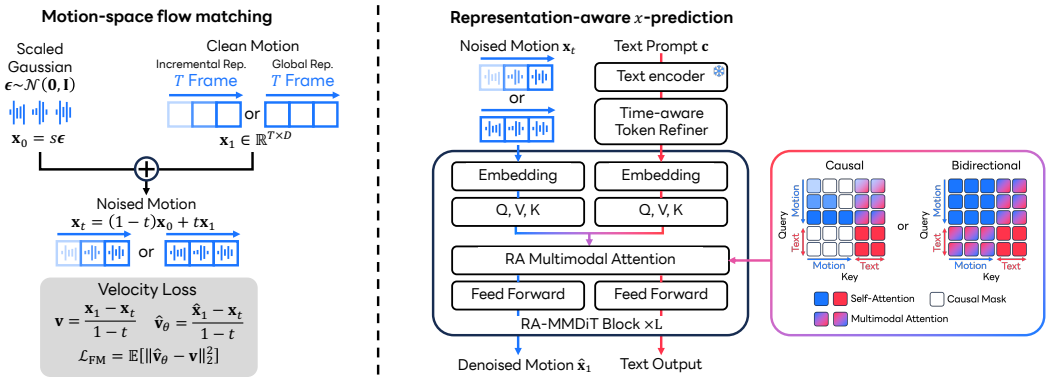

Figure 2: **Framework of MSFLOW. Left:** motion-space flow matching interpolates a scaled Gaussian source $\mathbf{x}_0 = s\boldsymbol{\epsilon}$ with a clean T-frame motion $\mathbf{x}_1$ in either an incremental or global representation. **Right:** RA-MMDiT jointly processes the noised motion $\mathbf{x}_t$ and time-refined text tokens to predict the clean endpoint $\hat{\mathbf{x}}_1$. Its attention mask follows the motion representation: a motion query attends only to its prefix and all text tokens in the causal variant, whereas the bidirectional variant uses the complete motion–text context.

the motion. We model the sequence as a continuous tensor at its original temporal resolution. We consider two commonly used representations with different temporal semantics:

- **Incremental 263D motion.** The standard HumanML3D representation (Guo et al., 2022a) contains root yaw and planar velocities, root height, root-relative joint positions, 6D joint rotations, local joint velocities, and foot-contact channels. In particular, the global root trajectory is obtained by accumulating frame-to-frame velocities.
- **Global XYZ motion.** Each frame contains the absolute 3D coordinates of 22 joints, resulting in $D = 66$. These variables are globally coupled across time but expose every frame and joint directly, which makes them suitable for spatial control.

In both cases, the flow acts on $\mathbf{x}_1$ itself; there is no learned map between a temporally compressed latent sequence and motion. The following bound isolates the reconstruction constraint that this design removes. Let P_1 be the data distribution over vectorized motions in $\mathbb{R}^{TD}$, and let $g : \mathcal{Z} \to \mathbb{R}^{TD}$ be a fixed decoder with range $\mathcal{R}_g = \{g(\mathbf{z}) : \mathbf{z} \in \mathcal{Z}\}$.

Proposition 1 (Autoencoder fidelity floor)**.** *For any generated distribution Q supported on $\mathcal{R}_g$,*

$$W_2^2(P_1, Q) \geq \mathbb{E}_{\mathbf{X} \sim P_1}\left[\operatorname{dist}^2(\mathbf{X}, \mathcal{R}_g)\right], \tag{1}$$

where W_2^2 denotes the squared 2-Wasserstein distance. Every decoded sample must lie in the decoder range, regardless of the latent generator's expressiveness. Because the lower bound can be zero when that range contains the data support, it does not claim that direct generation is universally better. Rather, it shows that direct motion-space flow eliminates this particular decoder-induced fidelity floor while retaining ordinary generative error.

3.2 CLEAN-MOTION PREDICTION AND FLOW SAMPLING

The left side of Fig. 2 defines the probability path. During training, we sample one time $t \sim \mathcal{U}(0, 1)$ per sequence and one Gaussian tensor $\boldsymbol{\epsilon}$ of the same shape as the motion, then form

$$\mathbf{x}_0 = s\boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \qquad \mathbf{x}_t = (1-t)\mathbf{x}_0 + t\mathbf{x}_1, \tag{2}$$

where $s > 0$ is the source scale. The same t is used for all valid frames so that $\mathbf{x}_t$ remains a coherent sequence. Rather than predicting velocity directly, RA-MMDiT predicts the clean endpoint,

$$\hat{\mathbf{x}}_1 = f_\theta(\mathbf{x}_t, t, \mathbf{c}). \tag{3}$$

This x-prediction parameterization follows JiT (Li & He, 2026) and provides the network a target with the same physical meaning at every noise level, where $\hat{\mathbf{x}}_1$ always retains the semantics and

units of the original motion representation. For the linear path, the exact sample-wise velocity and its endpoint-based estimate are calculated as follows:

$$\mathbf{v} = \mathbf{x}_1 - \mathbf{x}_0 = \frac{\mathbf{x}_1 - \mathbf{x}_t}{1-t}, \qquad \hat{\mathbf{v}}_\theta = \frac{\hat{\mathbf{x}}_1 - \mathbf{x}_t}{1-t}. \tag{4}$$

Thus, clean-motion prediction remains compatible with the flow ODE $d\mathbf{x}_t/dt = \hat{\mathbf{v}}_\theta$. The denominator in Eq. (4) becomes numerically unstable near $t = 1$. We therefore clip it as follows:

$$d_t = \max(1-t, \sigma_{\min}), \qquad \hat{\mathbf{v}}_\theta = \frac{\hat{\mathbf{x}}_1 - \mathbf{x}_t}{d_t}, \qquad \mathbf{v}^* = \frac{\mathbf{x}_1 - \mathbf{x}_t}{d_t}, \tag{5}$$

and optimize the valid-frame residual loss as follows:

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}\left[\frac{1}{|\Omega|}\sum_{i\in\Omega} \|\hat{\mathbf{v}}_{\theta,i} - \mathbf{v}_i^*\|_2^2\right], \tag{6}$$

where Ω is the set of non-padding frames. For $t \le 1 - \sigma_{\min}$, this is exactly the linear-path velocity objective, equivalently weighting endpoint error by $(1-t)^{-2}$. The terminal interval uses a bounded surrogate. At inference, we initialize the entire sequence from $\mathcal{N}(\mathbf{0}, s^2\mathbf{I})$ and integrate all frames jointly with a fixed-step Heun solver (Heun, 1900) and a final Euler update.

3.3 Representation-Aware Noise Scaling

Following standard practice (Guo et al., 2022a; Meng et al., 2025a), we normalize every motion channel before training. However, normalization alone does not make the motion distribution spherical. It rescales individual channels without removing correlations across joints, feature types, or frames. The resulting 263D and XYZ distributions can therefore have much more variation in some directions than in others, whereas the Gaussian source varies equally in every direction. We verify this structure empirically in Appendix E.2.

The source scale s in Eq. (2) is the standard deviation of the initial Gaussian noise. It is fixed during both training and sampling and should not be confused with an inference-time temperature. Increasing s has two intuitive effects: stronger noise causes the motion signal to appear later along the path, while its isotropic covariance makes intermediate states better conditioned.

To formalize these effects, let $\mathbf{X}_1$ be a vectorized clean motion for a fixed-length sequence. Assume $\mathbb{E}[\mathbf{X}_1] = \mathbf{0}$ and denote its covariance by $\mathbf{\Sigma}$, with smallest and largest eigenvalues $\lambda_{\min}$ and $\lambda_{\max}$.

Proposition 2 (Effects of source scale on the linear flow path)**.** *Let $\mathbf{X}_0 \sim \mathcal{N}(\mathbf{0}, s^2\mathbf{I})$ be independent of $\mathbf{X}_1$, and let*

$$\mathbf{X}_t = t\mathbf{X}_1 + (1-t)\mathbf{X}_0. \tag{7}$$

Then s controls when the motion signal emerges and how well conditioned the intermediate distribution is:

(i) For any unit eigenvector $\mathbf{u}$ of $\mathbf{\Sigma}$ with eigenvalue $\lambda_{\mathbf{u}}$,

$$\mathrm{SNR}_{\mathbf{u}}(t; s) = \frac{t^2\lambda_{\mathbf{u}}}{(1-t)^2 s^2}, \qquad t^*_{\mathbf{u}} = \frac{s}{s + \sqrt{\lambda_{\mathbf{u}}}}, \tag{8}$$

where $t^*_{\mathbf{u}}$ is the time at which the signal and noise variances are equal. A larger s moves this time closer to the clean endpoint $t = 1$, so the motion signal emerges later.

(ii) The intermediate covariance and its condition number for $t < 1$ are

$$\mathrm{Cov}(\mathbf{X}_t) = t^2\mathbf{\Sigma} + (1-t)^2 s^2 \mathbf{I}, \tag{9}$$

$$\kappa_t(s) = \frac{t^2\lambda_{\max} + (1-t)^2 s^2}{t^2\lambda_{\min} + (1-t)^2 s^2}. \tag{10}$$

The condition number $\kappa_t(s)$ is non-increasing in s. Thus, a larger isotropic source scale improves conditioning by reducing the relative disparity between high- and low-variance directions, while leaving their absolute eigenvalue gap unchanged.

Our ablations show that $s = 5$ works well for both the 263D and XYZ representations. We therefore use this value in our primary models.

3.4 Representation-Aware Multimodal Transformer

The right side of Fig. 2 shows RA-MMDiT. Each motion frame is projected to a 512-dimensional token. A frozen DistilBERT encoder (Sanh et al., 2019) produces 768-dimensional token-level language features. We use a two-layer, time-aware text module, which we call the Token Refiner, to project these features to the model width and adapt them to the current flow time. This makes the otherwise static language features responsive to the changing motion signal along the flow path. Rotary position embeddings (Su et al., 2024) encode motion order, adaptive layer normalization (Peebles & Xie, 2023) injects flow time, and the text condition is independently replaced by a null embedding with probability 0.1 to train classifier-free guidance.

The network contains eight RA-MMDiT blocks with four attention heads (Vaswani et al., 2017). Each block has modality-specific query, key, value, and feed-forward projections, while joint attention concatenates motion and text keys and values. This allows every motion frame to retrieve relevant words without first compressing the prompt to a single vector. The attention mask is the only architectural difference between the two representation variants:

- In the **causal** variant, motion query i attends to motion frames $1{:}i$ and all valid text tokens. Text queries attend only to text tokens, preventing future motion from reaching an earlier frame indirectly through the updated language stream.
- In the **bidirectional** variant, motion and text queries attend to the complete valid motion–text sequence. Padding tokens are masked in both variants.

The mask reflects each representation's structure. Latent models can handle the directionality of incremental 263D features implicitly through their encoder and decoder, whereas direct motion-space generation must model it explicitly. Because 263D root translation and yaw are integrated from frame-to-frame velocities, we use causal masking to encourage forward-consistent dependencies and limit reliance on bidirectional smoothing.

Absolute XYZ trajectories are globally coupled, so bidirectional context can propagate temporal and skeletal constraints. Causal attention still updates all frames jointly at each ODE step, but prevents future perturbations from affecting earlier frames. These filtering and smoothing properties motivate the causal 263D and bidirectional XYZ variants evaluated in Section 4.3; formal analyses appear in Appendix E.3. At sampling time, classifier-free guidance (Ho & Salimans, 2022) is applied to the predicted endpoints and the result is converted to a velocity using Eq. (5). Because the model predicts the target motion variables directly, the integrated endpoint requires no motion decoder.

3.5 Inference-Time Any-Joint, Any-Frame Control

Direct XYZ endpoints enable zero-shot spatial control at inference. The bidirectional XYZ model is trained on paired motions and text, without constraint masks or target coordinates. At inference, let $\mathbf{M} \in \{0,1\}^{T\times 66}$ select any valid frame–joint–axis entries and $\mathbf{y}$ contain their targets in normalized XYZ coordinates. At each solver step, we first project the predicted clean endpoint:

$$\hat{\mathbf{x}}_1^{\mathrm{proj}} = (\mathbf{1} - \mathbf{M}) \odot \hat{\mathbf{x}}_1 + \mathbf{M} \odot \mathbf{y}. \tag{11}$$

Simply overwriting the same entries of the noised state $\mathbf{x}_t$ with clean coordinates would mix two noise levels. Instead, we write the linear path as $\mathbf{x}_t = \alpha_t \mathbf{x}_1 + \sigma_t \mathbf{x}_0$, where $\alpha_t = t$ and $\sigma_t = 1 - t$, and recover the current source estimate $\hat{\mathbf{x}}_0 = (\mathbf{x}_t - \alpha_t \hat{\mathbf{x}}_1)/\sigma_t$ for $t < 1$. For the next solver time t', the projection sampler refreshes that estimate with a new source-scale noise ϵ_s and recomposes a path-consistent state:

$$\eta_{t'} = \mathrm{clip}\left(\frac{1 - \sigma_{t'}}{s^2}, 0, 1\right), \tag{12}$$

$$\widetilde{\mathbf{x}}_0 = \sqrt{1 - \eta_{t'}}\, \hat{\mathbf{x}}_0 + \sqrt{\eta_{t'}}\, \boldsymbol{\epsilon}_s, \quad \boldsymbol{\epsilon}_s \sim \mathcal{N}(\mathbf{0}, s^2\mathbf{I}), \tag{13}$$

$$\mathbf{x}_{t'} = \alpha_{t'} \hat{\mathbf{x}}_1^{\mathrm{proj}} + \sigma_{t'} \widetilde{\mathbf{x}}_0. \tag{14}$$

The injected noise uses the source scale of the training path, and a final hard projection removes residual numerical error at the endpoint. Following ProjFlow (Watanabe et al., 2026), our projection sampler also adopts its kinematics-aware metric and pseudo-observation construction. We refer readers to that paper for details. Neither $\mathbf{M}$ nor $\mathbf{y}$ appears in the training objective, so "training-free control" refers specifically to introducing spatial constraints only during inference.

Table 1: **Results of text-to-motion performance on HumanML3D.** We use the 67D evaluator following MARDM. MSFLOW (263D) denotes the incremental-263D model, while MSFLOW (XYZ) denotes the global-XYZ variant. The average is reported over 10 runs with 95% confidence intervals. **Bold** indicates the best result, and underline denotes the second-best result.

