MotionDuet: Dual-Conditioned 3D Human Motion Generation with Video-Regularized Text Learning

Supplementary Material

A. Evaluation Metrics and Datasets

A.1. Evaluation Metrics

(1) Motion Quality: Fréchet Inception Distance (FID) quantifies the similarity between generated and real motions in feature space; lower scores indicate better quality. **(2) Generation Diversity**: Diversity (DIV) measures variation across generated motions [10], while Multimodality (MM) evaluates diversity for multiple generations from identical inputs. **(3) Conditional Matching**: Motion Retrieval Accuracy (R Accuracy) computes Top 1/2/3 matches between text and motion, and Multimodal Distance (MM Dist) measures text-motion feature similarity [10].

A.2. Datasets

HumanML3D [9], combining HumanAct12 [8] and AMASS [18], features 14,616 motions spanning daily tasks, sports, acrobatics, and artistic performances. Annotated via Amazon MTurk, each clip includes 3-4 sentences, downsampled to 20 fps, lasting 2-10 s (avg. 7.1 s), totaling 28.59 hours. The dataset has $44,970$ descriptions averaging 12 words each from a vocabulary of $5,371$ unique words.

B. Qualitative Experiment

B.1. Qualitative Evaluation on Text to Motion Generation

We present a series of visualized motion results generated by our method to further evaluate its performance in real-world generation scenarios. These examples cover a variety of challenging textual descriptions, involving complex action compositions and directional changes, see Fig. 5. By directly comparing the input text with the corresponding generated motion sequences, we can clearly observe the model's capability to understand semantic intent, capture motion details, and maintain temporal coherence. These visual results not only demonstrate the model's precise response to natural language instructions but also highlight its strength in producing natural, coherent, and semantically consistent human motions.

B.2. Qualitative Ablation on Video-Guided Motion Generation

To deepen this comparison and isolate the contribution of video inputs, we also perform an ablation study in which video inputs are excluded during training. As a result, the DASH Loss is removed due to its reliance on video information, while the remaining components of the DUET module, except for DMM, are preserved to ensure a consistent and fair evaluation. In addition, we conduct qualitative evaluations of the generated motion sequences across a diverse set of textual prompts to further assess the effectiveness of our proposed method. As shown in Fig. 6, our model excels at generating realistic and semantically aligned human motions in response to complex natural language descriptions.

Compared to baseline models, our approach demonstrates superior physical plausibility and motion continuity, particularly in managing transitions between distinct motion primitives (e.g., turning, running, or crouching). These results underscore the model's ability to produce context-aware, text-consistent motions in scenarios demanding precise temporal ordering and stylistic fidelity. Overall, these qualitative examples highlight our method's exceptional ability to capture both high-level semantic intent and fine-grained motion dynamics.

B.3. Qualitative Evaluation of Generalization on Unseen Real-World Videos

To rigorously evaluate the model's real-world applicability and generalization ability, we select real-life videos from the reference [6], none of which appear during training or are included in the dataset. These videos are preprocessed and carefully trimmed into the input format required by our model. The selected samples feature several representative and high-difficulty actions, such as ballet spins, baseball pitching, hitting an incoming baseball with a bat, and golf swings (see Fig. 4 and Fig. 7). This evaluation serves as a strong qualitative test of the model's ability to handle complex real-world motion scenarios.

When simulating the action of hitting a baseball with a bat, the model successfully reproduces the complete process, including lifting the bat overhead, swinging it clockwise, and making contact with the ball. In the case of the ballet turn, the model demonstrates a clear understanding of the structural subtleties of the movement, accurately portraying the dancer's posture as they balance on one foot and rotate their body with grace. These results collectively highlight the model's capability to generate realistic, coherent, and diverse human motions across a wide range of complex actions.

In the table, the first column presents the motion sequences generated by our model. The accompanying text above each sequence is a manually written description based on the corresponding video content. The remaining five columns display the reference frames, which are sam-

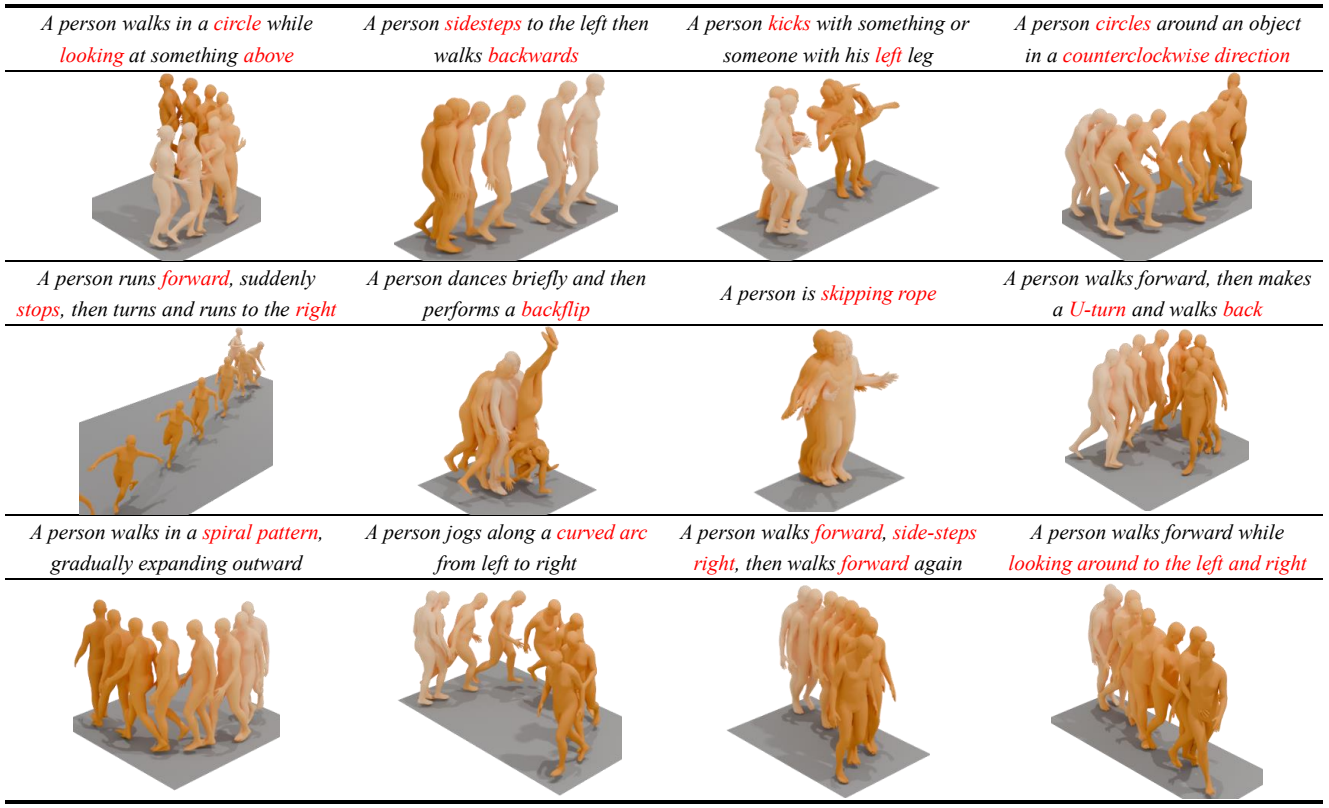

Figure 5. Qualitative experimental results. These examples cover a variety of challenging textual descriptions, involving complex action compositions and directional changes. MotionDuet is capable of generating motion sequences at a rate of approximately 199.61 poses per second during inference.

pled from the original real-life video at evenly spaced intervals.

