MotionCtrl: A Real-time Controllable Vision-Language-Motion Model

Bin Cao[1,2,3†] Sipeng Zheng[7†] Ye Wang[4] Lujie Xia[6] Qianshan Wei[5]
Qin Jin[4] Jing Liu[1,2] Zongqing Lu[6,7‡]

[1]Institute of Automation, Chinese Academy of Sciences [2]University of Chinese Academy of Sciences
[3]Beijing Academy of Artificial Intelligence [4]Renmin University of China
[5]Southeast University [6]Peking University [7]BeingBeyond

Abstract

Human motion generation involves synthesizing coherent human motion sequences conditioned on diverse multimodal inputs and holds significant potential for real-world applications. Despite recent advancements, existing vision-language-motion models (VLMMs) remain limited in achieving this goal. In this paper, we identify the lack of controllability as a critical bottleneck, where VLMMs struggle with diverse human commands, pose initialization, generation of long-term or unseen cases, and fine-grained control over individual body parts. To address these challenges, we introduce MotionCtrl, the first real-time, controllable VLMM with state-of-the-art performance. MotionCtrl achieves its controllability through training on HuMo100M, the largest human motion dataset to date, featuring over 5 million self-collected motions, 100 million multi-task instructional instances, and detailed part-level descriptions that address a long-standing gap in the field. Additionally, we propose a novel part-aware residual quantization technique for motion tokenization, enabling precise control over individual body parts during motion generation. Extensive experiments demonstrate MotionCtrl's superior performance across a wide range of motion benchmarks. Furthermore, we provide strategic design insights and a detailed time efficiency analysis to guide the development of practical motion generators.

1. Introduction

Motion generation has received increasing attention due to its potential applications in video games, film production, and humanoid control. However, current human motion generators [11, 13] face challenges in achieving real-time inference speed and controllability, limiting these models from becoming practical in reality. Controllability here refers to handling user commands, random initial poses, long-term or unseen motions, as well as part-aware control. While methods trained on narrow datasets [10, 18] perform well in fixed scenarios (e.g., text-to-motion), they falter outside these bounds. Inspired by large vision-language models (VLMs) [16, 19], recent efforts have developed vision-language-motion models (VLMMs) using multi-modality and multi-task training. These models [12] have enhanced motion generation capabilities, with some incorporating visual cues [24] for better motion understanding. Despite these advances, achieving full controllability remains a challenge, driving the motivation for this work.

For VLMs, leveraging massive multimodal data is crucial to unlocking their potential. However, motion generation is hindered by the scarcity of high-quality motion data. Recent efforts [23, 38] have explored extracting motions from web videos to build larger datasets, but most fail to utilize the collected data beyond mere scaling. This paper overcomes this problem by introducing **HuMo100M**, the largest motion generation dataset to date, with over 5 million motions and 100 million instructions for tasks like Instruct-to-Motion. HuMo100M offers three key advantages over previous datasets: (1) Part-level descriptions: These provide fine-grained supervision for part control, enabling precise alignment with part-level motions while masking low-quality, occluded, or blurred segments to improve data reliability. (2) Long-term motions: We propose a motion concatenation method that combines individual motions into continuous, spatiotemporally consistent sequences, enabling VLMMs to generate realistic, extended motions beyond short-term snippets. (3) Text-aligned visual clips: Unlike prior works [47], we argue that visual cues are particularly beneficial for Internet-collected motions. Even with unreliable motion data, VLLMs can learn weak supervision through the alignment of visual and textual contexts.

Building on this dataset, we introduce MotionCtrl, the

† Equal contribution.
‡ Correspondence to <zongqing.lu@pku.edu.cn >

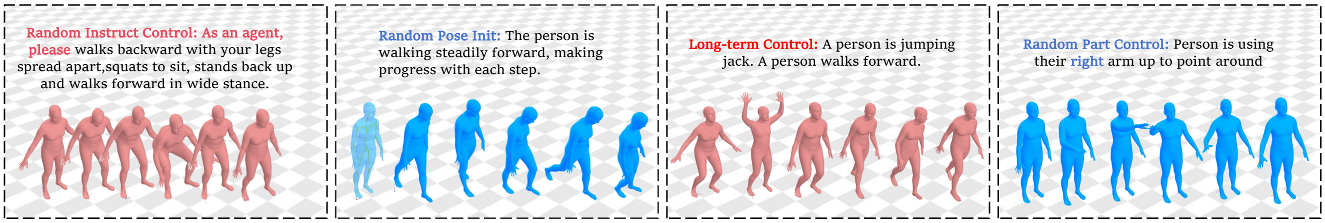

Random Instruct Control: As an agent, please walks backward with your legs spread apart,squats to sit, stands back up and walks forward in wide stance.
Random Pose Init: The person is walking steadily forward, making progress with each step.
Long-term Control: A person is jumping jack. A person walks forward.
Random Part Control: Person is using their right arm up to point around

Figure 1. Built on our million-scale dataset HuMo100M, we present MotionCtrl, the first real-time, controllable vision-language-motion model (VLMM), achieving high performance and practical efficiency. MotionCtrl supports controllability via random instructions, initial poses, long-term generation, unseen motions, and part-aware motion control.

first real-time, controllable VLMM achieving state-of-the-art performance across diverse motion benchmarks. Leveraging HuMo100M's million-scale instruction data and multi-modal inputs, MotionCtrl explores key design choices for practical time-efficient motion generation (e.g., motion decoding order), a topic rarely addressed before. We emphasize the importance of part-level control in human activities, a challenge for existing methods due to their reliance on single code embeddings for the entire body and the absence of part labels. Inspired by residual vector quantization (RQ) [11, 15], we propose part-aware RQ (PRQ) for motion tokenization, which splits whole-body motion features into shared-joint parts and quantizes them as discrete part-level codes. Unlike Guo et al. [11], MotionCtrl decodes motion codes frame-by-frame for real-time generation.