Method	Representation	R-Precision↑			FID↓	MM-Dist↓	MModality↑	CLIP↑
		Top 1	Top 2	Top 3				
T2M-GPT (Zhang et al., 2023a)	Discrete latent	$.470^{\pm.003}$	$.659^{\pm.002}$	$.758^{\pm.002}$	$.335^{\pm.003}$	$3.505^{\pm.017}$	$2.018^{\pm.053}$	$.607^{\pm.005}$
MMM (Pinyoanuntapong et al., 2024)		$.487^{\pm.003}$	$.683^{\pm.002}$	$.782^{\pm.002}$	$.132^{\pm.004}$	$3.359^{\pm.019}$	$2.241^{\pm.073}$	$.635^{\pm.003}$
MoMask (Guo et al., 2024)		$.490^{\pm.004}$	$.687^{\pm.003}$	$.786^{\pm.003}$	$.116^{\pm.006}$	$3.353^{\pm.010}$	$1.263^{\pm.079}$	$.637^{\pm.003}$
MLD (Chen et al., 2023)	Continuous latent	$.461^{\pm.004}$	$.651^{\pm.004}$	$.750^{\pm.003}$	$.431^{\pm.014}$	$3.445^{\pm.019}$	$\underline{3.506}^{\pm.031}$	$.615^{\pm.003}$
SALAD (Hong et al., 2025)		$.552^{\pm.003}$	$.748^{\pm.003}$	$.839^{\pm.002}$	$.124^{\pm.005}$	$2.990^{\pm.010}$	$1.833^{\pm.081}$	$.671^{\pm.001}$
MARDM (Meng et al., 2025b)		$.500^{\pm.004}$	$.695^{\pm.003}$	$.795^{\pm.003}$	$.114^{\pm.007}$	$3.270^{\pm.009}$	$2.231^{\pm.071}$	$.642^{\pm.002}$
ACMDM (Meng et al., 2025a)		$.522^{\pm.002}$	$.713^{\pm.002}$	$.807^{\pm.002}$	$.058^{\pm.004}$	$3.205^{\pm.008}$	$2.077^{\pm.083}$	$.652^{\pm.001}$
CMDM (Yu et al., 2026)		$.563^{\pm.004}$	$\underline{.759}^{\pm.003}$	$\underline{.849}^{\pm.002}$	$.078^{\pm.003}$	$2.920^{\pm.007}$	$1.827^{\pm.094}$	$\underline{.685}^{\pm.001}$
MDM (Tevet et al., 2023)	Raw motion	$.440^{\pm.007}$	$.636^{\pm.006}$	$.742^{\pm.004}$	$.518^{\pm.032}$	$3.640^{\pm.028}$	$\mathbf{3.604}^{\pm.031}$	$.578^{\pm.003}$
MotionDiffuse (Zhang et al., 2024)		$.450^{\pm.006}$	$.641^{\pm.005}$	$.753^{\pm.005}$	$.778^{\pm.035}$	$3.490^{\pm.023}$	$3.179^{\pm.046}$	$.606^{\pm.004}$
ReMoDiffuse (Zhang et al., 2023b)		$.468^{\pm.003}$	$.653^{\pm.003}$	$.754^{\pm.005}$	$.883^{\pm.021}$	$3.414^{\pm.020}$	$2.703^{\pm.154}$	$.621^{\pm.003}$
MSFLOW (263D)		$\mathbf{.571}^{\pm.005}$	$\mathbf{.764}^{\pm.003}$	$\mathbf{.853}^{\pm.003}$	$\underline{.046}^{\pm.004}$	$\mathbf{2.890}^{\pm.010}$	$1.450^{\pm.079}$	$\mathbf{.686}^{\pm.000}$
MSFLOW (XYZ)		$\underline{.566}^{\pm.004}$	$\underline{.759}^{\pm.003}$	$\underline{.849}^{\pm.003}$	$\mathbf{.038}^{\pm.004}$	$\underline{2.916}^{\pm.008}$	$1.518^{\pm.084}$	$.676^{\pm.001}$

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Dataset and representations. We conduct experiments on HumanML3D (Guo et al., 2022a), using sequences of at most 192 frames at 20 frames per second. The primary model generates normalized 263-dimensional HumanML3D features. The XYZ variant generates shared-axis-normalized positions of 22 joints, flattened to 66 values per frame. Following previous methods (Meng et al., 2025b;a; Yu et al., 2026), both variants are evaluated with the standard 67-dimensional root and rotation-invariant joint-position evaluator introduced in (Meng et al., 2025b). XYZ outputs are converted to this space through the HumanML3D post-processing pipeline. To test broader applicability, we additionally train MSFLOW on the 272D HumanML3D representation used by MotionStreamer (Xiao et al., 2025) and on the SnapMoGen dataset using its native 296D representation (Guo et al., 2025). Full protocols and results appear in Appendix A.5 and A.6, respectively.

Evaluation metrics. Following prior works, we report motion-embedding Fréchet distance (FID), R-Precision at Top 1–3, matching distance (MM-Dist), multimodality (MModality), and motion–language cosine similarity (CLIP score). Each generation result averages 10 stochastic evaluations.

Implementation details. The 68M-parameter transformer uses eight RA-MMDiT blocks, source scale $s = 5$, and 50-step flow sampling. Full training and inference details appear in Appendix C.

4.2 QUANTITATIVE TEXT-TO-MOTION RESULTS

Table 1 compares the incremental 263D and global XYZ variants of MSFLOW with existing methods using the same 67-dimensional evaluator. The global variant achieves the best FID of 0.038 ± 0.004, whereas the incremental variant attains the strongest point estimates for R-Precision, MM-Dist, and CLIP score. Relative to CMDM, the strongest prior method on most metrics, the global variant reduces FID from 0.078 to 0.038 (51.3%), while the incremental variant improves Top-1/2/3 R-Precision by $0.008/0.005/0.004$, reduces MM-Dist by 0.030, and increases CLIP by 0.001. Thus, we demonstrate that MSFLOW improves both fidelity and text alignment without a learned latent representation. Table 2 shows that MSFLOW also generalizes to SnapMoGen: it improves Top-1/3 R-Precision over CMDM from $.831/.958$ to $.910/.984$, while retaining a competitive FID of 16.342 and strong multimodality of 12.538. Complete metrics and baselines appear in Appendix A.6.

4.3 ANALYSIS OF REPRESENTATION-AWARE DESIGN CHOICES

Temporal attention depends on the motion representation. Table 3 (full version in Appendix A.3) shows that causal attention reduces 263D FID from 0.067 to 0.046, whereas bidirectional attention reduces XYZ FID from 1.563 to 0.038. This reversal supports a filtering–smoothing

Table 2: **Text-to-motion results on SnapMoGen.** Bold and underline mark the best and second-best generated results.

Method	Rep.	Top 1↑	Top 3↑	FID↓	MModality↑
GT	–	.940	.985	.001	–
T2M-GPT	Disc.	.618	.812	32.629	9.172
MoMask		.777	.927	17.404	8.183
MoMask^{++}		.802	.938	<u>15.061</u>	7.259
ScaleMoGen		.807	.941	16.350	9.399
StableMoFusion	Conti.	.679	.888	27.801	9.064
MARDM		.648	.856	26.348	9.883
MotionStreamer		.631	.836	30.023	7.543
CMDM		<u>.831</u>	<u>.958</u>	**14.451**	9.521
MDM	Raw	.503	.727	57.783	**13.412**
MSFLOW		**.910**	**.984**	16.342	<u>12.538</u>

Table 3: **Representation-aware design analysis on HumanML3D.** Bold marks the better result in each pair; gray marks the proposed settings.

Factor	Rep.	Configuration	Top 3↑	FID↓	MM-Dist↓
Attention	263D	Causal, $s = 5$	**.853**	**.046**	**2.890**
		Bidirectional, $s = 5$	.846	.067	2.973
	XYZ	Causal, $s = 5$	.800	1.563	3.243
		Bidirectional, $s = 5$	**.849**	**.038**	**2.916**
Scale	263D	Causal, $s = 1$	**.861**	.111	**2.851**
		Causal, $s = 5$	.853	**.046**	2.890
	XYZ	Bidirectional, $s = 1$	.841	.144	2.999
		Bidirectional, $s = 5$	**.849**	**.038**	**2.916**
Target	263D	Causal, x-pred.	**.853**	**.046**	**2.890**
		Causal, v-pred.	.851	.061	2.900
	XYZ	Bidirectional, x-pred.	**.849**	**.038**	**2.916**
		Bidirectional, v-pred.	.831	.249	3.008

account: causality biases incremental features toward forward-consistent evolution, while bidirectional context helps absolute coordinates maintain whole-trajectory consistency. We interpret this as a finite-model inductive bias rather than a universal advantage of restricted context; formal risk and dynamical analyses appear in Appendix E.3. Appendix E.4 analyzes the attention patterns for both models and provides attention-routing evidence for this interpretation.

Source scaling substantially affects FID. Increasing s from 1 to 5 reduces FID from 0.111 to 0.046 for causal 263D generation, with small trade-offs in retrieval performance and MM-Dist, while substantially improving every reported metric for bidirectional XYZ generation. Proposition 2 explains why: a larger s improves path isotropy, thereby mitigating anisotropy in motion space by delaying signal emergence. Appendix E.2 verifies the underlying correlation structure of each representation and quantifies the resulting midpoint SNR and correlation retention. Overall, these results show that the source scale should be selected for the representation rather than inherited from a VAE latent space, making it an important design choice for motion-space generation.

Clean-motion prediction improves distributional fidelity. Clean-motion prediction outperforms direct v-prediction on every reported metric in all four matched comparisons. FID improves from 0.061 to 0.046 for causal 263D generation and from 0.249 to 0.038 for bidirectional XYZ generation. These results show that clean-motion prediction consistently enhances distributional fidelity in high-dimensional motion spaces.

4.4 Inference-Time Spatial Control

Following the 67-dimensional setting used by ProjFlow (Watanabe et al., 2026), we adopt the OmniControl (Xie et al., 2024) protocol with six controlled joints: the pelvis, left foot, right foot, head, left wrist, and right wrist. The protocol evaluates five control densities, corresponding to 1, 2, 5, 49, and 196 keyframes, and averages each metric across these densities. We report FID, Top-3 R-Precision, diversity, foot-skating ratio, trajectory error, location error, and average control error. Table 4 shows that MSFLOW exactly satisfies all evaluated constraints without control-conditioned training. In the all-joints setting, it improves the zero-shot ProjFlow baseline from 0.097 to 0.061 FID and from 0.779 to 0.818 R-Precision@3. Per-joint results appear in Appendix B.

4.5 Qualitative Results

In Fig. 3, we show control over local limb placement, forward locomotion, circular motion, and a cartwheel. In each case, the controlled joints satisfy the constraints exactly, while the unconstrained motion remains coherent and aligned with the prompt. Appendix A.1 provides additional text-to-motion comparisons, and the supplementary videos visualize the generated motions.

Table 4: **Results of text-conditioned motion generation with spatial controls on HumanML3D.** The first block trains and evaluates on pelvis controls. The second trains on all joints and reports the average over separately controlled joints; complete per-joint results appear in Table 11. Bold and underline denote the best and second-best values, respectively.

Controlling Joint	Methods	Zero-shot?	FID↓	R-Precision Top 3	Diversity→	Foot Skating Ratio.↓	Traj. err.↓	Loc. err.↓	Avg. err.↓
	GT	-	0.000	0.795	10.455	-	0.000	0.000	0.000
Pelvis	MDM (Tevet et al., 2023)	✓	1.792	0.673	9.131	0.1019	0.4022	0.3076	0.5959
	PriorMDM (Shafir et al., 2024)	✗	0.393	0.707	9.847	0.0897	0.3457	0.2132	0.4417
	GMD (Karunratanakul et al., 2023)	✓	0.238	0.763	10.011	0.1009	0.0931	0.0321	0.1439
	OmniControl (Xie et al., 2024)	✗	0.081	0.789	10.323	<u>0.0547</u>	0.0387	0.0096	0.0338
	MotionLCM V2+CtrlNet (Dai et al., 2024)	✗	3.978	0.738	9.249	0.0901	0.1080	0.0581	0.1386
	MaskControl (Pinyoanuntapong et al., 2025)	✗	**0.066**	<u>0.799</u>	<u>10.474</u>	**0.0543**	**0.0000**	**0.0000**	0.0093
	ProjFlow (Watanabe et al., 2026)	✓	0.107	0.784	10.644	0.0629	**0.0000**	**0.0000**	**0.0000**
	MSFLOW	✓	<u>0.068</u>	**0.821**	**10.447**	0.0591	**0.0000**	**0.0000**	**0.0000**
All Joints (Average)	OmniControl (Xie et al., 2024)	✗	0.126	0.792	<u>10.276</u>	0.0608	0.0617	0.0107	0.0404
	MotionLCM V2+CtrlNet (Dai et al., 2024)	✗	4.504	0.715	9.230	0.1119	0.2740	0.1315	0.2464
	MaskControl (Pinyoanuntapong et al., 2025)	✗	<u>0.095</u>	<u>0.795</u>	10.159	**0.0545**	**0.0000**	**0.0000**	<u>0.0065</u>
	ProjFlow (Watanabe et al., 2026)	✓	0.097	0.779	10.651	0.0603	**0.0000**	**0.0000**	**0.0000**
	MSFLOW	✓	**0.061**	**0.818**	**10.593**	<u>0.0578</u>	**0.0000**	**0.0000**	**0.0000**

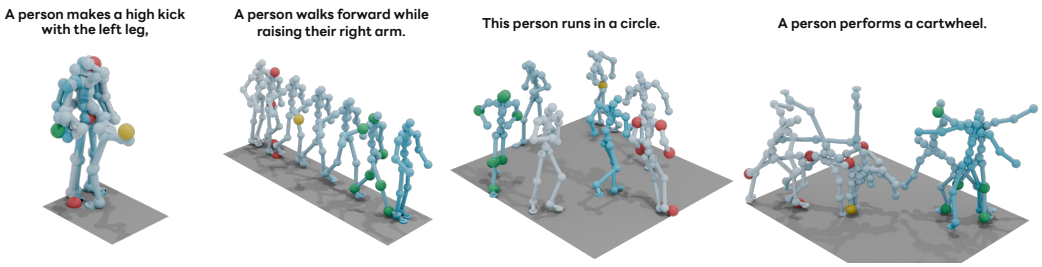

Figure 3: **Qualitative inference-time spatial control.** Colored markers indicate spatial constraints on selected joints and frames. MSFLOW follows the text prompts while satisfying these constraints.