C. Automated Video Data Cleaning

To ensure high-quality data input for downstream motion analysis tasks, we implement a robust data cleaning algorithm that filters out erroneous or low-quality video samples based on human body orientation consistency. The method utilizes pose landmarks extracted via MediaPipe and evaluates the subject's orientation through a series of geometric and kinematic criteria. The key components of the cleaning algorithm are outlined as follows:

Let a video sample $V = \{I_t\}_{t=1}^{T}$ consist of T frames. For computational efficiency, we sample a fixed subset of frames $\mathcal{F} = \{I_{t_i} \mid i = 1, 2, \ldots, N\}$ where $N \ll T$ using a uniform sampling strategy. Each frame I_{t_i} is processed by a pose estimator to extract a set of 3D landmarks $\mathbf{L}_{t_i} \in \mathbb{R}^{J\times 3}$, where J is the number of body joints.

C.1. Back-Face Consistency

Let $\vec{v}_{\text{back}} = \mathbf{L}_{\text{RShoulder}} - \mathbf{L}_{\text{LShoulder}}$ and $\vec{v}_{\text{hip}} = \mathbf{L}_{\text{RHip}} - \mathbf{L}_{\text{LHip}}$. The body orientation vector is defined as

$$\vec{v}_{\text{body}} = \frac{1}{2}(\vec{v}_{\text{back}} + \vec{v}_{\text{hip}}).$$

We also define the face direction vector as

$$\vec{v}_{\text{face}} = \mathbf{L}_{\text{Nose}} - \mathbf{L}_{\text{MidShoulder}},$$

where $\mathbf{L}_{\text{MidShoulder}} = \frac{1}{2}(\mathbf{L}_{\text{LShoulder}} + \mathbf{L}_{\text{RShoulder}})$. The body-face angle θ_{bf} is computed as

$$\theta_{\text{bf}} = \arccos\left(\frac{\vec{v}_{\text{body}} \cdot \vec{v}_{\text{face}}}{\|\vec{v}_{\text{body}}\| \cdot \|\vec{v}_{\text{face}}\|}\right).$$

A frame is valid if $\theta_{\text{bf}} \leq \theta_0$, where $\theta_0 = 20°$.

C.2. Head Pose Constraint

Let $\vec{v}_{\text{head}} = \mathbf{L}_{\text{Nose}} - \mathbf{L}_{\text{MidShoulder}}$. We constrain the head tilt angle θ_{head} against the vertical axis:

$$\theta_{\text{head}} = \arccos\left(\frac{\vec{v}_{\text{head}} \cdot \vec{e}_y}{\|\vec{v}_{\text{head}}\|}\right).$$

A frame is valid if $\theta_{\text{head}} \leq \theta_1$, with $\theta_1 = 30°$, and $\vec{e}_y$ is the global vertical axis.

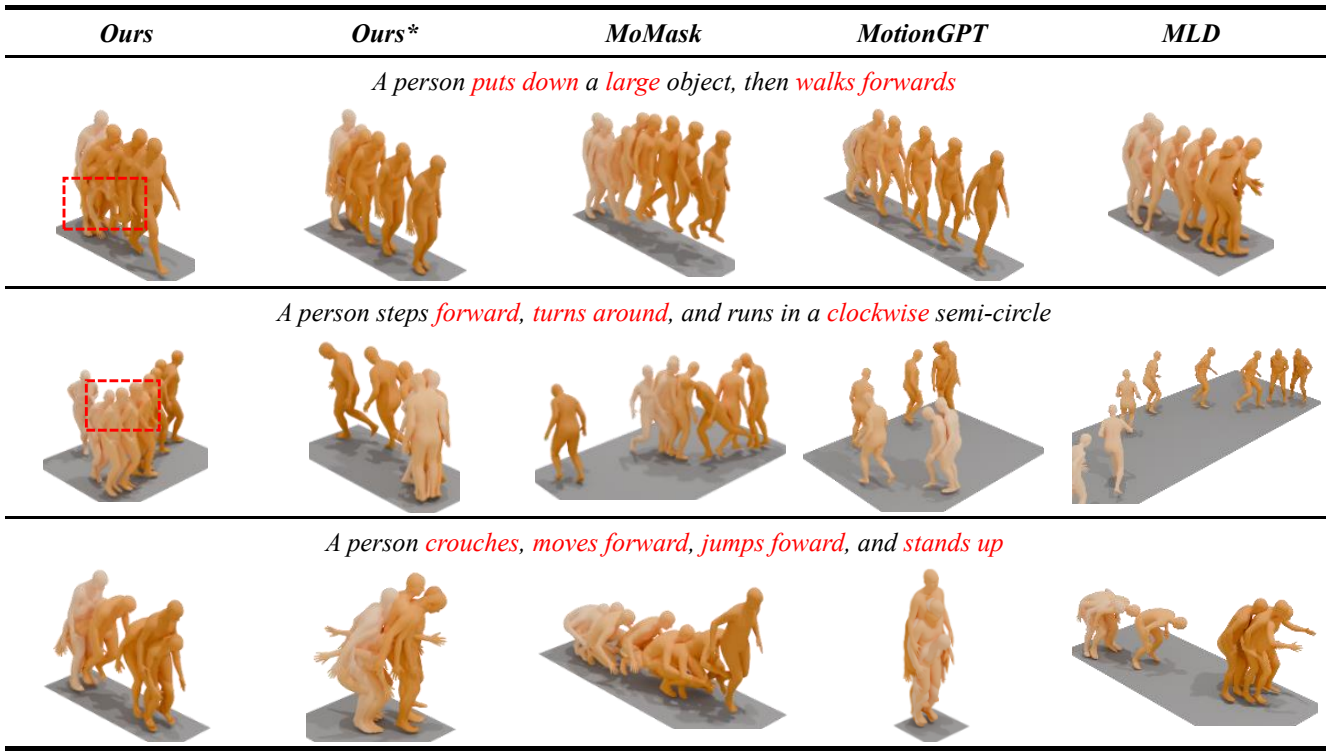

Figure 6. Comparison of qualitative experimental results. We conduct a qualitative comparison with three methods: MoMask, MotionGPT, and MLD. Compared to previous methods, our model generates more realistic and coherent motions, with better alignment to fine-grained language instructions such as "puts down a large object", "turn around", and "crouches and jumps forward". Our* denotes an ablation variant in which video inputs are excluded during training to validate their contribution to model performance. As video information is unavailable in this setting, the DASH Loss is removed accordingly, while the other components of the DUET module, excluding DMM, are retained.