Our key contributions are as follows:

- We present **HuMo100M**, the largest multimodal motion dataset to date, including 5 million motions and 100 million multi-task motion instructions with fine-grained, long-form, and part-level motion labels.
- We propose **MotionCtrl**, a highly controllable VLMM that outperforms existing models, with insights into key design choices and architecture.
- To enable part-level control in MotionCtrl, we develop part-aware residual quantization, utilizing HuMo100M's part labels for fine-grained motion control in our VLMM.

2. Related Work

Human Motion Generation. This task is categorized by control signals, such as text descriptions [10, 28], action labels [1], keyframe poses [48], and incomplete motion sequences [37]. Early deterministic T2M methods often produced blurry results [8, 9], while later approaches used stochastic techniques like VAEs [2] or GANs [39] to mitigate this. Recently, works like [13, 37] have integrated large language models (LLMs) to interpret human intent. MotionChain [14] leverages LLMs for multi-turn conversational motion or text generation. Chen et al. [4] introduced MotionLLM, a unified framework for motion understanding, captioning, and reasoning. Similar efforts include LMM [47] and MotionGPT [48]. Further progress, such as Luo et al. [24], explore human-centric videos to enhance motion understanding, with M^{3}GPT employing multimodal tokenizers for text, motion, and music encoding. However, prior research often neglects motion generation controllability and fails to balance model size with time efficiency.

Motion Tokenization. Existing works often use vector quantization (VQ) [35] for human motion representation. Beyond standard VQ, recent advances include residual quantization (RQ) [11], hierarchical quantization (HQ) [22, 42] lookup-free quantization (LFQ) [38, 43], and finite scalar quantization (FSQ) [23, 27], all showing significant improvements in motion representation. Recent work also explores part-level motion tokenization. For example, Chen et al. [3] split the body into upper and lower parts, while Lu et al. [22] and Zhang et al. [48] focused on body and hand components. However, these methods lack independent limb control and corresponding textual labels or benchmarks, motivating our development of a real-time, part-controllable motion tokenizer.

3. The MotionCtrl Model

We present MotionCtrl, a 7B-parameter vision-language-motion model (VLMM) trained on 5 million human motions and 100 million motion instructional instances, as illustrated in Figure 2. Despite recent progress in motion generation, critical questions about real-time, controllable VLMMs remain unresolved. To address this, we first offer a concise model overview (Section 3.1), followed by detailed discussions on controllable motion generation (Section 3.2) and real-time design choices (Section 3.3).

3.1. Overview of VLMM

Our VLMM is built on the LLaVA-video-7B framework [49]. Similar to recent VLMs [6, 20], LLaVA-video consists of three components: a 400M visual encoder (SigLIP [44]), a 2-layer MLP for visual projection, and a 7B LLaMA-2-chat backbone [33]. To efficiently process more frames, we adopt the slow-fast strategy from Zhang et al. [49] to reduce visual tokens. Our VLMM treats human motion as a foreign language. Given a motion sequence $m_{1:T}$, where $m_i \in \mathbb{R}^D$ represents a D-dim motion feature and