5 LIMITATIONS

Although MSFLOW achieves state-of-the-art text-to-motion performance and exact spatial control, several limitations remain. First, our experiments cover only HumanML3D and SnapMoGen, and whether the observed interplay among representation, source scale, and temporal attention holds across other datasets, skeletons, and motion domains remains to be evaluated. Second, operating directly on full-resolution motion preserves frame-level access but requires processing longer token sequences than temporally compressed latent models, which may limit efficiency for very long motions. Hierarchical or streaming direct-space models could improve scalability while retaining fine-grained control. Finally, our projection sampler enforces linear equality constraints on joint coordinates, but does not explicitly guarantee physical feasibility or support non-linear constraints such as collision avoidance, contact, and joint limits. Extending direct-space generation with physical priors and more expressive constraint solvers is an important direction for future work.

6 CONCLUSION

In this paper, we presented MSFLOW, a representation-aware framework for flow matching directly in continuous motion space. MSFLOW, which predicts clean motion without a learned encoder or decoder, introduces representation-aware noise scaling for anisotropic direct-space probability paths and uses RA-MMDiT to adapt temporal information flow to the motion representation. Experiments on HumanML3D and SnapMoGen demonstrate state-of-the-art text-to-motion performance and show that causal attention is favored for incremental 263D features, whereas bidirectional attention is essential for absolute XYZ coordinates. The XYZ variant further enables training-free any-joint, any-frame projection with exact constraint satisfaction.

7 Reproducibility Statement

We support reproducibility through detailed descriptions of the datasets, motion representations, evaluation protocols, and stochastic evaluation procedure in Section 4 and Appendix A. Appendix C reports the model configurations, hyperparameters, training and sampling algorithms, compute requirements, and checkpoint-selection details for the principal variants; Appendix D provides accompanying sample code; and Appendix E contains complete theoretical derivations and additional empirical analyses.

8 AI Disclosure

In this work, we used generative AI tools to formulate mathematical claims, assist in writing proofs, implement methods, and assist with translation. The following tasks requiring disclosure were not applicable to this work: generating synthetic datasets; developing theoretical models or conceptual frameworks; providing critical ingredients for proving mathematical claims; proposing or refining hypotheses; designing or providing feedback on research methodology or experiments; cleaning or reformatting datasets; supporting qualitative or thematic data analysis; and interpreting results. Additionally, we used generative AI tools to draft parts of the paper and improve its readability. We reviewed all AI-assisted work and verified LLM-generated code, outputs, and edits for correctness and appropriateness before including them in the final manuscript. We take responsibility for the final content of this work, including any text, claims, or artifacts produced with the aid of generative AI.

References

Michael S Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *ICLR*, 2023.

Michael S Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *arXiv preprint arXiv:2303.08797*, 2023.

Nikos Athanasiou, Alpár Češke, Markos Diomataris, Michael J. Black, and Gül Varol. Motionfix: Text-driven 3d human motion editing. In *SIGGRAPH Asia*, 2024.

Yiyi Cai, Yuhan Wu, Kunhang Li, You Zhou, Bo Zheng, and Haiyang Liu. Flooddiffusion: Tailored diffusion forcing for streaming motion generation. In *CVPR*, 2026.

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 *CVPR*, 2023.

Xiaoyan Cong, Zekun Li, Zhiyang Dou, Hongyu Li, Omid Taheri, Chuan Guo, Abhay Mittal, Sizhe An, Taku Komura, Wojciech Matusik, Michael J. Black, and Srinath Sridhar. Umo: Unified in-context learning unlocks motion foundation model priors. *arXiv preprint arXiv:2603.15975*, 2026.

Rishabh Dabral, Muhammad Hamza Mughal, Vladislav Golyanik, and Christian Theobalt. Mofusion: A framework for denoising-diffusion-based motion synthesis. In *CVPR*, 2023.

Wenxun Dai, Ling-Hao Chen, Jingbo Wang, Jinpeng Liu, Bo Dai, and Yansong Tang. Motionlcm: Real-time controllable motion generation via latent consistency model. In *ECCV*, 2024.

Kent Fujiwara, Mikihiro Tanaka, and Qing Yu. Chronologically accurate retrieval for temporal grounding of motion-language models. In *ECCV*, 2024.

Anindita Ghosh, Noshaba Cheema, Cennet Oguz, Christian Theobalt, and Philipp Slusallek. Synthesis of compositional animations from textual descriptions. In *ICCV*, 2021.

Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2motion: Conditioned generation of 3d human motions. In *ACM MM*, 2020.

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 *CVPR*, 2022a.

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 *ECCV*, 2022b.

Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. Momask: Generative masked modeling of 3d human motions. In *CVPR*, 2024.

Chuan Guo, Inwoo Hwang, Jian Wang, and Bing Zhou. Snapmogen: Human motion generation from expressive texts. In *NeurIPS*, 2025.

Anne Harrington, A. Sophia Koepke, Shyamgopal Karthik, Trevor Darrell, and Alexei A. Efros. It's never too late: Noise optimization for collapse recovery in trained diffusion models. In *CVPR*, 2026.

Karl Heun. Neue methode zur approximativen integration der differentialgleichungen einer unabhängigen veränderlichen. *Z. Math. Phys*, 1900.

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

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Seokhyeon Hong, Chaelin Kim, Serin Yoon, Junghyun Nam, Sihun Cha, and Junyong Noh. Salad: Skeleton-aware latent diffusion for text-driven motion generation and editing. In *CVPR*, 2025.

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 *ACMMM*, 2024a.

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 *ECCV*, 2024b.

Inwoo Hwang, Hojun Jang, Bing Zhou, Jian Wang, Young Min Kim, and Chuan Guo. Scalemogen: Autoregressive next-scale prediction for human motion generation. In *ECCV*, 2026.

Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. Motiongpt: Human motion as a foreign language. In *NeurIPS*, 2023.

Korrawe Karunratanakul, Konpat Preechakul, Supasorn Suwajanakorn, and Siyu Tang. Guided motion diffusion for controllable human motion synthesis. In *ICCV*, 2023.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. In *CVPR*, 2026.

Yuan-Ming Li, Qize Yang, Nan Lei, Shenghao Fu, Ling-An Zeng, Jian-Fang Hu, Xihan Wei, and Wei-Shi Zheng. Irg-motionllm: Interleaving motion generation, assessment and refinement for text-to-motion generation. In *ECCV*, 2026.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *ICLR*, 2023.

Hanchao Liu, Xiaohang Zhan, Shaoli Huang, Tai-Jiang Mu, and Ying Shan. Programmable motion generation for open-set motion control tasks. In *CVPR*, 2024.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

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 *ECCV*, 2024.

Jiafeng Mao, Xueting Wang, and Kiyoharu Aizawa. The lottery ticket hypothesis in denoising: Towards semantic-driven initialization. In *ECCV*, 2024.

Zichong Meng, Zeyu Han, Xiaogang Peng, Yiming Xie, and Huaizu Jiang. Absolute coordinates make motion generation easy. *arXiv preprint arXiv:2505.19377*, 2025a.

Zichong Meng, Yiming Xie, Xiaogang Peng, Zeyu Han, and Huaizu Jiang. Rethinking diffusion for text-driven human motion generation. In *CVPR*, 2025b.

Sakuya Ota, Qing Yu, Kent Fujiwara, Satoshi Ikehata, and Ikuro Sato. Retrieving and refining winning noise tickets for diffusion-based motion generation. In *ECCV*, 2026.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023.

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

Mathis Petrovich, Michael J Black, and Gül Varol. Tmr: Text-to-motion retrieval using contrastive 3d human motion synthesis. In *ICCV*, 2023.

Mathis Petrovich, Or Litany, Umar Iqbal, Michael J. Black, Gül Varol, Xue Bin Peng, and Davis Rempe. Multi-track timeline control for text-driven 3d human motion generation. In *CVPR-W*, 2024.

Ekkasit Pinyoanuntapong, Pu Wang, Minwoo Lee, and Chen Chen. Mmm: Generative masked motion model. In *CVPR*, 2024.

Ekkasit Pinyoanuntapong, Muhammad Usama Saleem, Korrawe Karunratanakul, Pu Wang, Hongfei Xue, Chen Chen, Chuan Guo, Junli Cao, Jian Ren, and Sergey Tulyakov. Maskcontrol: Spatio-temporal control for masked motion synthesis. In *ICCV*, 2025.

Davis Rempe, Mathis Petrovich, Ye Yuan, Haotian Zhang, Xue Bin Peng, Yifeng Jiang, Tingwu Wang, Umar Iqbal, David Minor, Michael de Ruyter, Jiefeng Li, Chen Tessler, Edy Lim, Eugene Jeong, Sam Wu, Ehsan Hassani, Michael Huang, Jin-Bey Yu, Chaeyeon Chung, Lina Song, Olivier Dionne, Jan Kautz, Simon Yuen, and Sanja Fidler. Kimodo: Scaling controllable human motion generation. *arXiv preprint arXiv:2603.15546*, 2026.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In *NeurIPS-W*, 2019.

Florian Schmidt. Generalization in generation: A closer look at exposure bias. In *EMNLP-WNGT*, 2019.

Yoni Shafir, Guy Tevet, Roy Kapon, and Amit Haim Bermano. Human motion diffusion as a generative prior. In *ICLR*, 2024.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2021.

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

Guy Tevet, Brian Gordon, Amir Hertz, Amit H Bermano, and Daniel Cohen-Or. Motionclip: Exposing human motion generation to clip space. In *ECCV*, 2022.

Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H. Bermano. Human motion diffusion model. In *ICLR*, 2023.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *NeurIPS*, 2017.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

Ruoyu Wang, Huayang Huang, Ye Zhu, Olga Russakovsky, and Yu Wu. The silent assistant: Noisequery as implicit guidance for goal-driven image generation. In *ICCV*, 2025.

Akihisa Watanabe, Qing Yu, Edgar Simo-Serra, and Kent Fujiwara. Projflow: Projection sampling with flow matching for zero-shot exact spatial motion control. In *CVPR*, 2026.

Yuxin Wen, Qing Shuai, Di Kang, Jing Li, Cheng Wen, Yue Qian, Ningxin Jiao, Changhai Chen, Weijie Chen, Yiran Wang, Jinkun Guo, Dongyue An, Han Liu, Yanyu Tong, Chao Zhang, Qing Guo, Juan Chen, Qiao Zhang, Youyi Zhang, Zihao Yao, Cheng Zhang, Hong Duan, Xiaoping Wu, Qi Chen, Fei Cheng, Liang Dong, Peng He, Hao Zhang, Jiaxin Lin, Chao Zhang, Zhongyi Fan, Yifan Li, Zhichao Hu, Yuhong Liu, Linus, Jie Jiang, Xiaolong Li, and Linchao Bao. Hy-motion 1.0: Scaling flow matching models for text-to-motion generation. *arXiv preprint arXiv:2512.23464*, 2025.

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

Yiming Xie, Varun Jampani, Lei Zhong, Deqing Sun, and Huaizu Jiang. Omnicontrol: Control any joint at any time for human motion generation. In *ICLR*, 2024.

Sijie Yan, Zhizhong Li, Yuanjun Xiong, Huahan Yan, and Dahua Lin. Convolutional sequence generation for skeleton-based action synthesis. In *ICCV*, 2019.

Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In *CVPR*, 2025.

Qing Yu, Mikihiro Tanaka, and Kent Fujiwara. Exploring vision transformers for 3d human motion-language models with motion patches. In *CVPR*, 2024.

Qing Yu, Mikihiro Tanaka, and Kent Fujiwara. Remogpt: Part-level retrieval-augmented motion-language models. In *AAAI*, 2025.

Qing Yu, Akihisa Watanabe, and Kent Fujiwara. Causal motion diffusion models for autoregressive motion generation. In *CVPR*, 2026.

Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. Generating human motion from textual descriptions with discrete representations. In *CVPR*, 2023a.

Jianrong Zhang, Hehe Fan, and Yi Yang. Energymogen: Compositional human motion generation with energy-based diffusion model in latent space. In *CVPR*, 2025.

Mingyuan Zhang, Xinying Guo, Liang Pan, Zhongang Cai, Fangzhou Hong, Huirong Li, Lei Yang, and Ziwei Liu. Remodiffuse: Retrieval-augmented motion diffusion model. In *ICCV*, 2023b.

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 TPAMI*, 2024.

Kaifeng Zhao, Gen Li, and Siyu Tang. Dartcontrol: A diffusion-based autoregressive motion model for real-time text-driven motion control. In *ICLR*, 2025.

Rui Zhao, Hui Su, and Qiang Ji. Bayesian adversarial human motion synthesis. In *CVPR*, 2020.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. In *ICLR*, 2026.

Chongyang Zhong, Lei Hu, Zihao Zhang, and Shihong Xia. Attt2m: Text-driven human motion generation with multi-perspective attention mechanism. In *ICCV*, 2023.

Zikai Zhou, Shitong Shao, Lichen Bai, Shufei Zhang, Zhiqiang Xu, Bo Han, and Zeke Xie. Golden noise for diffusion models: A learning framework. In *ICCV*, 2025.

Qiran Zou, Shangyuan Yuan, Shian Du, Yu Wang, Chang Liu, Yi Xu, Jie Chen, and Xiangyang Ji. Parco: Part-coordinating text-to-motion synthesis. In *ECCV*, 2024.

SUPPLEMENTARY MATERIAL

This supplementary material provides additional text-to-motion and spatial-control results, implementation details for reproducing the principal 263D and XYZ models, accompanying sample code, and theoretical and empirical analyses supporting our design choices.

SECTION INDEX

A Additional Text-to-Motion Results p. 15
A.1 Qualitative Results p. 15
A.2 Model Capacity and Text–Motion Attention p. 15
A.3 Complete Representation-Aware Design Analysis p. 17
A.4 Evaluation with the Standard 263D Protocol p. 17
A.5 Training and Evaluation with the 272D MotionStreamer Protocol p. 17
A.6 Training and Evaluation on SnapMoGen p. 18
A.7 Compute Efficiency p. 19
B Detailed Spatial-Control Results p. 20
C Implementation and Reproducibility Details p. 20
C.1 Causal 263D Text-to-Motion Configuration p. 21
C.2 Bidirectional XYZ Text-to-Motion Configuration p. 21
C.3 Training and Sampling Algorithms p. 22
C.4 Bidirectional XYZ Endpoint-Projection Configuration p. 22
D Sample Code p. 22
E Theoretical Details and Proofs p. 23
E.1 Proofs of Main Propositions p. 23
E.2 Empirical Spatiotemporal Covariance and Source Scale p. 24
E.3 Why Attention Depends on the Representation p. 26
E.4 Empirical Attention Routing p. 27

A ADDITIONAL TEXT-TO-MOTION RESULTS

A.1 QUALITATIVE RESULTS

Fig. 4 compares ACMDM, SALAD, and the global-XYZ and incremental-263D variants of MSFLOW on six prompts that probe action count, limb specificity, directed reaching, trajectory reversal, lateral direction, and ground-contact transitions. The red annotations identify visible baseline failures, such as repeating an action, moving the wrong limb or direction, omitting a return trajectory, and floating feet. Both MSFLOW variants more consistently preserve the highlighted prompt details while producing coherent motion. Because static trajectory overlays cannot fully convey timing and transition quality, we recommend viewing the accompanying supplementary videos.