C.3. Foot-Knee Direction Alignment

To ensure the plausibility of gait or standing postures, we constrain the angle between the hip-to-knee vector and the ankle-to-foot vector. For each leg side $s \in \{\text{Left}, \text{Right}\}$, we define the foot-knee angle as:

$$\theta_{\text{fk}}^{(s)} = \angle\left(\mathbf{L}_{\text{Hip}}^{(s)} - \mathbf{L}_{\text{Knee}}^{(s)},\ \mathbf{L}_{\text{Foot}}^{(s)} - \mathbf{L}_{\text{Ankle}}^{(s)}\right),$$

where $\mathbf{L}_{\text{Hip}}^{(s)}$, $\mathbf{L}_{\text{Knee}}^{(s)}$, $\mathbf{L}_{\text{Ankle}}^{(s)}$, and $\mathbf{L}_{\text{Foot}}^{(s)}$ are the coordinates of the respective joints on side s.

The frame is considered *valid* with respect to foot-knee alignment if:

$$\theta_{\text{fk}}^{(s)} \in [75^\circ,\ 180^\circ], \quad \forall s \in \{\text{Left}, \text{Right}\}$$

This constraint effectively filters out frames exhibiting unnatural foot twisting or anatomical inconsistencies, which often arise from pose tracking failures or annotation noise.

C.4. Frame Validity and Video Filtering

A frame I_{t_i} is marked as *valid* if it satisfies all of the following four constraints: (1) back-face consistency, (2) head pose constraint, and (3) foot-knee direction alignment.

Let B_i, H_i, and F_i denote Boolean indicators (1 if satisfied, 0 otherwise) for these three conditions on frame i. We define the overall video validity score as:

$$P(v) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}(B_i \wedge H_i \wedge F_i)$$

A video is considered valid if:

$$P(v) \geq \rho,$$

where $\rho = 0.7$ is the minimum acceptable ratio of valid frames.

To construct the cleaned validation dataset, we apply this automated filtering process to all raw videos. Each video is uniformly sampled into $N = 12$ frame, pose landmarks are extracted via MediaPipe, and only videos passing the

threshold are retained. This ensures that downstream models are trained on reliable, consistent human motion data, free from noisy or erroneous poses.

D. Additional Experiments

D.1. Evaluation of Hyperparameters λ_{DASH}

In this section, we first conduct a detailed analysis and discussion on the range of values for the hyperparameter λ_{DASH}, aiming to understand its influence on model performance, see Table 6. Experimental results reveal a clear trend: while introducing the DASH loss with a moderate weight can effectively improve the quality and consistency of motion generation, setting λ_{DASH} too high leads to a noticeable performance degradation. This is likely because an excessively strong DASH loss may overpower other learning signals, causing the model to overfit to the video features and thereby reducing its generalization ability, especially when video inputs are unavailable at inference time.

D.2. Study on Auto-Guidance Mechanism Weights ω

Automatic guidance identifies and corrects potential errors by measuring the discrepancy between the predictions of a strong model and a weaker one, thereby amplifying adjustments in a more favorable direction. When the two models produce similar outputs, the perturbation is negligible; however, when they diverge, the difference serves as an approximate signal toward a better sample distribution [14]. To investigate the effectiveness of our Auto Guidance under multimodal settings, we conduct an ablation study on two key factors: the modality-specific influence weights ω and the perturbation strategies—dropout and input noise. Specifically, we evaluate three groups of settings:

- **Dropout-only** configurations: $\mathcal{D}_1$ and $\mathcal{D}_2$ represent feature-level dropout rates (e.g., 5% and 10%) applied post-hoc to the base model. The guidance model operates using these degraded features to mimic a weaker model variant.
- **Noise-only** configurations: ϵ_1 and ϵ_2 indicate different levels of Gaussian noise (e.g., standard deviation increments of 5% and 10%) added to the input embeddings. This simulates corrupted conditions to encourage robust generation.

Across all groups, we systematically sweep the weighting parameter ω to determine optimal influence magnitudes for each degraded condition as shown in Table 7. We observe that dropout-only perturbation leads to more stable training compared to noise-based alternatives. This is likely because dropout removes a subset of the conditional inputs while preserving the semantic consistency of the remaining tokens. In contrast, noise injection distorts the content of the condition embeddings, potentially introducing semantic ambiguity and interfering with effective supervision. Moreover, dropout provides a natural curriculum for gradually increasing conditional strength, which is more conducive to stable convergence.

D.3. Evaluation of Loss Function

We conduct a comparative study between our proposed DASH Loss and infoNCE loss to evaluate their impact on motion generation quality. While cosine loss encourages alignment between motion and video features at the token level, it lacks explicit structural regularization and fails to preserve the internal relationships within each modality. In contrast, DASH Loss incorporates both token-level similarity and pairwise structural consistency, promoting better semantic grounding and distribution alignment. As shown in Table 8, our method achieves improved performance across all key metrics, demonstrating its effectiveness in bridging the modality gap and enhancing generation quality.

D.4. Supplementary Data on Multimodal Fusion Strategies

We provide the complete results of the ablation studies on multimodal fusion strategies for reference, see Table 9. These supplementary results offer a more comprehensive understanding of how different fusion methods perform under various conditions, and further support the analysis of the sources contributing to performance improvements.

D.5. Quantitative Evaluation of the Video Encoders

To gain deeper insights into the effectiveness and robustness of our framework, we conduct a set of ablation studies aimed at understanding the impact of fine-tuning and model scale on motion generation quality, see Table 10. These factors are critical for evaluating the model's generalization ability and its applicability under different resource constraints.

We begin by examining the role of fine-tuning. Specifically, we use the VideoMAEv2-based ViT-G model to perform motion inference directly, without applying any fine-tuning on the virtual skinned motion video dataset. This setup allows us to assess the model's zero-shot performance and its inherent capacity to generalize. Following this, we study the influence of model size by fine-tuning a smaller ViT-B model that has been distilled from the ViT-G variant, using the same training configuration. This comparison enables us to evaluate the trade-offs between model capacity, computational efficiency, and motion generation quality, providing valuable insights for selecting suitable architectures in practical scenarios.

D.6. Evaluation on Each Component

In this section, we provide additional quantitative results and analyses to complement those presented in the main