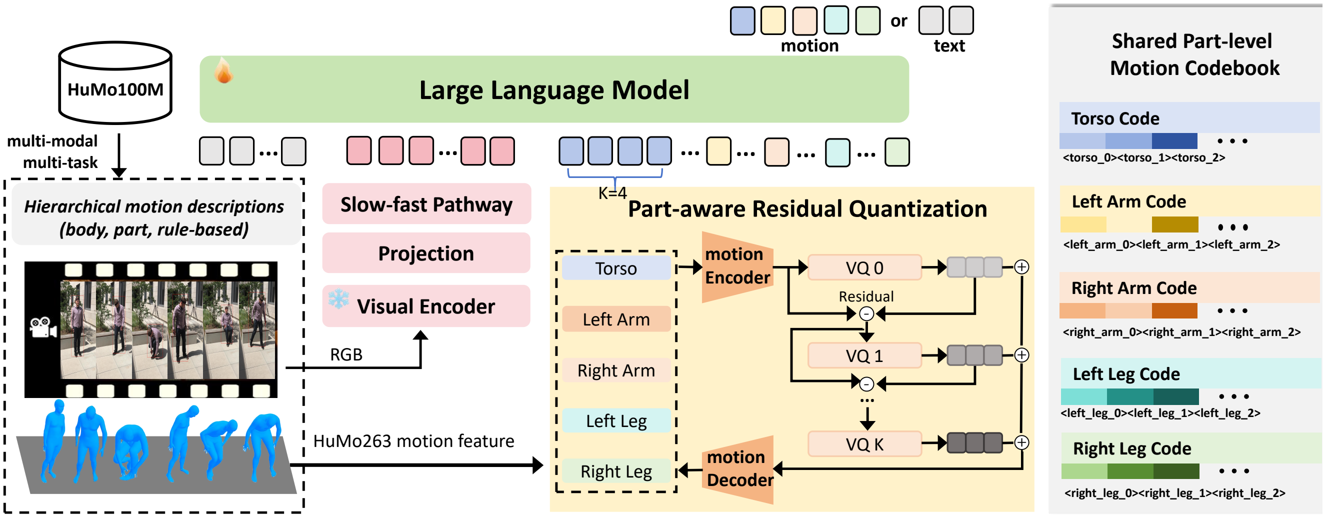

motion
or
text
HuMo100M
multi-modal
multi-task
Large Language Model
Hierarchical motion descriptions (body, part, rule-based)
Slow-fast Pathway
Projection
Visual Encoder
RGB
HuMo263 motion feature
K=4
Part-aware Residual Quantization
Torso
Left Arm
Right Arm
Left Leg
Right Leg
motion Encoder
motion Decoder
VQ 0
Residual
VQ 1
VQ K
Shared Part-level Motion Codebook
Torso Code
<torso_0><torso_1><torso_2>
Left Arm Code
<left_arm_0><left_arm_1><left_arm_2>
Right Arm Code
<right_arm_0><right_arm_1><right_arm_2>
Left Leg Code
<left_leg_0><left_leg_1><left_leg_2>
Right Leg Code
<right_leg_0><right_leg_1><right_leg_2>

Figure 2. **Model Illustration.** MotionCtrl supports multi-modal inputs/outputs, built on a 7B LLM backbone. It employs SigLIP+2MLP for visual encoding and projection with a slow-fast strategy, alongside part-aware residual quantization for motion tokenization.

T is the frame number, we use a motion tokenizer $\mathcal{Q}$ to quantize the sequence into discrete tokens. We extend the VLM vocabulary with K additional motion codes and introduce special token <mot> and </mot> to mark motion sequence boundaries. Each training sample is an instruction-following instance $\{\mathcal{X}_Q, \mathcal{X}_A\}$, representing a User-VLMM interaction. Paired or interleaved vision, language, and motion data are curated from web videos for tasks like text-to-motion or motion prediction. Given $\mathcal{X}_Q$, the VLLM generates $\mathcal{X}_A = \{y_1, y_2, ..., y_n\}$. The dataset follows this unified format. During training, we optimizes the next-token prediction objective via negative log-likelihood:

$$\mathcal{L}(\Theta) = -\sum_{j=1}^{L} \log P_\Theta(y_j \mid \mathcal{X}_Q, \hat{y}_{1:j-1}). \qquad (1)$$

Our VLMM training involves three stages: (1) motion-text alignment, aligning motion tokens with the LLM; (2) vision-text-motion alignment, integrating three modalities into a unified framework; and (3) motion instruction tuning, enhancing responsiveness to diverse instructions.

3.2. Controllable Motion Generation

Prior research has largely overlooked the controllable potential of VLMMs, limiting their practical application. This paper addresses this gap from two perspectives.

3.2.1. Multi-Task Motion Pretraining

We define "controllability" through five key aspects and develop this capability through careful data curation and the design of multiple instructional tasks.

Random Instruction Control. Most VLMMs struggle to handle arbitrary user commands effectively, reducing their usability. While some studies [47] have explored motion instruction tuning, we further enhance command responsiveness by creating a comprehensive instruction template set (e.g., "Show me how to perform <CAPTION>.", where <CAPTION> is the motion caption), and introducing the Instruct-to-Motion (I2M) task.

Random Pose Initialization Control. A VLMM should generate motion from any initial pose to mimic human adaptability, not just fixed ones like the T-pose. However, current VLMMs struggle with this due to data scarcity. To address this, we randomly slice prior, in-between, or post segments of motion sequences and task the VLMM with predicting the remaining parts, termed the Motion Prediction and In-between (MPI) task. This task requires a massive dataset, motivating us to scale the dataset to a million-level, enabling the VLMM to learn from varied initial poses.

Long-Term Motion Control. Humans perform activities seamlessly in succession, and a practical VLMM should be able to generate continuous motions rather than isolated ones. To achieve this, we incorporate the concatenated long-form motion sequences within HuMo100M and introduce the Instruct-to-LongMotion (I2LM) task.

Unseen Motion Control. Existing datasets lack the scale needed to ensure robustness in generating unseen motions. Considering this, we expand motion data through web-video collection and multi-task design. Leveraging HuMo100M's unprecedented scale, with millions of motion instances, VLMMs can now generate high-quality motions even for actions not encountered during training. We introduce the Instruct-to-Unseen (I2U) task to benchmark our model's generalization.

Random Part Control. VLMMs should also control specific body parts (e.g., "kicking with the left leg"). Previous efforts often fall short due to insufficient part-level data, even with part-aware motion encoding [3]. Using our part-level annotations, we propose the Instruct-to-PartMotion (I2PM) task, challenging the model to generate motion for specific body parts rather than the entire body.