A.2 MODEL CAPACITY AND TEXT–MOTION ATTENTION

We ablate model capacity, the flow-time-conditioned Token Refiner, and the attention between text and motion while retaining the selected source scale, prediction target, and representation-specific temporal masks. Table 5 reports the complete results for both the causal 263D and bidirectional XYZ generators.

The four-block model is consistently under-capacity: relative to the selected eight-block model, its FID increases from .046 to .206 for 263D and from .038 to .098 for XYZ, accompanied by weaker retrieval and matching scores. Scaling from eight to 12 blocks does not yield a consistent gain. The 12-block 263D model has worse FID, R-Precision, and MM-Dist, while its XYZ counterpart is statistically similar in FID ($.039 \pm .003$ versus $.038 \pm .004$) and only improves the CLIP point estimate. We therefore retain the eight-block, width-512 backbone as the best capacity–quality trade-off.

The component ablation reveals a fidelity–retrieval trade-off. Removing the Token Refiner from the multimodal-attention model leaves 263D FID nearly unchanged but weakens its text-alignment metrics; for XYZ it also increases FID from .038 to .045. Replacing multimodal-attention RA-MMDiT blocks with conventional motion self-attention and text cross-attention is competitive for causal 263D when the Token Refiner is retained, but raises XYZ FID from .038 to .079. With-

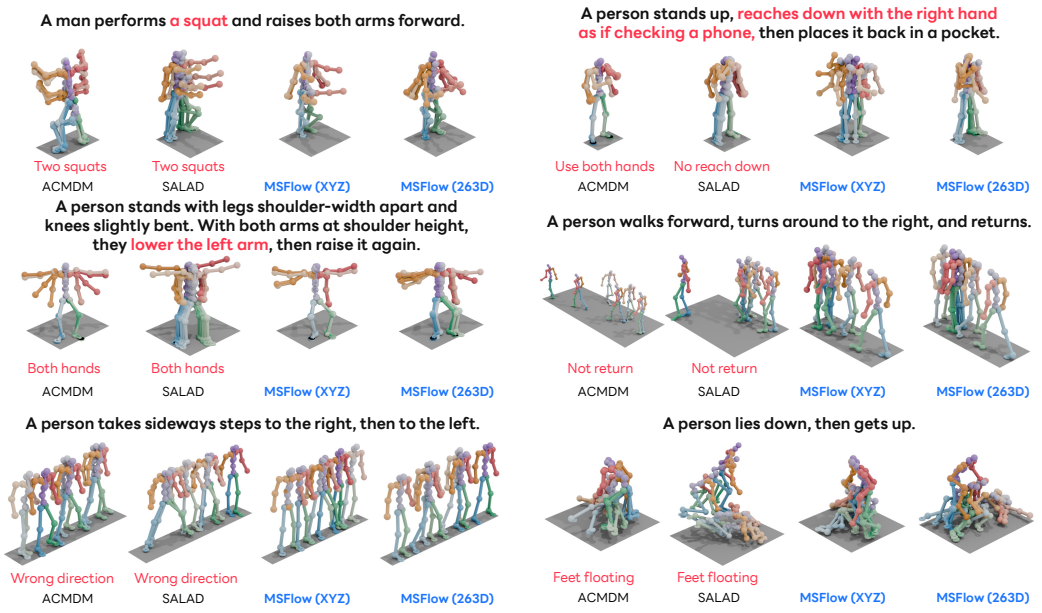

Figure 4: **Qualitative text-to-motion comparisons on HumanML3D.** We compare ACMDM, SALAD, and the global-XYZ and incremental-263D variants of MSFLOW across prompts requiring fine-grained action semantics, directional motion, and contact fidelity. Red annotations identify visible failure modes in the baselines; both MSFLOW variants more faithfully follow the highlighted prompt details. **For the clearest visualization of these dynamic behaviors, please view the accompanying supplementary videos.**

Table 5: **Model-capacity and text–motion attention ablations on HumanML3D.** All models use clean-motion prediction, source scale $s = 5$, classifier-free guidance weight 3, and the representation-aware temporal mask (causal for 263D and bidirectional for XYZ). MM denotes RA-MMDiT blocks that update motion and text through multimodal-attention; Cross denotes conventional motion self-attention followed by text cross-attention. Token Refiner denotes the two-layer, flow-time-conditioned module applied to text tokens. Each interval summarizes 10 stochastic evaluations. Bold denotes the best point estimate for each metric within each representation and ablation group. Gray cells mark the proposed setting.

Factor	Configuration				R-Precision↑			FID↓	MM-Dist↓	CLIP↑
	Rep.	Attention	Blocks/width	Token Refiner	Top 1	Top 2	Top 3			
Capacity	263D	MM	4/256	✓	$.555^{\pm.003}$	$.756^{\pm.003}$	$.848^{\pm.003}$	$.206^{\pm.008}$	$2.940^{\pm.011}$	$.678^{\pm.001}$
		MM	8/512	✓	$\mathbf{.571}^{\pm.005}$	$\mathbf{.764}^{\pm.003}$	$\mathbf{.853}^{\pm.003}$	$\mathbf{.046}^{\pm.004}$	$\mathbf{2.890}^{\pm.010}$	$\mathbf{.686}^{\pm.000}$
		MM	12/768	✓	$.561^{\pm.004}$	$.758^{\pm.003}$	$.847^{\pm.003}$	$.076^{\pm.006}$	$2.925^{\pm.012}$	$.681^{\pm.001}$
	XYZ	MM	4/256	✓	$.549^{\pm.004}$	$.750^{\pm.003}$	$.845^{\pm.003}$	$.098^{\pm.006}$	$3.000^{\pm.014}$	$.658^{\pm.001}$
		MM	8/512	✓	$\mathbf{.566}^{\pm.004}$	$.759^{\pm.003}$	$\mathbf{.849}^{\pm.003}$	$\mathbf{.038}^{\pm.004}$	$\mathbf{2.916}^{\pm.008}$	$.676^{\pm.001}$
		MM	12/768	✓	$.564^{\pm.003}$	$\mathbf{.760}^{\pm.003}$	$\mathbf{.849}^{\pm.002}$	$.039^{\pm.003}$	$2.922^{\pm.006}$	$\mathbf{.680}^{\pm.001}$
Attention/Token Refiner	263D	MM	8/512	✓	$.571^{\pm.005}$	$.764^{\pm.003}$	$.853^{\pm.003}$	$.046^{\pm.004}$	$2.890^{\pm.010}$	$\mathbf{.686}^{\pm.000}$
		MM	8/512	✗	$.568^{\pm.003}$	$.761^{\pm.003}$	$.848^{\pm.002}$	$\mathbf{.043}^{\pm.003}$	$2.922^{\pm.007}$	$.679^{\pm.001}$
		Cross	8/512	✓	$.572^{\pm.005}$	$.762^{\pm.004}$	$.854^{\pm.003}$	$.044^{\pm.003}$	$2.880^{\pm.005}$	$\mathbf{.686}^{\pm.001}$
		Cross	8/512	✗	$\mathbf{.581}^{\pm.005}$	$\mathbf{.776}^{\pm.004}$	$\mathbf{.865}^{\pm.001}$	$.099^{\pm.006}$	$\mathbf{2.822}^{\pm.010}$	$.684^{\pm.001}$
	XYZ	MM	8/512	✓	$.566^{\pm.004}$	$.759^{\pm.003}$	$.849^{\pm.003}$	$\mathbf{.038}^{\pm.004}$	$2.916^{\pm.008}$	$.676^{\pm.001}$
		MM	8/512	✗	$.560^{\pm.004}$	$.754^{\pm.004}$	$.844^{\pm.002}$	$.045^{\pm.005}$	$2.942^{\pm.010}$	$.675^{\pm.001}$
		Cross	8/512	✓	$.556^{\pm.006}$	$.756^{\pm.004}$	$.852^{\pm.002}$	$.079^{\pm.003}$	$2.954^{\pm.008}$	$\mathbf{.677}^{\pm.001}$
		Cross	8/512	✗	$\mathbf{.568}^{\pm.003}$	$\mathbf{.766}^{\pm.003}$	$\mathbf{.857}^{\pm.002}$	$.104^{\pm.004}$	$\mathbf{2.909}^{\pm.011}$	$.676^{\pm.001}$

out the Token Refiner, the cross-attention models obtain the strongest R-Precision and MM-Dist point estimates, yet their FID degrades to .099 for 263D and .104 for XYZ. Thus, retrieval alone favors the simpler cross-attention variant, whereas multimodal-attention with flow-time-aware text refinement gives the most robust distributional fidelity across both motion representations and is the configuration used by our primary models.

Table 6: **Complete analysis of representation-aware design choices.** All configurations use $s = 5$ and clean-motion prediction unless stated otherwise. Each interval summarizes 10 stochastic evaluations. Bold denotes the best result for each metric within each group, and gray cells mark the proposed settings.

Factor	Representation	Configuration	R-Precision↑ Top 1	R-Precision↑ Top 2	R-Precision↑ Top 3	FID↓	MM-Dist↓
Attention	263D	Causal	$\mathbf{.571}^{\pm.005}$	$\mathbf{.764}^{\pm.003}$	$\mathbf{.853}^{\pm.003}$	$\mathbf{.046}^{\pm.004}$	$\mathbf{2.890}^{\pm.010}$
		Bidirectional	$.558^{\pm.004}$	$.756^{\pm.003}$	$.846^{\pm.002}$	$.067^{\pm.002}$	$2.973^{\pm.005}$
	XYZ	Causal	$.506^{\pm.004}$	$.701^{\pm.002}$	$.800^{\pm.002}$	$1.563^{\pm.025}$	$3.243^{\pm.008}$
		Bidirectional	$\mathbf{.566}^{\pm.004}$	$\mathbf{.759}^{\pm.003}$	$\mathbf{.849}^{\pm.003}$	$\mathbf{.038}^{\pm.004}$	$\mathbf{2.916}^{\pm.008}$
Source scale	263D	$s = 1$, causal	$\mathbf{.580}^{\pm.004}$	$\mathbf{.775}^{\pm.003}$	$\mathbf{.861}^{\pm.002}$	$.111^{\pm.005}$	$\mathbf{2.851}^{\pm.007}$
		$s = 5$, causal	$.571^{\pm.005}$	$.764^{\pm.003}$	$.853^{\pm.003}$	$\mathbf{.046}^{\pm.004}$	$2.890^{\pm.010}$
	XYZ	$s = 1$, bidirectional	$.548^{\pm.002}$	$.748^{\pm.003}$	$.841^{\pm.003}$	$.144^{\pm.009}$	$2.999^{\pm.008}$
		$s = 5$, bidirectional	$\mathbf{.566}^{\pm.004}$	$\mathbf{.759}^{\pm.003}$	$\mathbf{.849}^{\pm.003}$	$\mathbf{.038}^{\pm.004}$	$\mathbf{2.916}^{\pm.008}$
Prediction target	263D	x-prediction, causal	$\mathbf{.571}^{\pm.005}$	$\mathbf{.764}^{\pm.003}$	$\mathbf{.853}^{\pm.003}$	$\mathbf{.046}^{\pm.004}$	$\mathbf{2.890}^{\pm.010}$
		v-prediction, causal	$.567^{\pm.003}$	$.763^{\pm.003}$	$.851^{\pm.003}$	$.061^{\pm.004}$	$2.900^{\pm.011}$
		x-prediction, bidirectional	$.558^{\pm.004}$	$.756^{\pm.003}$	$.846^{\pm.002}$	$.067^{\pm.002}$	$2.973^{\pm.005}$
		v-prediction, bidirectional	$.551^{\pm.004}$	$.752^{\pm.003}$	$.842^{\pm.002}$	$.101^{\pm.004}$	$3.010^{\pm.009}$
	XYZ	x-prediction, causal	$.506^{\pm.004}$	$.701^{\pm.002}$	$.800^{\pm.002}$	$1.563^{\pm.025}$	$3.243^{\pm.008}$
		v-prediction, causal	$.450^{\pm.002}$	$.637^{\pm.004}$	$.744^{\pm.004}$	$4.482^{\pm.030}$	$3.672^{\pm.012}$
		x-prediction, bidirectional	$\mathbf{.566}^{\pm.004}$	$\mathbf{.759}^{\pm.003}$	$\mathbf{.849}^{\pm.003}$	$\mathbf{.038}^{\pm.004}$	$\mathbf{2.916}^{\pm.008}$
		v-prediction, bidirectional	$.542^{\pm.004}$	$.737^{\pm.002}$	$.831^{\pm.003}$	$.249^{\pm.014}$	$3.008^{\pm.010}$

A.3 Complete Representation-Aware Design Analysis

Table 6 reports all attention, source-scale, and prediction-target configurations, including Top-1/2/3 R-Precision, FID, MM-Dist, and 95% confidence intervals. The main paper retains the comparisons needed for its discussion in compact form.

A.4 Evaluation with the Standard 263D Protocol

We complement the 67D evaluation in the main paper with the standard 263D HumanML3D evaluator used by MDM (Tevet et al., 2023). Table 7 shows that the incremental variant achieves the best Top-1 and Top-3 R-Precision scores of 0.591 and 0.863, respectively, and the best MM-Dist of 2.565, while tying CMDM for the best Top-2 R-Precision of 0.778. The global variant obtains the second-best FID of 0.050 and the second-best MM-Dist of 2.607. Together, these results demonstrate strong retrieval and distributional performance under both evaluator protocols.

A.5 Training and Evaluation with the 272D MotionStreamer Protocol

To compare under the protocol introduced by MotionStreamer (Xiao et al., 2025), we train a new causal MSFlow model from scratch on the 272D HumanML3D representation used by MotionStreamer, rather than converting or fine-tuning our 263D model. We use the proposed architecture with source scale $s = 5$ and causal attention because the 272D representation, like the 263D representation, encodes incremental motion. We evaluate the model using the same official 272D evaluator as MotionStreamer and UMO (Cong et al., 2026).