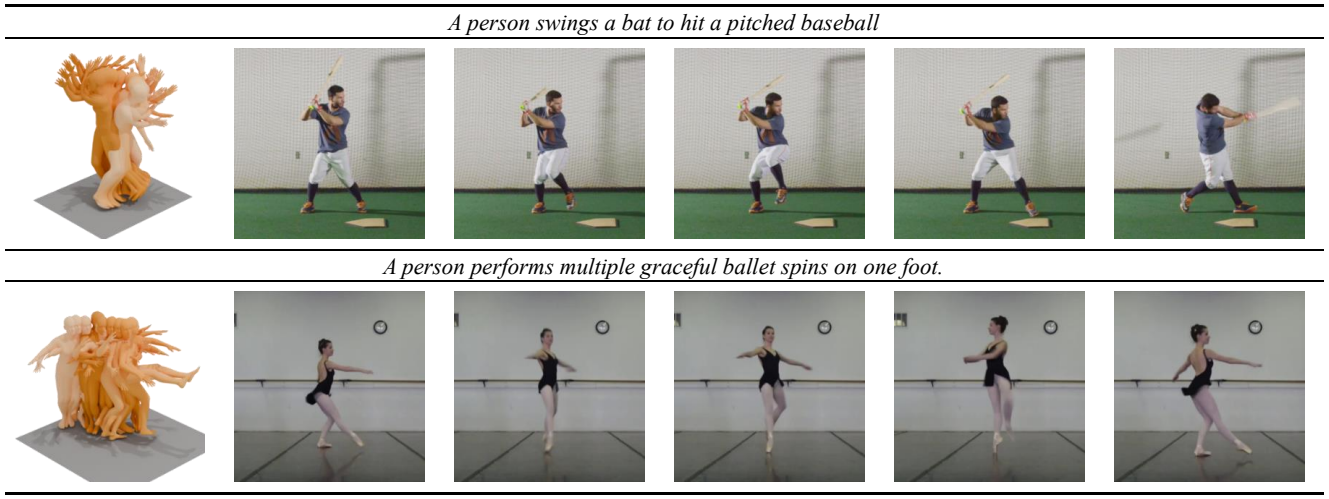

Figure 7. Qualitative results of model-generated motions for real-world videos involving complex actions. The examples include ballet spins, baseball pitching, hitting an incoming baseball with a bat, and golf swings. Although the model was never exposed to these specific videos during training, it successfully produces semantically consistent and physically plausible motions, demonstrating its ability to generalize to unseen real-world inputs.

λ_{DASH}	R Precision ↑			FID ↓	MM Dist ↓	Diversity →	MM ↑
	Top 1	Top 2	Top 3				
Real-filtering	$0.490^{\pm.003}$	$0.684^{\pm.003}$	$0.772^{\pm.002}$	$0.002^{\pm.000}$	$2.954^{\pm.010}$	$9.492^{\pm.002}$	–
0.1	$0.474^{\pm.003}$	$0.668^{\pm.003}$	$0.764^{\pm.003}$	$0.084^{\pm.012}$	$3.089^{\pm.010}$	$9.527^{\pm.071}$	$2.576^{\pm.071}$
0.3	$0.466^{\pm.003}$	$0.657^{\pm.003}$	$0.752^{\pm.002}$	$0.143^{\pm.024}$	$3.169^{\pm.010}$	$9.532^{\pm.071}$	$2.453^{\pm.018}$
0.5	$0.469^{\pm.003}$	$0.645^{\pm.003}$	$0.745^{\pm.002}$	$0.186^{\pm.024}$	$3.280^{\pm.010}$	$9.810^{\pm.071}$	$2.456^{\pm.018}$
0.7	$0.452^{\pm.003}$	$0.647^{\pm.003}$	$0.743^{\pm.002}$	$0.237^{\pm.024}$	$3.311^{\pm.010}$	$9.314^{\pm.071}$	$2.412^{\pm.018}$
0.9	$0.443^{\pm.003}$	$0.632^{\pm.003}$	$0.734^{\pm.003}$	$0.294^{\pm.012}$	$3.324^{\pm.010}$	$9.277^{\pm.071}$	$2.427^{\pm.018}$
1	$0.433^{\pm.003}$	$0.638^{\pm.003}$	$0.732^{\pm.003}$	$0.427^{\pm.024}$	$3.322^{\pm.010}$	$9.212^{\pm.071}$	$2.563^{\pm.018}$
50	$0.345^{\pm.003}$	$0.525^{\pm.003}$	$0.635^{\pm.003}$	$1.438^{\pm.024}$	$3.997^{\pm.010}$	$8.653^{\pm.071}$	$2.672^{\pm.018}$
100	$0.310^{\pm.003}$	$0.474^{\pm.003}$	$0.600^{\pm.003}$	$2.500^{\pm.012}$	$4.275^{\pm.010}$	$8.731^{\pm.071}$	$2.654^{\pm.018}$
200	$0.159^{\pm.003}$	$0.278^{\pm.003}$	$0.369^{\pm.003}$	$8.676^{\pm.012}$	$5.660^{\pm.010}$	$7.369^{\pm.071}$	$2.684^{\pm.018}$
300	$0.039^{\pm.003}$	$0.058^{\pm.003}$	$0.099^{\pm.003}$	$14.676^{\pm.012}$	$7.320^{\pm.010}$	$5.832^{\pm.071}$	$2.953^{\pm.018}$

Table 6. Parameter Study on λ_{DASH}. ↑ indicates higher is better, and ↓ indicates lower is better.

text, see Table 11. These supplementary results facilitate a more comprehensive evaluation of each component in our framework.

D.7. Inference Time

The model's inference statistics indicate approximately 7960 GFLOPs and 7932 GMACs per forward pass, representing the total number of floating-point and multiply-accumulate operations required to process each input. All inference experiments were conducted on a single NVIDIA A100 GPU with 80GB of memory. Under this configuration, the average inference time per sample (AITS) was observed to range from approximately 0.092 seconds, reflecting efficient runtime performance and effective hardware utilization, particularly in batch processing scenarios.

D.8. Model Parameter Statistics

To provide a comprehensive overview of our model architecture, we summarize the major components and their corresponding parameter counts in Table 12. The entire system consists of multiple encoders and decoders tailored for vision, text, and motion modalities. Notably, the largest component is the pretrained Vision Transformer encoder, containing 953M parameters, which remains frozen during training. Among all modules, only 21.6M parameters are trainable, ensuring efficient optimization while leveraging powerful pretrained backbones.

Overall, by freezing the majority of the parameters (1.4B non-trainable) and optimizing only a lightweight subset (21.6M trainable), our method strikes a balance between parameter efficiency and representation power.