3.2.2. Part-aware Residual Quantization (PRQ).

Given a motion sequence $m_{1:T} \in \mathbb{R}^{T \times D}$, our PRQ first splits each motion feature m_i into part features $m_{i,j} \in \mathbb{R}^d$, where $j \in [1, p]$, d is the part feature dimension, and p=5 represents the number of body parts — {left arm, right arm, left leg, right leg, torso}. Note that the elements in different $m_{i,j}$ may overlap with shared joints. PRQ then encodes part features into a latent vector sequence $\tilde{b}_{1:n;1:p}$ using the same encoder with a downsampling ratio of n/T. Each latent vector $\tilde{b}_{i,j}$, where $i \in [1, n]$, is quantized by finding its nearest code entry in a shared motion codebook $\mathbb{C}$, producing the code sequence $b_{1:n;1:p}$. Similar to residual quantization, PRQ represents a latent sequence $\tilde{b}_{1:n}$ as $K+1$ ordered code sequences across $K+1$ quantization layers, which can be expressed as PRQ($\tilde{b}_{1:n;1:p}$) = $[b^k_{1:n;1:p}]^K_{k=0}$, where b^k denotes the code sequence at layer k. To reconstruct the motion $\tilde{m}$, the PRQ's decoder maps $\tilde{b}_{1:n;1:p}$ back to part-level motion space $\tilde{m}_{1:n;1:p}$, then aggregates each $\tilde{m}_{i,1:p}$ to restore the motion feature $\tilde{m}_i$ by selecting corresponding elements from each part feature. During tokenization, starting from the initial residual $r^0 = \tilde{b}$, PRQ iteratively computes b^k as the approximation of residual r^k, updating the residual r^{k+1} as $b^k = \mathcal{Q}(r^k), r^{k+1} = r^k - b^k$. This residual processing is applied independently for each part. The final latent sequence approximation b_j is the sum of all quantized sequences: $\sum_{i=0}^{K} b_j^k$. Similar to RQ, PRQ is also trained using motion reconstruction and latent embedding objectives at each quantization layer, with $sg[\cdot]$ as the stop-gradient operation and β as the embedding weight:

$$\mathcal{L} = \sum^{p} ||m - \tilde{m}||_1 + \sum_{j=0}^{p} ||m_j - \tilde{m}_j||_1 + \beta \sum_{k=1}^{K} \sum_{j=1}^{p} ||r_j^k - sg[b_j^i]||_2^2. \tag{2}$$

Rather than using a single code to represent the whole body, PRQ adopts part-specific codes, enabling independent control of individual body parts. In addition, PRQ expands the codebook capacity without increasing its size, outperforming methods like LFQ or FSQ. Assuming part j references u_j codes in the codebook, the total number of distinct motions representable is $\prod_{j=1}^{p} u_j$. Unlike Chen et al. [3], which focuses only on upper and lower body parts, PRQ includes three key innovations: (1) finer control over additional body parts, (2) rich part-level textual descriptions, and (3) a shared motion codebook enabling joint representations across limbs to reducing joint errors. We provide more details (e.g., part feature definition) in Appendix 2.

3.3. Design Choices for Real-time Generation

LLM Backbone. To balance performance and efficiency, we experiment with multiple LLM backbones. We find that smaller models (e.g., GPT-2) fail to grasp human intent effectively, while models with the size larger than 13B suffer from slow inference speeds, hindering real-time motion generation. After evaluation, we choose the 7B-parameter LLaMA2 as MotionCtrl's backbone.

Motion Feature. Most works use HM3D263-Format [10] to represent motions. However, HM3D263-Format loses original rotation information and requires slow inverse kinematics for pose recovery, which increases the generation latency. Instead, we adopt HuMo263 as our motion feature in this paper (see Appendix 1), which preserves the original rotation information and directly parameterizes the human pose. HuMo263 enables accurate and efficient recovery of the original information and human motion.

Visual Resolution and Duration. Visual resolution significantly affects time efficiency. We compare various resolutions and find higher ones offer minimal gains but significantly increase computation. Experimental results show that higher resolutions do not yield significant improvements but instead substantially increase computational burden. For real-time performance, MotionCtrl uses 224×224 resolution and accepts up to 64 frames as input.

Tokenization Level and Order. The stacked quantization layers (e.g., RQ) improve motion accuracy but linearly raise computational costs. Our experiments show 4 layers in PRQ strike the best balance. Unlike RQ which generates motion layer-by-layer, delaying completion until the last token, PRQ uses frame-by-frame decoding, enabling streaming motion output with improved responsiveness.

4. The HuMo100M Dataset

Scaling up data for training large models is a cornerstone in computer vision, but this strategy is less viable for motion generation due to limited high-quality data [23]. To tackle this, we introduce HuMo100M, the largest multimodal human motion dataset to date. Below, we outline its construction pipeline as illustrated in Figure 3, and highlight three key insights (★) compared to prior counterparts [23, 38, 41]. For full details, see Appendix 3.

Overview of Dataset Pipeline. We begin by collecting hundreds of millions of web videos. To ensure relevance to human activities, we apply a two-stage filtering process. First, keyword-based filtering removes videos lacking human-related text descriptors. Second, we employ YOLO [30] to verify human presence through video tracking. We then use WHAM [31] to extract SMPL parameters from the collected videos, regressing 3D human motion in world coordinates, and refine motion quality with the RL-based policy PHC [25], following Wang et al. [38].