As shown in Table 8, the new 272D MSFlow model improves FID from 11.790 for MotionStreamer and 9.460 for UMO-Unified to 6.540. It also increases Top-1/2/3 R-Precision to $0.792/0.919/0.955$ and reduces MM-Dist to 13.864, outperforming both UMO variants on every reported metric. These comparisons use the same evaluator, although their generation spaces differ: MotionStreamer and our model are trained on 272D motion, whereas UMO generates a 201D HY-Motion representation and converts its output to 272D for evaluation.

Difference from the standard 263D representation. For $K = 22$ SMPL joints, the standard HumanML3D vector contains planar root velocity (2D), scalar yaw velocity (1D), root height (1D), root-relative positions for the $K - 1$ non-root joints ($3(K - 1)$D), local velocities for all joints ($3K$D), IK-derived 6D rotations for the non-root joints ($6(K-1)$D), and four foot-contact channels, totaling 263 dimensions. The 272D representation instead contains planar root velocity (2D), a 6D

Table 7: **Text-to-motion results on HumanML3D using the standard 263D evaluator.** MSFLOW (263D) denotes the causal incremental-263D variant, while MSFLOW (XYZ) denotes the bidirectional global-XYZ variant; both use clean-motion prediction and $s = 5$. All intervals are 95% confidence intervals. **Bold** indicates the best result, and underline denotes the second-best result.

Method	Representation	R-Precision↑			FID↓	MM-Dist↓	MModality↑
		Top 1	Top 2	Top 3			
GT	–	$.511^{\pm.003}$	$.703^{\pm.003}$	$.797^{\pm.002}$	$.002^{\pm.000}$	$2.974^{\pm.008}$	–
T2M-GPT (Zhang et al., 2023a)	Discrete latent	$.492^{\pm.003}$	$.679^{\pm.002}$	$.775^{\pm.002}$	$.141^{\pm.005}$	$3.121^{\pm.009}$	$1.831^{\pm.048}$
MMM (Pinyoanuntapong et al., 2024)		$.515^{\pm.002}$	$.708^{\pm.002}$	$.804^{\pm.002}$	$.089^{\pm.005}$	$2.926^{\pm.007}$	$1.226^{\pm.035}$
MoMask (Guo et al., 2024)		$.521^{\pm.002}$	$.713^{\pm.002}$	$.807^{\pm.002}$	$\mathbf{.045^{\pm.002}}$	$2.958^{\pm.008}$	$1.241^{\pm.040}$
IRG-MotionLLM (Li et al., 2026)		$.564^{\pm.002}$	$.754^{\pm.002}$	$.841^{\pm.002}$	$.208^{\pm.003}$	$2.628^{\pm.005}$	–
MLD V2 (Chen et al., 2023)	Continuous latent	$.542^{\pm.002}$	$.735^{\pm.002}$	$.827^{\pm.002}$	$.078^{\pm.004}$	$2.808^{\pm.006}$	$1.676^{\pm.060}$
MotionLCM V2 (Dai et al., 2024)		$.548^{\pm.002}$	$.743^{\pm.002}$	$.835^{\pm.002}$	$.092^{\pm.003}$	$2.760^{\pm.008}$	$1.800^{\pm.047}$
StableMoFusion (Huang et al., 2024a)		$.553^{\pm.003}$	$.748^{\pm.002}$	$.841^{\pm.002}$	$.098^{\pm.003}$	$2.715^{\pm.006}$	$1.774^{\pm.051}$
EnergyMoGen (Zhang et al., 2025)		$.523^{\pm.003}$	$.715^{\pm.002}$	$.815^{\pm.002}$	$.188^{\pm.006}$	$2.915^{\pm.007}$	$\underline{2.205}^{\pm.041}$
SALAD (Hong et al., 2025)		$.581^{\pm.003}$	$.769^{\pm.003}$	$.857^{\pm.002}$	$.076^{\pm.002}$	$2.649^{\pm.009}$	$1.751^{\pm.062}$
MARDM (Meng et al., 2025b)		$.517^{\pm.003}$	$.708^{\pm.003}$	$.805^{\pm.003}$	$.116^{\pm.006}$	$2.968^{\pm.010}$	$1.923^{\pm.105}$
MotionStreamer (Xiao et al., 2025)		$.496^{\pm.003}$	$.695^{\pm.002}$	$.793^{\pm.002}$	$.201^{\pm.005}$	$3.041^{\pm.009}$	$1.463^{\pm.075}$
FloodDiffusion (Cai et al., 2026)		$.523^{\pm.002}$	$.717^{\pm.002}$	$.810^{\pm.003}$	$.057^{\pm.002}$	$2.887^{\pm.007}$	–
CMDM (Yu et al., 2026)		$\underline{.588}^{\pm.004}$	$\mathbf{.778^{\pm.002}}$	$\underline{.860}^{\pm.003}$	$.068^{\pm.003}$	$2.620^{\pm.010}$	$1.785^{\pm.074}$
MDM (Tevet et al., 2023)	Raw motion	$.455^{\pm.006}$	$.645^{\pm.007}$	$.749^{\pm.002}$	$.489^{\pm.025}$	$3.330^{\pm.025}$	$\mathbf{2.290^{\pm.070}}$
MotionDiffuse (Zhang et al., 2024)		$.491^{\pm.001}$	$.681^{\pm.001}$	$.782^{\pm.001}$	$.630^{\pm.001}$	$3.113^{\pm.001}$	$1.553^{\pm.042}$
ReMoDiffuse (Zhang et al., 2023b)		$.510^{\pm.005}$	$.698^{\pm.006}$	$.795^{\pm.004}$	$.103^{\pm.004}$	$2.974^{\pm.016}$	$1.795^{\pm.043}$
MSFLOW (263D)		$\mathbf{.591^{\pm.004}}$	$\mathbf{.778^{\pm.004}}$	$\mathbf{.863^{\pm.003}}$	$.057^{\pm.003}$	$\mathbf{2.565^{\pm.007}}$	$.890^{\pm.048}$
MSFLOW (XYZ)		$.581^{\pm.003}$	$\underline{.773}^{\pm.003}$	$.858^{\pm.002}$	$\underline{.050}^{\pm.004}$	$\underline{2.607}^{\pm.009}$	$1.287^{\pm.099}$

Table 8: **Text-to-motion results on HumanML3D using the MotionStreamer 272D evaluator.** We include all baselines reported by MotionStreamer and UMO. UMO converts the 201D outputs of HY-Motion and its own models to 272D for evaluation. Our causal 272D model is trained from scratch with clean-motion prediction and $s = 5$. Its intervals are 95% confidence intervals. **Bold** indicates the best generated result.

Method	Representation	FID↓	R-Precision↑			MM-Dist↓
			Top 1	Top 2	Top 3	
Real motion	–	.002	.702	.864	.914	15.151
T2M-GPT (Zhang et al., 2023a)	Discrete latent	12.475	.606	.774	.838	16.812
MotionGPT (Jiang et al., 2023)		14.375	.456	.598	.628	17.892
MoMask (Guo et al., 2024)		12.232	.621	.784	.846	16.138
AttT2M (Zhong et al., 2023)		15.428	.592	.765	.834	15.726
MLD (Chen et al., 2023)	Continuous latent	18.236	.546	.730	.792	16.638
MotionStreamer (Xiao et al., 2025)		11.790	.631	.802	.859	16.081
MDM (Tevet et al., 2023)	Raw motion	23.454	.523	.692	.764	17.423
HY-Motion (Wen et al., 2025)		61.035	.667	.818	.876	17.530
UMO-Expert (Cong et al., 2026)		17.040	.763	.889	.931	15.490
UMO-Unified (Cong et al., 2026)		9.460	.774	.892	.933	15.220
MSFLOW		$\mathbf{6.540^{\pm.120}}$	$\mathbf{.792^{\pm.003}}$	$\mathbf{.919^{\pm.002}}$	$\mathbf{.955^{\pm.002}}$	$\mathbf{13.864^{\pm.014}}$

root angular increment, and positions, velocities, and 6D rotations for all K joints ($3K + 3K + 6K$D), totaling 272 dimensions. Thus, it removes explicit root height and foot-contact channels, includes the root in the position and rotation blocks, and replaces scalar yaw velocity with a 6D rotation increment. More importantly, its joint rotations come directly from the source SMPL motion rather than being recovered from positions by inverse kinematics. This retains twist information and permits direct SMPL animation without the slow, error-prone SMPLify post-processing required when only positions from the standard representation are used.

A.6 TRAINING AND EVALUATION ON SNAPMOGEN

We additionally train an MSFLOW model from scratch on SnapMoGen (Guo et al., 2025) using its native 296D motion representation with source scale $s = 10$. We evaluate the results using the same protocol used by previous works (Guo et al., 2025).

Table 9: **Text-to-motion results on SnapMoGen.** Our result uses a model trained from scratch on the native 296D SnapMoGen representation with clean-motion prediction and $s = 10$. All intervals are 95% confidence intervals over 10 runs. **Bold** and underline indicate the best and second-best generated results.

Method	Representation	R-Precision↑			FID↓	MModality↑
		Top 1	Top 2	Top 3		
GT	–	$.940^{\pm.001}$	$.976^{\pm.001}$	$.985^{\pm.001}$	$.001^{\pm.000}$	–
T2M-GPT (Zhang et al., 2023a)	Discrete latent	$.618^{\pm.002}$	$.773^{\pm.002}$	$.812^{\pm.002}$	$32.629^{\pm.087}$	$9.172^{\pm.181}$
MoMask (Guo et al., 2024)		$.777^{\pm.002}$	$.888^{\pm.002}$	$.927^{\pm.002}$	$17.404^{\pm.051}$	$8.183^{\pm.184}$
MoMask^{++} (Guo et al., 2025)		$.802^{\pm.001}$	$.905^{\pm.002}$	$.938^{\pm.001}$	$\underline{15.061}^{\pm.065}$	$7.259^{\pm.180}$
ScaleMoGen (Hwang et al., 2026)		$.807^{\pm.004}$	$.908^{\pm.003}$	$.941^{\pm.003}$	$16.350^{\pm.086}$	$9.399^{\pm.669}$
StableMoFusion (Huang et al., 2024a)	Continuous latent	$.679^{\pm.002}$	$.823^{\pm.002}$	$.888^{\pm.002}$	$27.801^{\pm.063}$	$9.064^{\pm.138}$
MARDM (Meng et al., 2025b)		$.648^{\pm.002}$	$.801^{\pm.002}$	$.856^{\pm.002}$	$26.348^{\pm.208}$	$9.883^{\pm.147}$
MotionStreamer (Xiao et al., 2025)		$.631^{\pm.002}$	$.791^{\pm.002}$	$.836^{\pm.002}$	$30.023^{\pm.131}$	$7.543^{\pm.195}$
CMDM (Yu et al., 2026)		$\underline{.831}^{\pm.004}$	$\underline{.926}^{\pm.003}$	$\underline{.958}^{\pm.002}$	$\mathbf{14.451}^{\pm.089}$	$9.521^{\pm.196}$
MDM (Tevet et al., 2023)	Raw motion	$.503^{\pm.002}$	$.653^{\pm.002}$	$.727^{\pm.002}$	$57.783^{\pm.092}$	$\mathbf{13.412}^{\pm.231}$
MSFLOW		$\mathbf{.910}^{\pm.002}$	$\mathbf{.969}^{\pm.002}$	$\mathbf{.984}^{\pm.001}$	$16.342^{\pm.118}$	$\underline{12.538}^{\pm.591}$

Native motion representation and BVH recovery. Each SnapMoGen frame contains root-yaw velocity (1D), planar root velocity (2D), root height (1D), heading-normalized global 6D rotations (24×6D), joint positions (24×3D), joint velocities (24×3D), and four foot-contact indicators, totaling 296 dimensions. Unlike HumanML3D's parent-relative rotations, SnapMoGen rotations are global after removing root heading. BVH recovery requires only the first 148 channels: integrating root motion recovers the trajectory, while converting 6D rotations to quaternions, restoring heading, and applying the template hierarchy yields local BVH rotations. This also differs from evaluation on HumanML3D: its 67D protocol embeds four root channels and root-relative XYZ positions for 21 non-root joints ($4 + 21 \times 3 = 67$), whereas the official SnapMoGen evaluator embeds the first 148 root-and-rotation channels. Our model nevertheless trains on and generates all 296 channels, with the remaining channels providing redundant learning signals.

Attention graph. HumanML3D's incremental features naturally support forward accumulation and therefore favor causal attention. In contrast, SnapMoGen uses a hybrid representation that combines root motion and global joint rotations with redundant position, velocity, and contact features. Bidirectional attention uses context from both directions to keep these features consistent across the full motion, and it outperforms causal attention on SnapMoGen. SnapMoGen also uses longer sequences (312 versus 192 frames), amplifying the context asymmetry of a causal mask. We therefore use bidirectional attention, while viewing this design as an interaction among representation, sequence length, and source scale rather than a universal advantage.

Table 9 shows that MSFLOW obtains the strongest text–motion alignment, improving Top-1/2/3 R-Precision from $0.831/0.926/0.958$ for CMDM to $0.910/0.969/0.984$ and exceeding ScaleMoGen's $0.807/0.908/0.941$. Its FID of 16.342 is higher than CMDM (14.451) and MoMask^{++} (15.061), but slightly lower than ScaleMoGen (16.350) and all other remaining baselines, including MoMask (17.404). MSFLOW also obtains the second-best multimodality score of 12.538, behind MDM (13.412). The result therefore improves semantic retrieval and distributional fidelity over the previous SnapMoGen checkpoint while retaining strong conditional diversity.

A.7 Compute Efficiency

We profile MSFLOW and the selected prior methods under a common 196-frame generation protocol. For each method, we use its official classifier-free guidance and sampling schedule and measure the complete sampling trajectory, including decoding into motion space. We retain each release's native motion representation: MARDM generates 67D features, MotionStreamer generates 272D features, and the remaining methods generate 263D features. All runs use batch size one, FP32 with TF32 enabled, and the prompt "a person walks forward." on a single NVIDIA A100-SXM4-80GB GPU. We report the median of five synchronized runs after two warm-up runs. FLOPs are measured with the PyTorch flop counter, with a multiply–add counted as two operations.

Table 10: **Quality and compute efficiency for generating a 196-frame motion.** Parameters include the complete learned generation stack and motion encoder, where applicable, but exclude frozen text encoders. FLOPs count a multiply–add as two operations. Time is the median of five runs on one NVIDIA A100-SXM4-80GB GPU. FID and R-Top3 are point estimates from the standard 263D evaluation in Table 7. **Bold** and underline denote the best and second-best available results among the methods shown.