$\mathcal{D}_1$	ω	R Precision ↑ Top 1	Top 2	Top 3	FID ↓	MM Dist ↓	Diversity →	MM ↑
Real-filtering	–	$0.490^{\pm.003}$	$0.684^{\pm.003}$	$0.772^{\pm.002}$	$0.002^{\pm.000}$	$2.954^{\pm.010}$	$9.492^{\pm.002}$	–
5%	0.75	$0.462^{\pm.005}$	$0.651^{\pm.006}$	$0.742^{\pm.005}$	$0.121^{\pm.022}$	$3.095^{\pm.014}$	$9.320^{\pm.077}$	$2.543^{\pm.065}$
	1.00	$0.469^{\pm.004}$	$0.657^{\pm.004}$	$0.744^{\pm.003}$	$0.142^{\pm.018}$	$3.082^{\pm.009}$	$9.355^{\pm.069}$	$2.580^{\pm.073}$
	1.25	$0.474^{\pm.003}$	$0.668^{\pm.003}$	$0.764^{\pm.003}$	$0.084^{\pm.012}$	$3.089^{\pm.010}$	$9.527^{\pm.071}$	$2.576^{\pm.071}$
	1.50	$0.463^{\pm.005}$	$0.654^{\pm.005}$	$0.745^{\pm.005}$	$0.102^{\pm.024}$	$3.100^{\pm.015}$	$9.310^{\pm.073}$	$2.598^{\pm.068}$
	1.75	$0.469^{\pm.004}$	$0.657^{\pm.004}$	$0.749^{\pm.003}$	$0.097^{\pm.018}$	$3.082^{\pm.009}$	$9.355^{\pm.069}$	$2.513^{\pm.073}$
$\mathcal{D}_2$	ω	R Precision ↑ Top 1	Top 2	Top 3	FID ↓	MM Dist ↓	Diversity →	MM ↑
10%	0.75	$0.468^{\pm.004}$	$0.662^{\pm.003}$	$0.755^{\pm.004}$	$0.102^{\pm.020}$	$3.090^{\pm.012}$	$9.480^{\pm.075}$	$2.572^{\pm.068}$
	1.0	$0.473^{\pm.003}$	$0.666^{\pm.003}$	$0.758^{\pm.003}$	$0.132^{\pm.019}$	$3.082^{\pm.011}$	$9.503^{\pm.070}$	$2.579^{\pm.071}$
	1.25	$0.474^{\pm.003}$	$0.668^{\pm.003}$	$0.760^{\pm.003}$	$0.153^{\pm.018}$	$3.089^{\pm.010}$	$9.510^{\pm.072}$	$2.580^{\pm.069}$
	1.50	$0.471^{\pm.004}$	$0.663^{\pm.003}$	$0.756^{\pm.004}$	$0.113^{\pm.022}$	$3.088^{\pm.011}$	$9.495^{\pm.071}$	$2.575^{\pm.070}$
	1.75	$0.469^{\pm.003}$	$0.661^{\pm.004}$	$0.753^{\pm.003}$	$0.323^{\pm.021}$	$3.085^{\pm.010}$	$9.485^{\pm.073}$	$2.570^{\pm.068}$
ϵ_1	ω	R Precision ↑ Top 1	Top 2	Top 3	FID ↓	MM Dist ↓	Diversity →	MM ↑
5%	0.75	$0.458^{\pm.005}$	$0.645^{\pm.005}$	$0.735^{\pm.004}$	$0.102^{\pm.023}$	$3.095^{\pm.013}$	$9.315^{\pm.074}$	$2.727^{\pm.069}$
	1.00	$0.464^{\pm.004}$	$0.650^{\pm.004}$	$0.740^{\pm.004}$	$0.132^{\pm.022}$	$3.090^{\pm.012}$	$9.345^{\pm.070}$	$2.575^{\pm.070}$
	1.25	$0.467^{\pm.004}$	$0.653^{\pm.003}$	$0.743^{\pm.003}$	$0.101^{\pm.020}$	$3.088^{\pm.011}$	$9.355^{\pm.071}$	$2.576^{\pm.068}$
	1.50	$0.466^{\pm.004}$	$0.654^{\pm.004}$	$0.745^{\pm.004}$	$0.173^{\pm.021}$	$3.090^{\pm.011}$	$9.350^{\pm.069}$	$2.573^{\pm.071}$
	1.75	$0.462^{\pm.004}$	$0.648^{\pm.004}$	$0.737^{\pm.004}$	$0.152^{\pm.023}$	$3.092^{\pm.012}$	$9.338^{\pm.072}$	$2.571^{\pm.070}$
ϵ_1	ω	R Precision ↑ Top 1	Top 2	Top 3	FID ↓	MM Dist ↓	Diversity →	MM ↑
10%	0.75	$0.446^{\pm.005}$	$0.643^{\pm.005}$	$0.726^{\pm.004}$	$0.143^{\pm.023}$	$3.135^{\pm.013}$	$9.853^{\pm.074}$	$2.767^{\pm.069}$
	1.00	$0.461^{\pm.004}$	$0.643^{\pm.004}$	$0.737^{\pm.004}$	$0.134^{\pm.022}$	$3.103^{\pm.012}$	$9.338^{\pm.070}$	$2.687^{\pm.070}$
	1.25	$0.465^{\pm.004}$	$0.646^{\pm.003}$	$0.739^{\pm.003}$	$0.165^{\pm.020}$	$3.132^{\pm.011}$	$9.285^{\pm.071}$	$2.523^{\pm.068}$
	1.50	$0.461^{\pm.004}$	$0.649^{\pm.004}$	$0.742^{\pm.004}$	$0.198^{\pm.021}$	$3.138^{\pm.011}$	$9.380^{\pm.069}$	$2.543^{\pm.071}$
	1.75	$0.454^{\pm.004}$	$0.643^{\pm.004}$	$0.737^{\pm.004}$	$0.182^{\pm.023}$	$3.132^{\pm.012}$	$9.398^{\pm.072}$	$2.592^{\pm.070}$
CFG	ω	R Precision ↑ Top 1	Top 2	Top 3	FID ↓	MM Dist ↓	Diversity →	MM ↑
	6.5	$0.457^{\pm.004}$	$0.656^{\pm.004}$	$0.737^{\pm.004}$	$0.133^{\pm.023}$	$3.088^{\pm.012}$	$9.285^{\pm.072}$	$2.523^{\pm.070}$

Table 7. Parameter Study on ω and Dropout. ↑ indicates higher is better, and ↓ lower is better.

Method	R Precision ↑ Top 1	Top 2	Top 3	FID ↓	MM Dist ↓	Diversity →	MM ↑
Real-filtering	$0.490^{\pm.003}$	$0.684^{\pm.003}$	$0.772^{\pm.002}$	$0.002^{\pm.000}$	$2.954^{\pm.010}$	$9.492^{\pm.002}$	–
infoNCE loss	$0.458^{\pm.003}$	$0.642^{\pm.003}$	$0.746^{\pm.003}$	$1.773^{\pm.012}$	$3.131^{\pm.010}$	$9.583^{\pm.071}$	$2.632^{\pm.071}$
Token-wise Margin Loss	$0.473^{\pm.003}$	$0.665^{\pm.003}$	$0.756^{\pm.003}$	$0.096^{\pm.012}$	$3.102^{\pm.010}$	$9.534^{\pm.071}$	$2.535^{\pm.071}$
DASH Loss	$0.474^{\pm.003}$	$0.668^{\pm.003}$	$0.762^{\pm.003}$	$0.084^{\pm.012}$	$3.089^{\pm.010}$	$9.527^{\pm.071}$	$2.576^{\pm.071}$

Table 8. Evaluation of Loss Function. ↑ indicates higher is better, ↓ indicates lower is better, and → indicates closer is better.

E. Video Motion Dataset

E.1. Overview of the HumanML3D Dataset

The HumanML3D dataset provides a comprehensive and standardized representation of human motion, focusing on skeleton-level analysis. Each motion sequence is stored as a NumPy array with 263-dimensional features per frame, capturing both rotation-invariant and rotation-related information, including joint positions, velocities, angular changes, and joint rotations. Instead of using raw Skinned Multi-Person Linear (SMPL) parameters, the dataset represents motion through a consistent 22-joint skeleton structure with