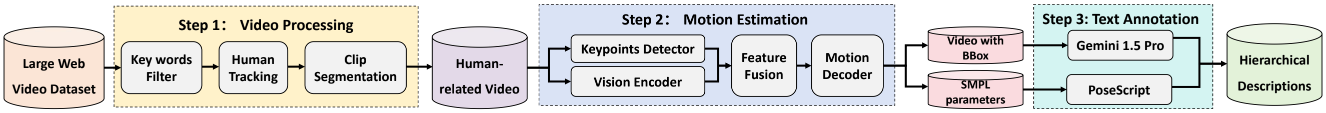


Figure 3. **Illustration of data pipeline.** We introduce HuMo100M, the largest multimodal dataset to date, featuring over 5 million motion sequences, paired visual clips, threefold more hierarchical and part-level textual descriptions, and 100 million multi-task instruction instances. Full pipeline details are in Appendix 3.

★ **Part-Level Text Descriptions.** Existing datasets lack fine-grained body part labels, limiting their usage for part-level control. To address this issue, we enrich each motion with limb-specific descriptions in addition to whole-body descriptions [10, 18]. These descriptions are generated using Gemini-1.5-pro [32] with tailored prompts and PoseScript [7]. We also include rule-based descriptions using *posecodes* to extract semantic pose details (e.g., "the left hand is below the right hand"), capturing the relative positions between different joints. This enables part-level control and allows training the VLMM on high-confidence keypoints while ignoring occluded or low-confidence parts, significantly improving motion data quality from web videos.

★ **Long-term Motion Sequence.** Current datasets mainly contain individual short-term motion sequences. While LLMs can generate long-range textual tokens, the lack of continuous motion data poses a challenge. To tackle this, we propose a motion concatenation method (see Appendix 3) that merges individual sequences into longer, temporally coherent motion sequences.

★ **Text-Aligned Visual Clips.** While some unified motion models [3, 17, 47] have incorporated visual cues, the full potential of vision in motion generation remains untapped. In this work, we argue vision adds limited value to high-quality motions (e.g., MoCap datasets like HumanML3D) but is more beneficial for training on low-quality motions collected from web videos. This is because, even with unreliable motion data, VLMMs can learn weak alignments from text-aligned visual clips, leveraging in-context information to enhance motion understanding. Furthermore, as a unified model, integrating visual clips allows VLMMs to perform motion estimation tasks, mimicking human actions and broadening real-world applications.

5. Experiments

5.1. Experimental Setup

Datasets. For the text-to-motion task, we use three datasets: HumanML3D [10] which includes 14,616 motion sequences from AMASS [26] with corresponding 44,970 text descriptions. KIT-ML [29] which offers a smaller benchmark with 3,911 motion sequences and 6,278 text descriptions. Both of these datasets are split into training (80%), validation (5%), and test sets (15%). In addition to these two datasets, we introduce HuMo-T2M by collecting 200K samples from HuMo100M to formulate a new testbed with larger scale and visual modality. For the datasets of remaining benchmarks, see Appendix 1.

Evaluation Metrics. MotionCtrl is evaluated across multiple motion-related tasks. For motion generation, we use the following metrics to assess the quality of generated motion and its alignment with texts: (1) Frechet Inception Distance (FID) which evaluates motion quality by comparing generated and real motion distributions between high-level features; (2) Motion-retrieval Precision (R-Precision) which measures text-motion alignment via top-1, top-2, and top-3 retrieval accuracy; (3) Multimodal Distance (MMDist) which quantifies the distance between matched text-motion pairs. For motion reconstruction and prediction, we employ Mean Per Joint Position Error (MPJPE) and FID, with MPJPE calculating the average joint position error (in millimeters) between predicted and ground-truth poses.

Implementation Details. Our part-aware residual VQ-VAE (PRQ) uses residual blocks for the motion encoder and decoder, with a temporal downsampling rate of 4. The codebook of PRQ has 1024 entries and 512-dimensional embeddings, employing 4 quantization layers. The discrete codes are added as vocabulary to the LLM. For real-time efficiency, we use LLaMA2-7b [34] as the LLM backbone. As a comparison, we also implement a VQ motion tokenizer with the same codebook size. The tokenizer is trained with a batch size of 256 and a learning rate of 1e-4 for 300K iterations. The training of VLLMs involves three stages: (1) motion-text alignment with full parameter tuning on 16×A800 GPUs with a batch size of 2048 for 50 epochs, using a learning rate of 2e-5; (2) vision-text-motion alignment with a batch size of 128 for 5 epochs. (3) motion instruction tuning with a batch size of 128 for one epoch.

5.2. Multi-Task Benchmarking

We compare MotionCtrl against prior works across multiple benchmarks. For each benchmark, all methods use the same motion data, except MotionCtrl*, which is trained on the full HuMo100M dataset.

Text-to-Motion (T2M). This task is a key benchmark for motion understanding. Table 1 shows results with all models trained on HumanML3D. Using the same vector quantization (VQ) approach, MotionCtrl outperforms existing works [47, 48] in terms of both accuracy and fidelity. By replacing standard VQ with our proposed PRQ, MotionCtrl