Method	Params (M) ↓	FLOPs (TF) ↓	Time (s) ↓	FID ↓	R-Top3 ↑
MARDM (Meng et al., 2025b)	309.65	75.477	9.579	0.116	0.805
SALAD (Hong et al., 2025)	**10.06**	**0.467**	**0.477**	0.076	0.857
MotionStreamer (Xiao et al., 2025)	318.51	1.959	9.023	0.201	0.793
CMDM (Yu et al., 2026)	115.01	<u>0.572</u>	1.809	<u>0.068</u>	<u>0.860</u>
FloodDiffusion (Cai et al., 2026)	131.22	7.060	3.537	**0.057**	0.810
MSFLOW	<u>68.77</u>	1.502	<u>1.742</u>	**0.057**	**0.863**

For model size, we count all learned modules required for motion generation, including the full motion encoder when a method uses one, but omit frozen text encoders because they perform cacheable, one-time prompt preprocessing outside the iterative generator. Their execution is likewise excluded from FLOPs and timing. MARDM uses its official trained checkpoint and the actual cached CLIP embedding because its adaptive Dopri5 solver has conditioning-dependent function evaluations. FloodDiffusion internally requires 50 latent tokens, which decode to 197 frames; we follow its length construction and crop the final decoded frame so that every reported output contains exactly 196 frames.

As shown in Table 10, SALAD is the most efficient method on all three compute measures. MSFLOW is second-smallest at 68.77M parameters and second-fastest at 1.742 seconds, while its 1.502 TFLOPs is the third-lowest after SALAD and CMDM. Using the standard 263D evaluation from Table 7, it obtains the highest R-Top3 of 0.863 and ties FloodDiffusion for the best FID of 0.057. Relative to CMDM, it uses 40.2% fewer parameters and is 3.7% faster, while improving R-Top3 by 0.003 and FID by 0.011, although it requires more FLOPs. It is also faster and uses fewer FLOPs than MotionStreamer, FloodDiffusion, and MARDM. Thus, direct full-sequence motion generation remains competitive in model size and latency while attaining the strongest reported quality.

B Detailed Spatial-Control Results

The main paper reports the pelvis-only and all-joints-average results in Table 4. Table 11 gives the complete per-joint evaluation under the same OmniControl (Xie et al., 2024) protocol. Across the pelvis, feet, head, and wrists, MSFLOW has zero trajectory, location, and average errors. Compared with ProjFlow (Watanabe et al., 2026), it lowers FID and raises R-Precision@3 for every reported joint. Averaged over the all-joints setting, FID improves from 0.097 to 0.061 and R-Precision@3 from 0.779 to 0.818. The average foot-skating ratio also decreases from 0.0603 to 0.0578, although exact satisfaction of projected coordinates and a single physical metric do not establish the quality of the unconstrained motion.

Following MaskControl (Pinyoanuntapong et al., 2025) and ProjFlow (Watanabe et al., 2026), upper-body editing conditions on the ground-truth pelvis, left-foot, and right-foot signals at every frame and generates the remaining motion from the text description. The final block reports this task using the comparison and metric format of ProjFlow. MSFLOW achieves the best FID, reducing it from 0.066 to 0.034 relative to the strongest baseline, and obtains R-Precision of 0.511, 0.701, and 0.794 at ranks 1–3, a matching distance of 3.214, and diversity of 10.714.

C Implementation and Reproducibility Details

We next specify the two principal MSFLOW variants and give the training, sampling, and endpoint-projection algorithms corresponding to the framework in Fig. 2. Unless otherwise stated, models are trained for 500 epochs with batch size 64 using AdamW (Loshchilov & Hutter, 2019) and learning rate 2×10^{-4}. The 68M-parameter transformer has eight RA-MMDiT blocks of width 512, four attention heads, and feed-forward width 1024. We use text dropout 0.1, classifier-free guidance

Table 11: **Complete results for text-conditioned spatial control and upper-body editing on HumanML3D.** The first block trains and evaluates on pelvis controls. The per-joint blocks use methods trained on all joints. The final block follows the upper-body editing format of ProjFlow (Watanabe et al., 2026). Bold and underline denote the best and second-best values, respectively; diversity is ranked by absolute deviation from GT, ties share the same formatting, and gray cells mark MSFLOW.

Controlling Joint	Methods	Zero-shot?	FID↓	R-Precision Top 3	Diversity→	Foot Skating Ratio.↓	Traj. err.↓	Loc. err.↓	Avg. err.↓
	GT	–	0.000	0.795	10.455	-	0.000	0.000	0.000
Train On Pelvis	MDM (Tevet et al., 2023)	✓	1.792	0.673	9.131	0.1019	0.4022	0.3076	0.5959
	PriorMDM (Shafir et al., 2024)	✗	0.393	0.707	9.847	0.0897	0.3457	0.2132	0.4417
	GMD (Karunratanakul et al., 2023)	✓	0.238	0.763	10.011	0.1009	0.0931	0.0321	0.1439
	OmniControl (Xie et al., 2024)	✗	0.081	0.789	10.323	**0.0547**	0.0387	0.0096	0.0338
	ProjFlow (Watanabe et al., 2026)	✓	0.107	0.784	10.645	0.0630	**0.0000**	**0.0000**	**0.0000**
	MSFLOW	✓	**0.068**	**0.821**	**10.447**	0.0591	**0.0000**	**0.0000**	**0.0000**
Pelvis	OmniControl (Xie et al., 2024)	✗	0.135	0.790	10.314	0.0571	0.0404	0.0085	0.0367
	MotionLCM V2+CtrlNet (Dai et al., 2024)	✗	4.726	0.713	9.209	0.1162	0.1617	0.0841	0.1838
	MaskControl (Pinyoanuntapong et al., 2025)	✗	0.087	0.795	10.168	**0.0544**	0.0003	**0.0000**	0.0114
	ProjFlow (Watanabe et al., 2026)	✓	0.107	0.784	10.645	0.0630	**0.0000**	**0.0000**	**0.0000**
	MSFLOW	✓	**0.068**	**0.821**	**10.447**	0.0591	**0.0000**	**0.0000**	**0.0000**
Left foot	OmniControl (Xie et al., 2024)	✗	0.093	0.794	**10.338**	0.0692	0.0594	0.0094	0.0314
	MotionLCM V2+CtrlNet (Dai et al., 2024)	✗	4.810	0.706	9.158	0.1047	0.2607	0.1229	0.2304
	MaskControl (Pinyoanuntapong et al., 2025)	✗	0.074	0.793	10.241	**0.0561**	**0.0000**	**0.0000**	0.0066
	ProjFlow (Watanabe et al., 2026)	✓	0.095	0.771	10.644	0.0609	**0.0000**	**0.0000**	**0.0000**
	MSFLOW	✓	**0.041**	**0.817**	10.630	0.0639	**0.0000**	**0.0000**	**0.0000**
Right foot	OmniControl (Xie et al., 2024)	✗	0.137	0.798	10.241	0.0668	0.0666	0.0120	0.0334
	MotionLCM V2+CtrlNet (Dai et al., 2024)	✗	4.756	0.705	9.303	0.1026	0.2459	0.1127	0.2278
	MaskControl (Pinyoanuntapong et al., 2025)	✗	0.080	0.793	10.159	**0.0552**	**0.0000**	**0.0000**	0.0062
	ProjFlow (Watanabe et al., 2026)	✓	0.096	0.770	**10.651**	0.0613	**0.0000**	**0.0000**	**0.0000**
	MSFLOW	✓	**0.043**	**0.818**	10.729	0.0642	**0.0000**	**0.0000**	**0.0000**
Head	OmniControl (Xie et al., 2024)	✗	0.146	0.796	10.239	0.0556	0.0422	0.0079	0.0349
	MotionLCM V2+CtrlNet (Dai et al., 2024)	✗	4.580	0.715	9.278	0.1138	0.1971	0.0977	0.2136
	MaskControl (Pinyoanuntapong et al., 2025)	✗	0.090	0.797	10.131	0.0531	**0.0000**	**0.0000**	0.0064
	ProjFlow (Watanabe et al., 2026)	✓	0.099	0.788	10.754	0.0595	**0.0000**	**0.0000**	**0.0000**
	MSFLOW	✓	**0.066**	**0.821**	**10.642**	**0.0528**	**0.0000**	**0.0000**	**0.0000**
Left wrist	OmniControl (Xie et al., 2024)	✗	0.119	0.783	10.217	0.0562	0.0801	0.0134	0.0529
	MotionLCM V2+CtrlNet (Dai et al., 2024)	✗	4.103	0.726	9.188	0.1167	0.3965	0.1912	0.3150
	MaskControl (Pinyoanuntapong et al., 2025)	✗	0.118	0.797	10.153	0.0546	**0.0000**	**0.0000**	0.0044
	ProjFlow (Watanabe et al., 2026)	✓	0.089	0.783	10.601	0.0586	**0.0000**	**0.0000**	**0.0000**
	MSFLOW	✓	**0.075**	**0.818**	**10.576**	**0.0535**	**0.0000**	**0.0000**	**0.0000**
Right wrist	OmniControl (Xie et al., 2024)	✗	0.128	0.792	10.309	0.0601	0.0813	0.0127	0.0519
	MotionLCM V2+CtrlNet (Dai et al., 2024)	✗	4.051	0.725	9.242	0.1176	0.3822	0.1806	0.3079
	MaskControl (Pinyoanuntapong et al., 2025)	✗	0.121	0.797	10.105	0.0537	**0.0000**	**0.0000**	0.0044
	ProjFlow (Watanabe et al., 2026)	✓	0.096	0.780	10.610	0.0584	**0.0000**	**0.0000**	**0.0000**
	MSFLOW	✓	**0.075**	**0.815**	**10.533**	**0.0534**	**0.0000**	**0.0000**	**0.0000**
Average	OmniControl (Xie et al., 2024)	✗	0.126	0.792	10.276	0.0608	0.0617	0.0107	0.0404
	MotionLCM V2+CtrlNet (Dai et al., 2024)	✗	4.504	0.715	9.230	0.1119	0.2740	0.1315	0.2464
	MaskControl (Pinyoanuntapong et al., 2025)	✗	0.095	0.795	10.159	**0.0545**	0.0001	**0.0000**	0.0065
	ProjFlow (Watanabe et al., 2026)	✓	0.097	0.779	10.651	0.0603	**0.0000**	**0.0000**	**0.0000**
	MSFLOW	✓	**0.061**	**0.818**	**10.593**	0.0578	**0.0000**	**0.0000**	**0.0000**

	Methods	Zero-shot?	FID↓	R-Precision Top 1	R-Precision Top 2	R-Precision Top 3	Matching↓	Diversity→	–
Upper-Body Edit	MDM (Tevet et al., 2023)	✓	1.918	0.359	0.556	0.654	4.793	9.210	–
	OmniControl (Xie et al., 2024)	✗	0.909	0.428	0.614	0.722	3.694	10.207	–
	MotionLCM V2+CtrlNet (Dai et al., 2024)	✗	3.922	0.404	0.592	0.692	5.610	9.309	–
	MaskControl (Pinyoanuntapong et al., 2025)	✗	0.066	0.501	0.695	0.794	3.227	10.159	–
	ACMDM-S-PS22+CtrlNet (Meng et al., 2025a)	✗	0.076	**0.532**	**0.719**	**0.820**	**3.098**	10.586	–
	ProjFlow (Watanabe et al., 2026)	✓	0.087	0.501	0.690	0.787	3.319	**10.571**	–
	MSFLOW	✓	**0.034**	0.511	0.701	0.794	3.214	10.714	–

weight 3, one sequence-level $t \sim \mathcal{U}(0,1)$ per training example, and $\sigma_{\min} = 0.05$ for training and 0.01 for inference. Sampling uses EMA weights, fixed-step Heun updates, and a final Euler step. A complete 500-epoch training run of MSFLOW takes approximately 450 minutes on one NVIDIA A100-SXM4-80GB GPU.

C.1 CAUSAL 263D TEXT-TO-MOTION CONFIGURATION

Table 12 summarizes the configuration of the causal 263D generator.

C.2 BIDIRECTIONAL XYZ TEXT-TO-MOTION CONFIGURATION

Table 13 summarizes the corresponding configuration of the bidirectional XYZ generator.

Table 12: Configuration of the causal 263D model used for the primary HumanML3D result. The evaluated EMA checkpoint was not retained with an immutable run and epoch identifier.

Item	Selected value
Model variant	causal 263D, $s = 5$
Training representation	HumanML3D, 263 dimensions
Evaluation representation	root/RIC subset, 67 dimensions
Backbone	8 joint blocks, width 512, 4 heads
Motion block size	1 frame
Training / inference mask	causal / causal
Flow path / time	linear / uniform sequence-level t
Network output / loss	endpoint $\hat{\mathbf{x}}_1$ / stabilized residual MSE
Source distribution	$\mathcal{N}(0, 25\mathbf{I})$
Text encoder	frozen DistilBERT, token features
Token Refiner	2 layers, timestep/context conditioned
CFG / condition dropout	3.0 / 0.1
Sampler	50 time points, Heun + final Euler
Evaluated weights	latest EMA checkpoint

Table 13: Configuration of the bidirectional XYZ model used for the primary HumanML3D result.

Item	Selected value
Model variant	bidirectional XYZ, $s = 5$
Training representation	shared-axis-normalized XYZ, 66 dimensions
Evaluation representation	root/RIC subset, 67 dimensions
Backbone	8 joint blocks, width 512, 4 heads
Motion block size	1 frame
Training / inference mask	bidirectional / bidirectional
Flow path / time	linear / uniform sequence-level t
Network output / loss	endpoint $\hat{\mathbf{x}}_1$ / stabilized residual MSE
Source distribution	$\mathcal{N}(0, 25\mathbf{I})$
Text encoder	frozen DistilBERT, token features
Token Refiner	2 layers, timestep/context conditioned
CFG / condition dropout	3.0 / 0.1
Sampler	50 time points, Heun + final Euler
Evaluated weights	latest EMA checkpoint labeled

C.3 Training and Sampling Algorithms

Algorithms 1 and 2 detail the shared training and text-to-motion sampling procedures, respectively.

C.4 Bidirectional XYZ Endpoint-Projection Configuration

Spatial control reuses the source scale and attention graph of the trained bidirectional XYZ generator. Arbitrary binary masks select valid XYZ entries, while neither masks nor target coordinates are provided during training. Table 14 summarizes this inference-only configuration, and Algorithm 3 details the endpoint-projection sampler.