Method	R Precision ↑ Top 1	Top 2	Top 3	FID ↓	MM Dist ↓	Diversity →	MM ↑
Real-filtering	$0.490^{\pm.003}$	$0.684^{\pm.003}$	$0.772^{\pm.002}$	$0.002^{\pm.000}$	$2.954^{\pm.010}$	$9.492^{\pm.002}$	–
Concat	$0.463^{\pm.003}$	$0.652^{\pm.003}$	$0.742^{\pm.003}$	$0.192^{\pm.012}$	$3.296^{\pm.010}$	$9.687^{\pm.071}$	$2.412^{\pm.071}$
+ Cross-Attn	$0.380^{\pm.002}$	$0.568^{\pm.002}$	$0.684^{\pm.002}$	$0.707^{\pm.024}$	$3.652^{\pm.010}$	$9.308^{\pm.071}$	$\mathbf{3.276^{\pm.018}}$
Concat	$0.463^{\pm.003}$	$0.652^{\pm.003}$	$0.742^{\pm.003}$	$0.192^{\pm.012}$	$3.296^{\pm.010}$	$9.687^{\pm.071}$	$2.412^{\pm.071}$
+ FFT	$0.430^{\pm.003}$	$0.626^{\pm.003}$	$0.726^{\pm.003}$	$0.364^{\pm.012}$	$3.392^{\pm.010}$	$9.703^{\pm.071}$	$2.640^{\pm.071}$
Concat	$0.463^{\pm.003}$	$0.652^{\pm.003}$	$0.742^{\pm.003}$	$0.192^{\pm.012}$	$3.296^{\pm.010}$	$9.687^{\pm.071}$	$2.412^{\pm.071}$
+ DMM	$0.466^{\pm.002}$	$0.651^{\pm.002}$	$0.749^{\pm.002}$	$0.131^{\pm.024}$	$3.132^{\pm.010}$	$9.643^{\pm.071}$	$2.346^{\pm.018}$
Concat	$0.463^{\pm.003}$	$0.652^{\pm.003}$	$0.742^{\pm.003}$	$0.192^{\pm.012}$	$3.296^{\pm.010}$	$9.687^{\pm.071}$	$2.412^{\pm.071}$
+ Self-Attn	$0.433^{\pm.003}$	$0.622^{\pm.003}$	$0.714^{\pm.003}$	$0.222^{\pm.012}$	$3.314^{\pm.010}$	$9.763^{\pm.071}$	$2.380^{\pm.071}$
+ DMM	$0.432^{\pm.003}$	$0.613^{\pm.003}$	$0.721^{\pm.003}$	$0.228^{\pm.012}$	$3.320^{\pm.010}$	$9.737^{\pm.071}$	$2.390^{\pm.071}$
Hadamard Product	$0.441^{\pm.003}$	$0.636^{\pm.003}$	$0.741^{\pm.003}$	$0.243^{\pm.012}$	$3.219^{\pm.010}$	$9.393^{\pm.071}$	$2.370^{\pm.071}$
+ FFT	$0.443^{\pm.003}$	$0.661^{\pm.003}$	$0.743^{\pm.003}$	$0.292^{\pm.012}$	$3.226^{\pm.010}$	$9.319^{\pm.071}$	$2.434^{\pm.071}$
+ DMM	$0.438^{\pm.003}$	$0.623^{\pm.003}$	$0.727^{\pm.003}$	$0.280^{\pm.012}$	$3.311^{\pm.010}$	$9.901^{\pm.071}$	$2.347^{\pm.071}$
Element-Wise Addition	$0.452^{\pm.003}$	$0.632^{\pm.003}$	$0.747^{\pm.003}$	$0.168^{\pm.012}$	$3.388^{\pm.010}$	$9.617^{\pm.017}$	$2.321^{\pm.071}$
+ FFT	$0.435^{\pm.003}$	$0.634^{\pm.003}$	$0.743^{\pm.003}$	$0.204^{\pm.012}$	$3.256^{\pm.010}$	$9.430^{\pm.017}$	$2.352^{\pm.071}$
Element-Wise Addition	$0.452^{\pm.003}$	$0.632^{\pm.003}$	$0.747^{\pm.003}$	$0.168^{\pm.012}$	$3.388^{\pm.010}$	$9.617^{\pm.017}$	$2.321^{\pm.071}$
+ DMM	$0.451^{\pm.003}$	$0.643^{\pm.003}$	$0.750^{\pm.003}$	$0.204^{\pm.012}$	$3.256^{\pm.010}$	$9.445^{\pm.017}$	$2.402^{\pm.071}$
+ FFT	$0.435^{\pm.003}$	$0.630^{\pm.003}$	$0.744^{\pm.003}$	$0.254^{\pm.012}$	$3.299^{\pm.010}$	$9.624^{\pm.017}$	$2.402^{\pm.071}$
Element-Wise Addition	$0.452^{\pm.003}$	$0.632^{\pm.003}$	$0.747^{\pm.003}$	$0.168^{\pm.012}$	$3.388^{\pm.010}$	$9.617^{\pm.017}$	$2.321^{\pm.071}$
+ DMM	$0.451^{\pm.003}$	$0.643^{\pm.003}$	$0.750^{\pm.003}$	$0.204^{\pm.012}$	$3.256^{\pm.010}$	$9.445^{\pm.017}$	$2.402^{\pm.071}$
+ FFT	$0.453^{\pm.003}$	$0.648^{\pm.003}$	$0.750^{\pm.003}$	$0.163^{\pm.012}$	$3.178^{\pm.010}$	$9.691^{\pm.071}$	$2.447^{\pm.071}$
+ Identity	$0.459^{\pm.003}$	$0.652^{\pm.003}$	$0.752^{\pm.003}$	$0.147^{\pm.012}$	$3.124^{\pm.010}$	$9.677^{\pm.071}$	$2.347^{\pm.071}$
+ Conv (DUET)	$\mathbf{0.473^{\pm.003}}$	$\mathbf{0.664^{\pm.003}}$	$\mathbf{0.755^{\pm.003}}$	$\mathbf{0.101^{\pm.024}}$	$\mathbf{3.087^{\pm.010}}$	$\mathbf{9.472^{\pm.071}}$	$2.460^{\pm.071}$

Table 9. Performance comparison of different multimodal fusion strategies. Table indentation denotes the sequential integration of modules, with each indented block representing a component appended downstream within the overall architecture. The top results in each column are highlighted with **bold** (best).

Method	R Precision ↑ Top 1	Top 2	Top 3	FID ↓	MM Dist ↓	Diversity →	MM ↑
Real	$0.511^{\pm.003}$	$0.703^{\pm.003}$	$0.797^{\pm.003}$	$0.002^{\pm.000}$	$2.974^{\pm.008}$	$9.503^{\pm.000}$	—
MLD (Baseline)	$0.481^{\pm.003}$	$0.673^{\pm.003}$	$0.772^{\pm.002}$	$0.473^{\pm.013}$	$3.196^{\pm.010}$	$9.724^{\pm.082}$	$2.413^{\pm.079}$
VIT-G with fine-turing	$0.497^{\pm.003}$	$0.698^{\pm.003}$	$0.795^{\pm.003}$	$0.179^{\pm0.024}$	$3.154^{\pm.010}$	$9.532^{\pm.080}$	$2.496^{\pm.018}$
VIT-G without fine-turing	$0.446^{\pm.003}$	$0.643^{\pm.003}$	$0.751^{\pm.003}$	$0.238^{\pm.024}$	$3.334^{\pm.010}$	$9.653^{\pm.071}$	$2.654^{\pm.018}$
VIT-B with fine-turing	$0.486^{\pm.003}$	$0.679^{\pm.003}$	$0.782^{\pm.003}$	$0.182^{\pm.012}$	$3.178^{\pm.010}$	$9.574^{\pm.071}$	$2.438^{\pm.071}$

Table 10. Performance Assessment of the Video Encoders. ↑ indicates higher is better, ↓ indicates lower is better, and → indicates closer is better.

normalized body shape across all samples. By intentionally excluding skinned mesh data, textures, and clothing, HumanML3D emphasizes clean skeletal motion suitable for tasks involving motion understanding rather than detailed 3D surface rendering.