	LLM	FID ↓	R@1 ↑	R@3 ↑	MMDist ↓
Real	-	0.002	0.511	0.797	2.974
MLD [5]	-	0.473	0.481	0.772	3.196
MotionDiffuse [46]	-	0.630	0.491	0.782	3.113
T2M-GPT [45]	GPT-2	0.141	0.492	0.775	3.121
MotionGPT^1 [13]	T5	0.162	0.409	0.667	3.992
MotionGPT^2 [48]	LLaMA-13B	0.542	0.411	0.696	3.584
MotionLLM [40]	Gemma-2b	0.491	0.482	0.770	3.138
AvatarGPT [50]	LLaMA-13B	0.567	0.389	0.623	-
MotionGPT-v2 [37]	LLaMA3-8B	0.191	0.496	0.782	3.080
LMM [47]	LLaMA3-8B	0.191	0.496	0.782	3.080
MotionCtrl-VQ$_1$	LLaMA2-7B	**0.141**	**0.528**	**0.815**	**2.953**
ScaMo-FSQ [23]	3B	0.101	0.512	0.796	2.990
MoMask-RQ$_6$ [11]	760M	**0.045**	0.521	0.807	2.958
MotionCtrl-PRQ$_4$	LLaMA2-7B	0.056	**0.535**	**0.821**	**2.865**

Table 1. Comparison with previous motion methods on HumanML3D, where the superscript ^{1} and ^{2} denote different works with the same model name, and the subscript n of $\mathcal{Q}_n$ denotes the number of quantization layers for the quantizer $\mathcal{Q}$.

	LLM backbone	FID ↓	R@1 ↑	R@3 ↑	MMDist ↓
T2M-GPT [45]	GPT-2	0.682	0.154	0.275	4.251
MotionGPT^1 [13]	T5	0.382	0.268	0.352	3.621
MotionGPT^2 [48]	LLaMA-13B	0.314	0.336	0.438	3.437
MoMask-RQ$_6$ [11]	760M	0.324	0.325	0.382	3.441
MotionCtrl-PRQ$_4$	LLaMA2-7B	**0.148**	**0.428**	**0.625**	**3.259**

Table 2. Comparison with previous motion generation methods on the I2M task using the HuMo-I2M testbed. Note that this comparison is limited to works with publicly available training code.

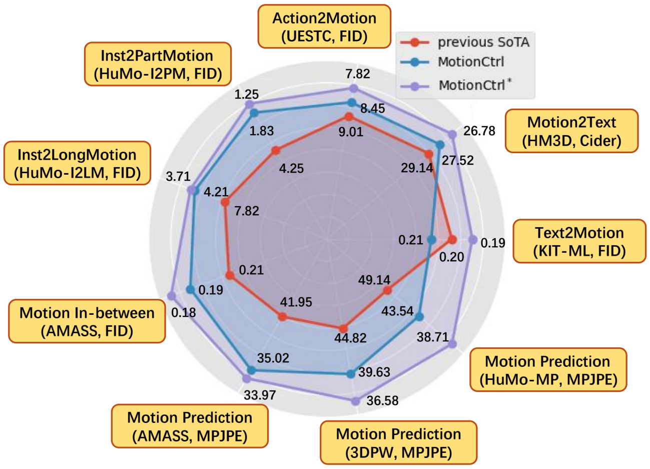


Figure 4. Comparison with previous SoTA across nine different benchmarks. Here, MotionCtrl* denotes the model trained on the full HuMo100M dataset. For newly proposed benchmarks, such as I2PM, we use MotionGPT [37] as the baseline for comparison.

Training Data	FID ↓	R@1 ↑	R@3 ↑	MMDist ↓
HumanML3D [10]	65.04	0.068	0.148	9.72
MotionX [18]	43.28	0.092	0.162	8.65
HuMo100M	**8.65**	**0.136**	**0.245**	**7.31**

Table 3. Comparison of the Instruct-to-Unseen (I2U) task on the HuMo-unseen testbed across different training datasets.

achieves further improvement, with the FID score reduced from 0.141 to 0.056. We also compare MotionCtrl with two recently proposed models based on advanced VQ variants: SCaMo with FSQ [23] and MoMask with RQ [11]. As can be seen, our model consistently outperforms these methods, despite their reliance on larger codebooks (64K entries) or deeper quantization layers, which increase computational costs. Results on KIT-ML are shown in Figure 4.

Instruct-to-Motion (I2M). Unlike T2M, the I2M task requires generating motion from arbitrary human commands, testing the model's responsiveness to handle real-world instructions. As shown in Table 2, MoMask performs significantly worse on I2M than T2M, underscoring the need for an LLM to interpret human intent accurately. Compared to other LLM-based methods like MotionGPT [37], our model achieves better results. We attribute the improvement to two important factors: the more effective motion instruction tuning and the proposal of PRQ.

Instruct-to-Unseen (I2U). To build this benchmark, we collect 200K novel motions, termed HuMo-unseen, not included in any training data. As shown in Table 3, performance on HuMo-unseen improves with data scaling from HumanML3D to HuMo100M, highlighting the importance of data scaling for robust unseen motion generation.

Instruct-to-PartMotion (I2PM). Part-level control, a key aspect of controllability, has been largely ignored in prior work. While some studies have explored part-aware motion quantization, they lack fine-grained labels and benchmarks to validate effectiveness. To address this, we use Gemini-1.5 Pro to collect motion sequences with specific part-level commands (e.g., "raise your left arm") and build the HuMo-I2PM benchmark. Results on this dataset are presented in Figure 4. MotionCtrl outperforms MotionGPT by a significant margin, primarily due to our part-aware motion encoding. Further analysis is provided in Section 5.3.

Instruct-to-LongMotion (I2LM). This task evaluates the model's ability to generate long-term, continuous motions. We create this benchmark by concatenating individual motions (e.g., "Salute with your left hand and perform ballet"). Similarly, MotionCtrl shows significant improvement.