D Sample Code

We provide code for training and evaluating the proposed MSFlow on the HumanML3D dataset. ***Please refer to the README file in the sample-code directory of the supplementary materials for details.***

Algorithm 1 Representation-aware direct flow training

Require: Motion–text minibatch $(\mathbf{x}_1, \mathbf{c})$, source scale s
1: Sample $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$ and $t \sim \mathcal{U}(0, 1)$
2: Form $\mathbf{x}_0 \leftarrow s\boldsymbol{\epsilon}$ and $\mathbf{x}_t \leftarrow (1-t)\mathbf{x}_0 + t\mathbf{x}_1$
3: Refine frozen text tokens using t and their masked global mean
4: Choose the fixed mask $\mathbf{M}_{\mathrm{Att}}$ (causal 263D or bidirectional XYZ)
5: Predict $\hat{\mathbf{x}}_1 \leftarrow f_\theta(\mathbf{x}_t, t, \mathbf{c}; \mathbf{M}_{\mathrm{Att}})$
6: Convert $\hat{\mathbf{x}}_1$ and $\mathbf{x}_1$ to stabilized residual fields with Eq. (5)
7: Update θ with valid-frame residual MSE; update EMA weights

Algorithm 2 MSFLOW sampling

Require: Text $\mathbf{c}$, length T, source scale s, guidance w
1: Sample the full sequence $\mathbf{x}_0 \sim \mathcal{N}(0, s^2\mathbf{I})$
2: Encode and refine conditional and null text features
3: **for** successive time points t_k, t_{k+1} **do**
4: Predict the guided endpoint with classifier-free guidance using the checkpoint mask $\mathbf{M}_{\mathrm{Att}}$
5: Convert the endpoint to the residual field with Eq. (5)
6: Update all motion frames with one Heun step
7: **end for**
8: Replace the final corrector with one Euler update
9: **return** generated motion features $\mathbf{x}_1$

E THEORETICAL DETAILS AND PROOFS

This section provides the proofs underlying the three design questions in the main paper: the autoencoder fidelity floor in latent generation, the geometry induced by a scaled source, and the relationship between temporal attention and motion representation. For the last question, we also provide the full statistical formulation and conclude with the dynamical prefix property of the causal vector field.

E.1 PROOFS OF MAIN PROPOSITIONS

Proposition 1 formalizes the irreducible fidelity error introduced when generated motions are restricted to a fixed decoder's range, motivating direct motion-space generation. Proposition 2 explains how the Gaussian source scale controls both when motion signal emerges along the flow path and the conditioning of intermediate distributions.

Proof of Proposition 1. Let π be any coupling of $\mathbf{X} \sim P_1$ and $\mathbf{Y} \sim Q$. Because Q is supported on $\mathcal{R}_g$, $\mathbf{Y} \in \mathcal{R}_g$ almost surely. Pointwise,

$$\|\mathbf{X} - \mathbf{Y}\|_2^2 \geq \inf_{\mathbf{y} \in \mathcal{R}_g} \|\mathbf{X} - \mathbf{y}\|_2^2 = \mathrm{dist}^2(\mathbf{X}, \mathcal{R}_g). \tag{15}$$

Taking the expectation under π and then the infimum over all couplings yields Eq. (1). □

Proof of Proposition 2. Project Eq. (2) onto an eigenvector $\mathbf{u}$: $\mathbf{u}^\top \mathbf{X}_t = t\mathbf{u}^\top \mathbf{X}_1 + (1-t)s\mathbf{u}^\top \boldsymbol{\epsilon}$. Independence gives signal variance $t^2\lambda_{\mathbf{u}}$ and source-noise variance $(1-t)^2 s^2$, proving Eq. (8). The inequality $\mathrm{SNR}_{\mathbf{u}}(t; s) \leq 1$ is equivalent on $[0, 1]$ to $t \leq s/(s + \sqrt{\lambda_{\mathbf{u}}})$. Under uniform time sampling, the probability is the length of this interval.

Independence also gives Eq. (9). Its extreme eigenvalues are $t^2\lambda_{\min} + (1-t)^2 s^2$ and $t^2\lambda_{\max} + (1-t)^2 s^2$, yielding Eq. (10). Let $a = t^2$ and $b = (1-t)^2 s^2$. For $t < 1$,

$$\frac{\partial}{\partial b} \frac{a\lambda_{\max} + b}{a\lambda_{\min} + b} = \frac{a(\lambda_{\min} - \lambda_{\max})}{(a\lambda_{\min} + b)^2} \leq 0. \tag{16}$$

Since b is increasing in $s > 0$, $\kappa_t(s)$ is non-increasing. □

Table 14: Inference configuration for bidirectional XYZ endpoint projection. The model is trained for text-to-motion generation without masks or target coordinates.

Item	Control-evaluation setting
Training attention	bidirectional, fixed for the XYZ checkpoint
Inference attention	bidirectional
Control training	none
Raw controls	any valid frames, selected joint IDs, and XYZ axes
CFG weight / projected steps	3.0 / 100
Source noise	scale s of the training path
Noise refresh	source-scale corrected

Algorithm 3 Bidirectional zero-shot XYZ control

Require: Bidirectional XYZ checkpoint, text $\mathbf{c}$, length T, target $\mathbf{y}$, mask $\mathbf{M}$
1: Resolve source scale s from the checkpoint
2: Sample $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, s^2\mathbf{I})$
3: **for** successive time points t, t' **do**
4: Predict the CFG velocity and recover $(\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_0)$
5: Project $\hat{\mathbf{x}}_1$ onto $(\mathbf{M}, \mathbf{y})$ using Eq. (11)
6: Optionally refresh $\hat{\mathbf{x}}_0$ using Eq. (13)
7: Recompose $\mathbf{x}_{t'}$ using Eq. (14)
8: **end for**
9: Hard-replace controlled coordinates and zero padded frames
10: **return** controlled direct-space motion $\mathbf{x}_1$

E.2 Empirical Spatiotemporal Covariance and Source Scale

We complement Proposition 2 with a full-split covariance analysis of the two training representations. For each window length $T_0 \in \{32, 64, 128\}$, we draw one uniformly positioned window from every eligible HumanML3D training motion and flatten it to a vector $\mathbf{x}_n \in \mathbb{R}^p$, where $p = T_0 D$. Each coordinate of $\mathbf{x}_n$ therefore identifies one feature channel d at one frame τ. A covariance entry indexed by (τ, d) and (τ', d') measures how those two variables vary together across motions. It includes same-frame relationships between channels, temporal relationships between frames, and cross-channel relationships across different frames; we therefore call it *spatiotemporal* covariance. For XYZ, the channels are joint coordinates, while for 263D they are the channels of the HumanML3D representation.

Sampling at most one window per motion avoids overweighting long or highly correlated sequences. The filter $\max(40, T_0) \le T < 200$ yields 22,326, 20,850, and 13,328 motions for $T_0 = 32$, 64, and 128, respectively. The 263D windows use the official feature-wise HumanML3D normalization, while XYZ uses the shared-axis normalization used to train the global generator.

We first form the empirical covariance across the N sampled motion windows. Ledoit–Wolf shrinkage then moves this estimate toward a scaled identity matrix to stabilize its spectrum when the dimension p is large relative to N. We compare every result with an isotropic Gaussian matrix processed by the identical estimator and matched in both N and p. For correlation matrix $\widehat{\mathbf{R}}$ and shrinkage-covariance eigenvalues λ_k, we report

$$C_{\text{off}} = \left(\frac{1}{p(p-1)} \sum_{i \neq j} \widehat{R}_{ij}^2 \right)^{1/2}, \qquad A_{\text{spec}} = \frac{\lambda_{\max}}{p^{-1} \sum_k \lambda_k}, \tag{17}$$

$$\overline{r}_{\text{eff}} = \frac{1}{p} \exp\left(- \sum_k q_k \log q_k \right), \qquad q_k = \frac{\lambda_k}{\sum_j \lambda_j}. \tag{18}$$

Here, C_{off} is the root-mean-square off-diagonal correlation. It summarizes pairwise dependence between distinct frame–channel variables: zero means no linear pairwise correlation, and larger values indicate stronger dependence. A_{spec} divides the largest covariance eigenvalue by the mean

Table 15: **Spatiotemporal covariance diagnostics across window lengths.** Each statistic is reported as motion / a matched isotropic Gaussian with the same sample count N, dimension p, and covariance estimator. Lower C_{off}, lower A_{spec}, and higher $\overline{r}_{\text{eff}}$ indicate greater isotropy.

Representation	T_0	N	C_{off}	A_{spec}	$\overline{r}_{\text{eff}}$
263D	32	22,326	.166/.007	1088.4/1.0	.009/1.000
	64	20,850	.148/.007	2116.0/1.0	.007/1.000
	128	13,328	.137/.009	4336.5/1.0	.004/1.000
XYZ	32	22,326	.485/.007	1027.0/1.0	.002/1.000
	64	20,850	.444/.007	1808.7/1.0	.001/1.000
	128	13,328	.390/.009	3346.3/1.0	.001/1.000

Table 16: **Covariance diagnostics of pretrained latent representations.** All methods use the same 20,850 HumanML3D training motions and one 64-frame window per motion. Latent shape excludes the batch dimension. Each diagnostic is reported as latent / a matched isotropic Gaussian with the same N, p, and covariance estimator.

Method	Latent shape	C_{off}	A_{spec}	$\overline{r}_{\text{eff}}$
CMDM	16×64	.181/.007	118.9/1.0	.077/1.000
SALAD	$16 \times 7 \times 32$	.076/.007	129.3/1.0	.238/1.000

eigenvalue. It equals one when every direction has equal variance and grows when one direction dominates the covariance spectrum.

The normalized weight q_k is the fraction of total covariance variance assigned to eigen-direction k, so $q_k \geq 0$ and $\sum_k q_k = 1$. Exponentiating the entropy of these weights gives the effective number of active covariance directions; division by p produces $\overline{r}_{\text{eff}} \in (0, 1]$. A value near one means variance is spread evenly across the available directions, whereas a value near zero means it is concentrated in relatively few directions. Thus, an isotropic population has $C_{\text{off}} = 0$, $A_{\text{spec}} = 1$, and $\overline{r}_{\text{eff}} = 1$. Effective rank describes the covariance spectrum and is not an estimate of the nonlinear intrinsic dimension of the motion manifold.

Table 15 shows substantial spatiotemporal dependence after normalization. Across all window lengths, motion has much larger residual correlation and spectral anisotropy than its matched Gaussian, together with a far smaller normalized effective rank. For example, at $T_0 = 64$, 263D motion has $C_{\text{off}} = .148$, $A_{\text{spec}} = 2116.0$, and $\overline{r}_{\text{eff}} = .007$, whereas the matched Gaussian gives .007, 1.0, and 1.000. XYZ is even more correlated and spectrally concentrated, with .444, 1808.7, and .001, respectively. The same separation holds at 32 and 128 frames, directly verifying that marginal normalization does not make either motion space isotropic.

For comparison with latent-space generators, we apply the same estimator to the pretrained representations used by CMDM and SALAD. Both results use the common $T_0 = 64$ protocol and the same 20,850 motions, producing 16 latent frames for each method; we preserve each representation's native frame structure when flattening and use one seeded posterior sample per motion, matching the variables seen during generator training.

Table 16 reports only the covariance diagnostics for CMDM and SALAD; we compare them here with the direct-motion results in Table 15. SALAD is the most isotropic representation by both $C_{\text{off}} = .076$ and $\overline{r}_{\text{eff}} = .238$. CMDM has slightly higher pairwise correlation than 263D (.181 versus .148), but its much smaller A_{spec} (118.9 versus 2116.0) and larger $\overline{r}_{\text{eff}}$ (.077 versus .007) show that variance is distributed over substantially more directions. These CMDM and SALAD results are consistent with variational regularization, although both remain far from their matched Gaussian references. Because dimensionality and preprocessing differ across representations, these statistics characterize covariance geometry rather than rank generators by sample quality. Nevertheless, they reveal clear differences between latent-space and motion-space representations.

We next substitute the extreme eigenvalues of each estimated shrinkage covariance into Eq. (10). This quantifies the conditioning of the corresponding spatiotemporal probability path without Monte

Table 17: **Source scale changes path conditioning and generation quality.** The columns headed by t report $\log_{10} \kappa_t(s)$, computed at $T_0 = 64$ from the extreme eigenvalues of the estimated shrinkage covariance; lower is better. FID uses the matched clean-prediction configurations from Table 3: causal attention for 263D and bidirectional attention for XYZ.

Representation	Scale s	$t = 0.1$	$t = 0.5$	$t = 0.9$	FID↓
263D	1	1.516	3.410	5.241	$.111^{\pm.005}$
	5	**.356**	**2.017**	**3.918**	$\mathbf{.046}^{\pm.004}$
XYZ	1	1.253	3.136	5.023	$.144^{\pm.009}$
	5	**.224**	**1.746**	**3.646**	$\mathbf{.038}^{\pm.004}$

Carlo noise. We report $\log_{10} \kappa_t(s)$ at $T_0 = 64$ for an early path time $t = 0.1$, the midpoint $t = 0.5$, and a late path time $t = 0.9$.

Increasing s from 1 to 5 reduces $\log_{10} \kappa_t$ from $1.516/3.410/5.241$ to $.356/2.017/3.918$ at $t = 0.1/0.5/0.9$ for 263D (Table 17). For XYZ, the corresponding values fall from $1.253/3.136/5.023$ to $.224/1.746/3.646$. Thus, s materially changes the conditioning of the correlated motion path at early, middle, and late times; it is not merely an inference-time diversity parameter. The matched clean-prediction ablations exhibit correspondingly large sensitivity: FID falls by 58.6% for causal 263D and by 73.6% for bidirectional XYZ when moving from $s = 1$ to $s = 5$. These controlled results establish the importance of source scale for the selected motion-space generators, while the covariance analysis verifies its path-level conditioning effect. They do not imply that improved conditioning alone causes the FID gains or that $s = 5$ is universally optimal: the best scale can change with the representation, attention graph, and prediction parameterization.

E.3 Why Attention Depends on the Representation

The attention choice reflects what a frame token represents. Incremental features describe local changes whose global effect is obtained by forward accumulation, suggesting a filtering-style causal dependency. Absolute coordinates describe a globally coupled trajectory for which future observations can help smooth an earlier state. We formalize this distinction and its limitations below.

Let $\mathrm{Att}(i)$ be the set of motion-key frames visible to the prediction at frame i:

$$\mathrm{Att}_{\mathrm{C}}(i) = \{1, \dots, i\}, \qquad \mathrm{Att}_{\mathrm{B}}(i) = \{1, \dots, T\}. \tag{19}$$

The corresponding field component has the dependency

$$\hat{\mathbf{v}}_{\theta,i}(\mathbf{x}_t, t, \mathbf{c}) = \hat{\mathbf{v}}_{\theta,i}(\mathbf{x}_{t,\mathrm{Att}(i)}, t, \mathbf{c}). \tag{20}$$

To isolate the statistical cost of the causal restriction, set $Y_i = \mathbf{X}_{1,i}$, $\mathcal{G}_i = \sigma(\mathbf{X}_{t,1:i}, t, \mathbf{c})$, and $\mathcal{H} = \sigma(\mathbf{X}_{t,1:T}, t, \mathbf{c})$. Let $R_{\mathrm{C},i}$ and $R_{\mathrm{B},i}$ be the mean-squared errors of the corresponding Bayes-optimal conditional means.