Motion Data Representation. The HumanML3D dataset offers an extensive repository of motion data seamlessly integrated with vivid natural language descriptions, stored as NumPy arrays and text files. Each motion sequence is elegantly organized as an M×263 matrix, where M signifies the number of frames. Every 263-dimensional feature vector per frame encapsulates a sophisticated array of rotation-invariant and rotation-related attributes, including root joint angular velocity, translation velocity, vertical displacement, local joint positions and velocities, 6D joint rotation representations, and binary foot contact indicators, forming a robust foundation for advanced motion analysis.

Standardized Joint-Based Motion Features. Uniquely, HumanML3D refrains from including raw SMPL parameters such as pose, shape, or translation. Instead, it transforms motion data into standardized joint sequences and derived features, utilizing the 22-joint structure of the SMPL skeleton to precisely articulate human poses. Each frame is defined by accurate 3D coordinates for these joints. Shape parameters are deliberately uniform, with all motion data normalized to a consistent human template, ensuring no

	R Precision ↑			FID ↓	MM Dist ↓	Diversity →	MM ↑
	Top 1	Top 2	Top 3				
Real	$0.511^{\pm.003}$	$0.703^{\pm.003}$	$0.797^{\pm.003}$	$0.002^{\pm.000}$	$2.974^{\pm.008}$	$9.503^{\pm.065}$	–
Baseline	$0.481^{\pm.003}$	$0.673^{\pm.003}$	$0.772^{\pm.002}$	$0.473^{\pm.024}$	$3.196^{\pm.010}$	$9.724^{\pm.071}$	$2.413^{\pm.018}$
+ Filtering	$0.446^{\pm.003}$	$0.628^{\pm.003}$	$0.734^{\pm.002}$	$0.396^{\pm.024}$	$3.156^{\pm.010}$	$9.710^{\pm.071}$	$2.433^{\pm.018}$
Real-filtering	$0.490^{\pm.003}$	$0.684^{\pm.003}$	$0.772^{\pm.002}$	$0.002^{\pm.000}$	$2.954^{\pm.010}$	$9.492^{\pm.081}$	–
+ Video	$0.463^{\pm.003}$	$0.652^{\pm.003}$	$0.742^{\pm.003}$	$0.192^{\pm.012}$	$3.296^{\pm.010}$	$9.687^{\pm.071}$	$2.412^{\pm.071}$
+ DUET	$0.473^{\pm.003}$	$0.664^{\pm.003}$	$0.755^{\pm.003}$	$0.101^{\pm.024}$	$\mathbf{3.087^{\pm.010}}$	$9.472^{\pm.071}$	$2.460^{\pm.071}$
+ DASH Loss	$\mathbf{0.474^{\pm.003}}$	$\mathbf{0.668^{\pm.003}}$	$\mathbf{0.764^{\pm.003}}$	$\mathbf{0.084^{\pm.012}}$	$3.089^{\pm.010}$	$\mathbf{9.527^{\pm.071}}$	$\mathbf{2.576^{\pm.071}}$

Table 11. Evaluation on each component. The top results are highlighted in each column with **bold**.

Module Name	Component	Param. Count
pretrainVisionTransformerEncoder	VisionTransformer	953 M
text_encoder	MldTextEncoder	427 M
vae	MldVae	18.8 M
denoiser	MldDenoiser	21.6 M
t2m_textencoder	TextEncoderBiGRUCo	4.1 M
t2m_moveencoder	MovementConvEncoder	1.8 M
t2m_motionencoder	MotionEncoderBiGRUCo	15.7 M

Table 12. Model components and parameter statistics. "Trainable params" refer to parameters updated during training.

variations in body shape across samples for streamlined analysis.

Skeleton-Level Data Without Skinning. HumanML3D is intentionally crafted to focus exclusively on skeleton-level motion data, explicitly excluding skinned 3D body mesh models or skinning processes. It omits mesh vertex sequences, FBX files, texture maps, and clothing models, prioritizing skeletal motion data and its associated feature representations over skinned vertex clouds or fully animated mesh sequences. This deliberate exclusion of skinning underscores HumanML3D's suitability for applications centered on skeletal motion analysis rather than detailed 3D mesh rendering or skinning-dependent visualizations.

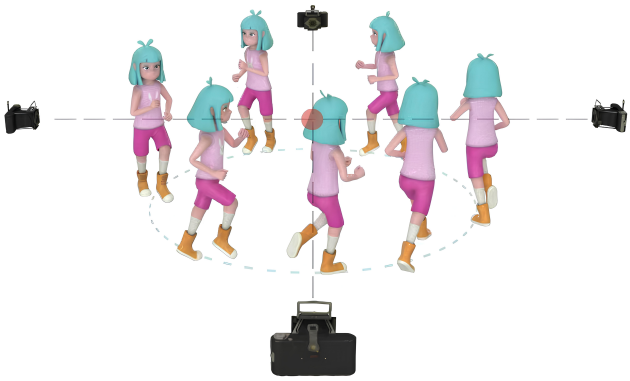

Figure 8. Video motion dataset creation workflow visualization.

E.2. HumanML3D Visualization

To achieve visualization and in-depth analysis of the HumanML3D dataset, we first converted the .npy files into Biovision Hierarchy (BVH) format for convenient visualization using Blender software. However, skeletal-based virtual human motions often lack realism. Therefore, we further converted the BVH-format data into SMPL model format and applied skinning to enhance the visual authenticity of the motion. Notably, the BVH format utilizes a skeletal structure comprising 17 joints, whereas the SMPL model includes 22 joints. The five additional joints in the SMPL model correspond to vertices at the extremities of the limbs and the top of the head. During conversion, to ensure compatibility, the values for these additional joints were set to zero. After generating the SMPL models, we assigned skinning weights and standardized the initial human shape to an A-pose to maintain consistency and standardization.

Subsequently, we converted the SMPL-format data into FBX format and utilized Blender software to set up four virtual cameras, capturing the motion sequences comprehensively from the east, south, west, and north directions, see Fig. 8. This process yielded a total of 116,800 video motion videos. To ensure high data quality, we employed the data cleaning approach described in Appendix B, ultimately obtaining 71,220 video motion videos with limited