Following Zhang et al. [47], we also carry out comparisons on additional benchmarks, including **motion prediction** on AMASS [26], 3DPW [36] and HuMo-MP, **motion in-between** on AMASS, **action-to-motion** on UESTC [12] and **motion-to-text** on HumanML3D (HM3D) [10]. The comparison results are illustrated in Figure 4.

5.3. Analysis of Motion Quantization

In Table 4, we compare our part-aware residual quantization (PRQ) with existing motion tokenizers. First, PRQ outperforms lookup-free methods like 2D-LFQ and FSQ using a codebook only 1.5% their size. The advantage grows on the large-scale HuMo100M testbed, highlighting PRQ's generalization. We attribute this to the part decomposition strategy, which boosts codebook capacity without increasing size. PRQ also surpasses RQ-VAE, even with fewer quantization layers, due to part-level encoding. As we have introduced, partial joints are shared across different body parts, designed to increase part-to-part connection so as to reduce the joint error. To validate the effectiveness of such strategy, we conduct an ablation experiment by comparing PRQ and its variant without shared joints (PRQ w/o SHA). PRQ consistently outperforms this variant.

		HumanML3D		Motion-X		HuMo100M	
Tokenizer	Code	FID	MPJPE	FID	MPJPE	FID	MPJPE
VQ-VAE$_1$	1024	0.183	47.54	0.077	38.32	5.324	123.61
H^{2}VQ [42]	512	-	-	-	62.34	-	-
RQ-VAE$_6$ [15]	1024	0.032	23.58	0.035	21.64	3.928	68.17
RQ-VAE$_8$	1024	0.009	20.42	0.012	18.11	3.526	64.56
2D-LFQ$_1$ [38]	16384	0.092	45.60	0.295	54.10	-	-
FSQ$_1$	65536	0.051	35.04	0.108	29.82	4.326	77.15
PRQ$_4$ w/o SHA	1024	0.042	19.87	0.058	23.78	3.129	48.96
PRQ$_4$	1024	0.007	14.06	0.013	17.25	2.317	38.06
PRQ$_6$	1024	**0.004**	**13.56**	**0.007**	**17.18**	**2.195**	**36.47**

Table 4. Comparison with previous motion tokenizers. The subscript of tokenizer name denotes quantization layer number. Here "w/o SHA" denotes different part features contain no shared joints.

5.4. Analysis of Time Efficiency

MotionCtrl, a 7B-parameter model, requires more memory at inference than smaller models [47] (under 1B parameters). We thus evaluate its inference speed on various GPUs, as shown in Figure 5. MotionCtrl achieves the highest throughput with 4bit quantization, reaching at least 20FPS on all GPUs and up to 28.9FPS on the H100. Specifically, with PRQ's temporal downsampling rate of 4, 5 body parts, and 4 quantization layers, MotionCtrl is required to generate at least 100 tokens per second to maintain 20FPS. Speed can be further improved by reducing quantization layers or limiting part-level control. Notably, frame-by-frame decoding is essential, as traditional layer-by-layer strategy used in RQ can delay motion generation by dozens of seconds.

5.5. Further Discussion

How does the VLMM benefit from part-level motion? We evaluate the impact of part-level motions on the HuMo-I2PM benchmark, with results illustrated in Table 5. First, removing shared joints raises the FID score from 1.831 to 2.471, as predicting isolated 3D parts without joint dependencies is challenging. In fact, the joints of our humans are highly structured, with each joint's position often dependent on others. Thus, sharing joints for different parts can strengthen their connections and therefore improve the performance. Second, comparing PRQ and RQ with part-level descriptions, MotionCtrl-RQ$_6$ underperforms despite deeper quantization layers and part labels. Further RQ experiments show minimal gains from part labels, underscoring PRQ's effectiveness to learn from such fine-grained descriptions. In addition, MotionCtrl-PRQ$_4$ without part labels performs worse than RQ, indicating that the PRQ's structure and part labels are complementary. At last, we increase the quantization layer of PRQ to validate the impact of layer numbers. While it brings improvement on the reconstruction task, the deeper quantization layers damage the generation performance. We attribute this to the added decoding complexity, especially with MotionCtrl's frame-by-frame decoding strategy, which contrasts with generating the base code layer first like Guo et al. [11].

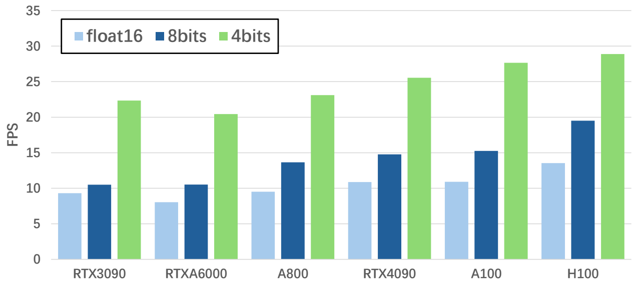
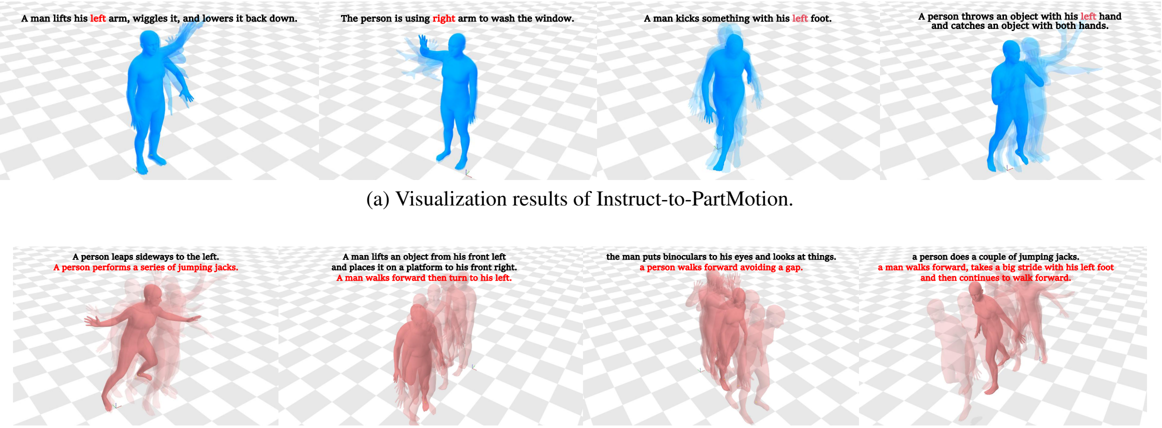