Proposition 3 (Causal excess prediction risk)**.** *For square-integrable motion,*

$$R_{\mathrm{C},i} - R_{\mathrm{B},i} = \mathbb{E}\left[\|\mathbb{E}[Y_i \mid \mathcal{H}] - \mathbb{E}[Y_i \mid \mathcal{G}_i]\|_2^2\right] \geq 0. \tag{21}$$

Equality holds exactly when the full-sequence conditional mean is already measurable from the noisy prefix.

Proof of Proposition 3. Because $\mathcal{G}_i \subseteq \mathcal{H}$, conditional expectation is an orthogonal projection in L^2. Add and subtract $\mathbb{E}[Y_i \mid \mathcal{H}]$ inside the causal residual:

$$Y_i - \mathbb{E}[Y_i \mid \mathcal{G}_i] = Y_i - \mathbb{E}[Y_i \mid \mathcal{H}] \tag{22}$$

$$+ \mathbb{E}[Y_i \mid \mathcal{H}] - \mathbb{E}[Y_i \mid \mathcal{G}_i]. \tag{23}$$

The first term is orthogonal to every $\mathcal{H}$-measurable random variable, including the second term. Taking squared norms and expectations gives

$$R_{\mathrm{C},i} = R_{\mathrm{B},i} + \mathbb{E}\,\|\mathbb{E}[Y_i \mid \mathcal{H}] - \mathbb{E}[Y_i \mid \mathcal{G}_i]\|_2^2, \tag{24}$$

which proves Eq. (21). Equality holds if and only if the two conditional means agree almost surely. □

Additional context cannot increase Bayes-optimal squared error, but it need not provide useful information. The following idealized model identifies when the inequality is strict.

Proposition 4 (Increment prediction versus position smoothing)**.** *Let $V_k \sim \mathcal{N}(0, q)$ and $E_k \sim \mathcal{N}(0, 1)$ be mutually independent, with $t > 0$ and $\tau > 0$. For an increment representation observed as $Z_k = tV_k + \tau E_k$, suffix observations $Z_{i+1:T}$ do not reduce the Bayes risk for V_i. For an absolute-position representation $P_k = \sum_{r \le k} V_r$ observed as $Z_k = tP_k + \tau E_k$, observing Z_{i+1} strictly reduces the Bayes risk for P_i whenever its prefix posterior variance is nonzero.*

Proof of Proposition 4. For the increment representation, let the noisy observation of frame k be $Z_k = tV_k + \tau E_k$, where $\tau > 0$ and the E_k are independent standard Gaussians. Independence across k gives $V_i \perp Z_{i+1:T} \mid Z_{1:i}$, hence $\mathbb{E}[V_i \mid Z_{1:T}] = \mathbb{E}[V_i \mid Z_{1:i}]$ and Proposition 3 gives equal risks.

For the position representation, let $P_k = \sum_{r \le k} V_r$ and $Z_k = tP_k + \tau E_k$. Conditional on the prefix observations, $Z_{i+1} = tP_i + tV_{i+1} + \tau E_{i+1}$. Since V_{i+1} and E_{i+1} are independent of $(P_i, Z_{1:i})$,

$$\mathrm{Cov}(P_i, Z_{i+1} \mid Z_{1:i}) = t \,\mathrm{Var}(P_i \mid Z_{1:i}). \tag{25}$$

For $t > 0$ and nonzero prefix posterior variance, this conditional covariance is nonzero. Gaussian conditioning therefore strictly reduces $\mathrm{Var}(P_i \mid Z_{1:i})$ when Z_{i+1} is added. The full suffix contains Z_{i+1}, so bidirectional Bayes risk is strictly lower. □

This is the classical filtering–smoothing distinction. Importantly, Propositions 3 and 4 do not claim that causal attention has a lower Bayes-optimal reconstruction error. Bidirectional context contains the causal context, so its optimal squared error can only be smaller; for independent increments the two risks are equal because the suffix is redundant. The empirical advantage of a causal mask must therefore come from its inductive bias in a finite model rather than from greater information. When suffix observations carry little task-relevant information, removing their edges reduces the dependency class, discourages a denoising shortcut based on symmetric temporal smoothing, and directs limited capacity toward prefix-conditioned transitions. Moreover, the flow-matching loss is a local regression objective, whereas FID and retrieval evaluate the distribution produced after repeatedly applying the learned field. A context pattern that eases framewise regression need not yield the best integrated generative dynamics.

HumanML3D is not a sequence of independent increments: it mixes root velocities, root-relative positions, rotations, local velocities, and contacts under a text condition. Its root trajectory nevertheless has a distinguished forward accumulation structure, while an individual frame already specifies a complete root-relative pose. This makes the idealized increment model a plausible explanation for why suffix motion can be redundant for important components, not a literal model of the entire representation. Conversely, absolute XYZ trajectories are temporally correlated observations of global joint positions, making their suffix genuinely informative. The proposition therefore supplies a mechanism, not a universal ranking of masks; the representation-specific preference remains an empirical question.

E.4 Empirical Attention Routing

We inspect eight models with a balanced $2 \times 2 \times 2$ comparison of 263D versus XYZ representations, causal versus bidirectional masks, and source scales $s \in \{1, 5\}$. We use 512 HumanML3D test motions available in both representations, with identical captions and representation-specific Gaussian draws. We evaluate $t \in \{0.2, 0.5, 0.8\}$ and average within each motion before aggregating, so heads and layers are not treated as independent observations. Motion-direction statistics condition on the mass assigned to valid motion keys and therefore do not depend on prompt length or text-attention mass.

Table 18 reports the midpoint statistics for all eight checkpoints, and Fig. 5 visualizes their representation-dependent routing patterns. The $s = 5$ maps expose a learned difference within

Table 18: **Midpoint attention routing for all eight selected checkpoints.** Representation specifies the motion input format, Scale is the Gaussian source scale s, and Mask is the temporal attention graph. The remaining columns report percentages at $t = 0.5$, averaged over 512 paired test motions. Future is the fraction of motion-to-motion attention from query frame i assigned to later key frames $j > i$; Local ± 4 is the fraction assigned within four frames of the query; and Span is the attention-weighted expected absolute offset $|j - i|$, normalized by sequence length. Text is the fraction of a motion query's total attention assigned to text keys, while Text→motion is the fraction of a text query's total attention assigned to motion keys.

Representation	Scale	Mask	Future	Local ± 4	Span	Text	Text→motion
263D	$s = 1$	Causal	0.0	43.6	13.6	48.9	0.0
		Bidirectional	47.4	50.3	9.7	31.6	76.4
	$s = 5$	Causal	0.0	46.6	10.4	27.2	0.0
		Bidirectional	49.1	38.1	12.7	17.5	86.4
XYZ	$s = 1$	Causal	0.0	24.2	20.6	61.6	0.0
		Bidirectional	49.6	25.4	17.5	24.2	86.1
	$s = 5$	Causal	0.0	24.5	18.2	52.0	0.0
		Bidirectional	51.4	15.8	20.9	29.1	76.7

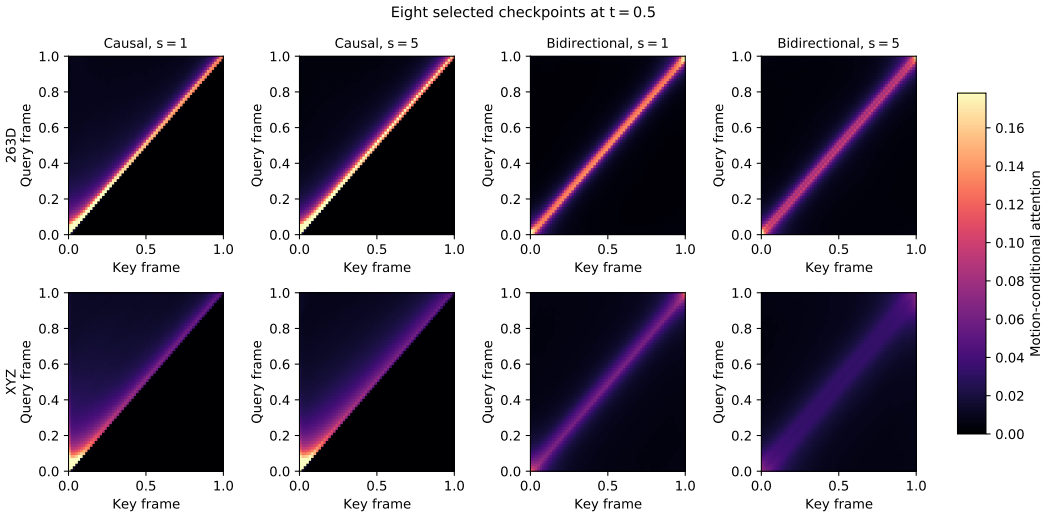

Figure 5: **Representation-dependent attention routing for all eight selected checkpoints.** Rows compare 263D and XYZ; columns compare causal and bidirectional masks at source scales $s = 1$ and $s = 5$. Each map averages 512 paired test motions, four heads, and eight layers at $t = 0.5$ after conditioning each motion-query row on its mass assigned to valid motion keys. The causal mask removes the suffix exactly. Within the bidirectional graph, XYZ distributes attention over a broader temporal range than 263D at both source scales.

the same bidirectional graph: XYZ assigns only 15.8% of motion attention within ± 4 frames, compared with 38.1% for 263D, and its normalized span is 20.9% rather than 12.7%. The independently trained $s = 1$ pair shows the same ordering: XYZ is less local (25.4% versus 50.3%) and broader (17.5% versus 9.7%). Averaged over the two selected scales, bidirectional XYZ remains less local (20.6% versus 44.2%), has a longer span (19.2% versus 11.2%), and assigns a slightly larger fraction to future motion (50.5% versus 48.3%). As shown in Fig. 6, the span ordering holds at every evaluated time: XYZ versus 263D is 22.7% versus 14.4% at $t = 0.2$, 19.2% versus 11.2% at $t = 0.5$, and 16.4% versus 8.0% at $t = 0.8$.

These statistics are consistent with the proposed smoothing role of bidirectional XYZ attention: when suffix edges are available, both selected XYZ models use broader two-sided context rather than merely retaining a nominally dense mask. Conversely, all four selected causal checkpoints assign exactly zero weight to future motion and prevent text queries from reading motion, verifying the intended filtering graph and excluding an indirect suffix path through the updated text stream. At-

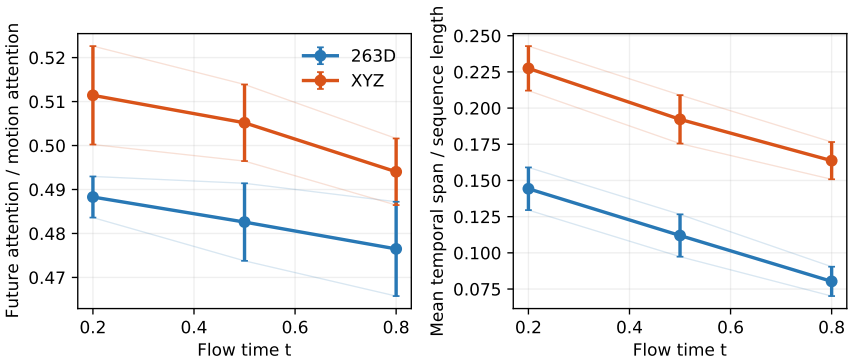

Figure 6: **Bidirectional routing across flow time for the selected checkpoints.** Each checkpoint statistic averages 512 paired test motions. Thin curves show the individual $s = 1$ and $s = 5$ models; markers and error bars show their mean and standard error. Future attention is the fraction of motion-to-motion attention from a query frame i assigned to later key frames $j > i$; span is the attention-weighted expected absolute offset $|j-i|$, normalized by sequence length. XYZ has a longer normalized temporal span at every evaluated time, while both representations allocate approximately half of their motion attention to future frames.

tention weights describe routing rather than causal attribution, however; values, output projections, residual paths, and subsequent layers also determine the influence on the generated motion.

Beyond the Gaussian analysis in Proposition 4, Proposition 5 establishes a distribution-independent dynamical guarantee of the causal mask: the generated prefix is autonomous from perturbations to the source suffix.

Proposition 5 (Prefix autonomy)**.** *Assume the causal field is locally Lipschitz in motion state, so its sampling ODE has a unique solution. For any $k \leq T$, the generated trajectory $\mathbf{x}_{t,1:k}$ depends only on the source prefix $\mathbf{x}_{0,1:k}$ and text $\mathbf{c}$. Changing $\mathbf{x}_{0,k+1:T}$ cannot alter this prefix. Wherever the field is differentiable, its motion Jacobian is block lower triangular.*

Proof of Proposition 5. For any k, causality implies that the first k sampling equations can be written as the closed subsystem

$$\frac{d\mathbf{x}_{t,1:k}}{dt} = \hat{\mathbf{v}}_{\theta,1:k}(\mathbf{x}_{t,1:k}, t, \mathbf{c}). \tag{26}$$

Its initial condition is $\mathbf{x}_{0,1:k}$. Local Lipschitz continuity gives uniqueness, so two full-sequence initial states with the same prefix must induce the same prefix trajectory even if their suffixes differ. If the field is differentiable, Eq. (20) with $\mathrm{Att} = \mathrm{Att}_{\mathrm{C}}$ gives $\partial\hat{\mathbf{v}}_{\theta,i}/\partial\mathbf{x}_{t,j} = \mathbf{0}$ for $j > i$, which is exactly block lower triangularity. □

Causality thus constrains information flow without making generation autoregressive: all T frames are initialized together and updated at every ODE step. The lower-triangular dependency prevents a suffix error or source perturbation from feeding back into an earlier prefix through subsequent solver evaluations. This structural guarantee does not by itself prove smaller numerical error or better samples, but it removes one route for global error propagation and preserves the forward accumulation semantics of incremental root motion. Bidirectional attention gives up prefix autonomy in exchange for full-sequence smoothing and two-sided propagation of spatial constraints, which is beneficial when coordinates are absolute and globally coupled. These complementary properties motivate the attention variants evaluated in Section 4.3.