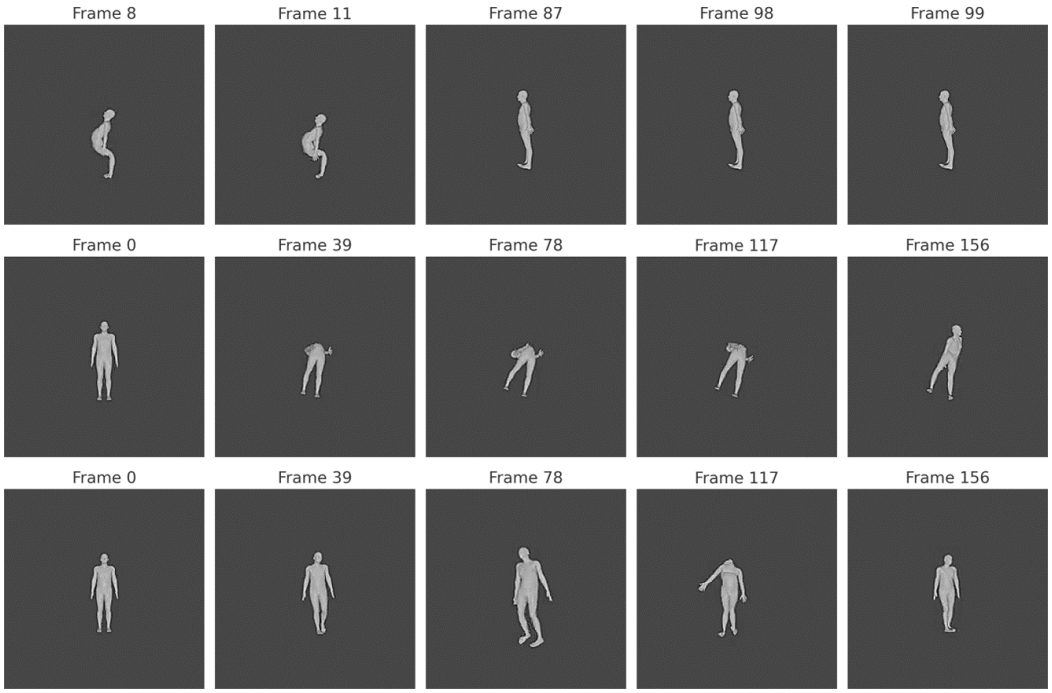

Figure 9. Skinning errors.

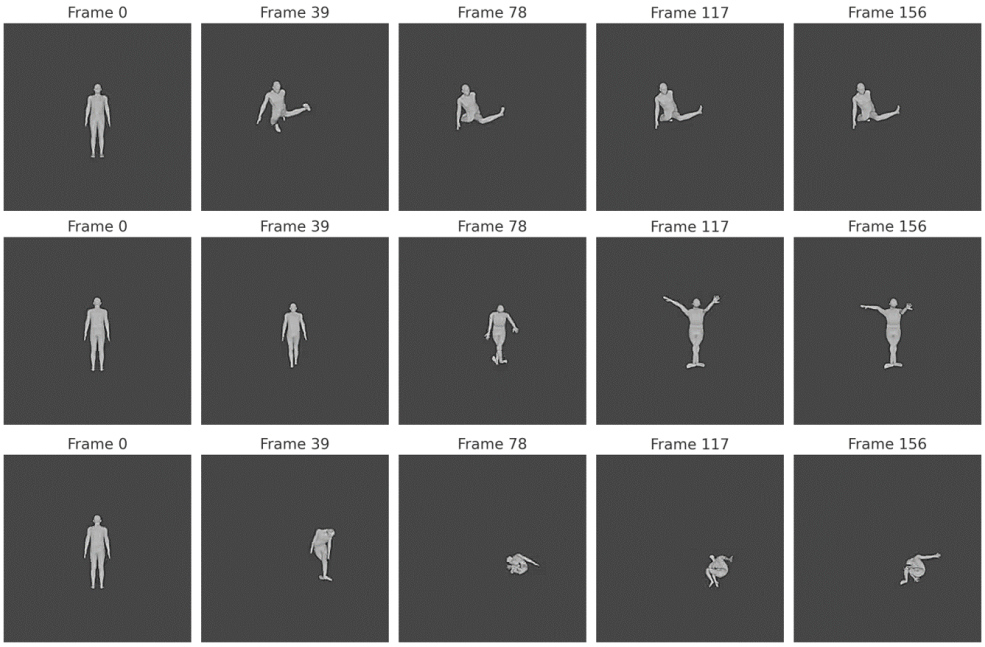

Figure 10. Data quality issues.

but inevitable errors, accounting for approximately 61% of the original dataset. The entire process took 45 days to complete, utilizing four NVIDIA RTX 4090 GPUs to ensure efficient and high-fidelity rendering and processing.

E.3. Anomaly Data Analysis

Based on the analysis results, anomalous motion samples account for 39% of the entire dataset. These refer to video clips that, after automatic preprocessing by our data-

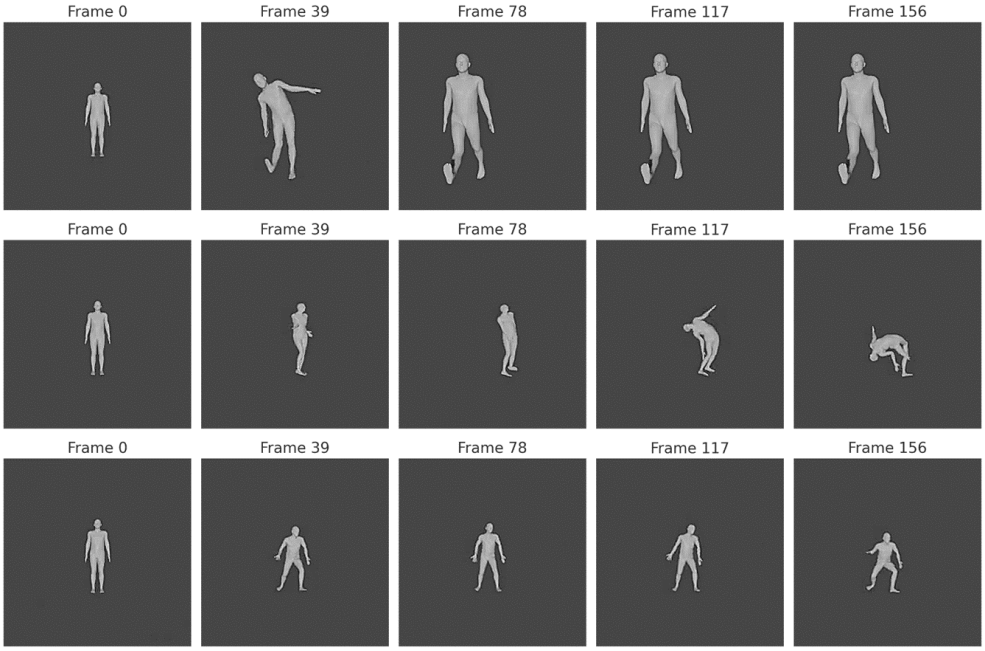

Figure 11. Mild deviations in the motion itself.

cleaning script, still contain artifacts or motion inconsistencies, and are thus categorized as anomalous motion samples. Through visualization analysis, we were able to identify these anomalous samples and began investigating the reasons behind such a high anomaly rate. We categorized the anomalous data into three main types:

- **Skinning errors**, which result in incorrect or inverted skin deformations of the human body, as shown in Fig. 9;
- **Data quality issues**, where the overall motion appears generally normal but contains locally unbalanced or disproportionate movements, as shown in Fig. 10;
- **Mild deviations in the motion itself**, where the motion sequence displays subtle but noticeable unnatural or unrealistic elements, as shown in Fig. 11.

In the future, we plan to further improve our visualization methods by integrating more advanced techniques to gain a deeper understanding of and better monitor the quality of motion generation. These enhancements will help us identify deficiencies in the data creation process and guide the refinement of both generation and curation pipelines. Ultimately, this will facilitate the production of higher-quality motion samples.