Figure 5. MotionCtrl inference speed for various GPUs. We speed up the generation by using the modern LLM inference framework llama.cpp [21]. Our model achieves real-time inference speed based on the 7B-parameter LLaMA backbone.

	PT?	FID	R@1	R@3	MMDist
MotionCtrl-RQ$_6$	No	4.025	0.208	0.395	7.01
MotionCtrl-PRQ$_4$	No	4.281	0.182	0.367	7.88
MotionCtrl-RQ$_6$	Yes	3.752	0.215	0.408	7.21
MotionCtrl-PRQ$_4$ w/o SHA	Yes	2.471	0.325	0.561	5.32
MotionCtrl-PRQ$_4$	Yes	1.831	0.384	0.685	4.12
MotionCtrl-PRQ$_6$	Yes	2.357	0.351	0.662	4.37

Table 5. Impact of part-level motions under different setups on the HuMo-I2PM testbed. Here, PT? denotes whether to use the part-level descriptions during training.

Does the visual modality benefit motion pre-training? Yes, the results in Table 6 show that MotionCtrl with vision-text-motion alignment consistently outperforms models without it. Visual cues provide weak alignment between visual and textual contexts, offering valuable supervision for motion understanding, especially when motion data is unreliable. As a unified model, VLMM can also perform motion estimation to mimic human actions using visual inputs, ex-


A man lifts his left arm, wiggles it, and lowers it back down.
The person is using right arm to wash the window.
A man kicks something with his left foot.
A person throws an object with his left hand and catches an object with both hands.
A person leaps sideways to the left.
A person performs a series of jumping jacks.
A man lifts an object from his front left and places it on a platform to his front right.
A man walks forward then turn to his left.
the man puts binoculars to his eyes and looks at things.
a person walks forward avoiding a gap.
a person does a couple of jumping jacks.
a man walks forward, takes a big stride with his left foot and then continues to walk forward.

(a) Visualization results of Instruct-to-PartMotion.

(b) Visualization results of Instruct-to-LongMotion.

Figure 6. Qualitative examples generated by MotionCtrl for Instruct-to-PartMotion (I2PM) and Instruct-to-LongMotion (I2LM). The results demonstrate MotionCtrl's ability to generate motion sequences that accurately align with both part-level and long-term instructions.

panding its applications. We leave further exploration of this capability for future work.

	FID ↓	R@1 ↑	R@3 ↑	MMDist ↓
MotionCtrl w/o 2rd vis	7.053	0.198	0.418	10.13
MotionCtrl	5.791	0.206	0.445	8.85

Table 6. Effectiveness of text-aligned visual clips on HuMo-t2m testbed, where "2rd vis" denotes the second stage training of vision-text-motion alignment.

Does the multi-task training increase the controllability of motion generation? Yes. Table 7 shows results for different motion task configurations during training, evaluated on the HuMo-T2M testing set. The initial data ratio for T2M : I2M : MPI : I2PM : I2LM is 5:5:3:2:1. Rows 1-3 indicate that removing I2PM or I2LM slightly reduces T2M performance, though the improvements on their respective benchmarks are more significant. Rows 4-5 highlight the importance of the MPI task, reducing the FID score from 6.582 to 6.052 and enabling the VLMM to generate motions from random pose initialization. Rows 5-7 confirm the necessity of I2M, with Row 5 vs. Row 6 showing that the improvements stem from the diversity from different motion tasks, rather than simply scaling data.

5.6. Visualization

We present visualization examples of part-level and long-term motion control generated by our MotionCtrl to demonstrate its controllability. Unlike previous models, MotionCtrl achieves real-time responsiveness, enabling seamless integration into animation workflows with minimal delay, as illustrated in the figures.

	multi-task configuration	FID	R@1	R@3	MMDist
1	T2M+I2M+MPI+I2PM+I2LM	5.791	0.206	0.445	8.85
2	T2M+I2M+MPI+I2PM	5.765	0.195	0.453	9.01
3	T2M+I2M+MPI+I2LM	5.952	0.192	0.453	8.91
4	T2M+I2M+MPI	6.052	0.184	0.425	9.01
5	T2M+I2M	6.582	0.154	0.386	9.77
6	2×T2M	7.058	0.128	0.342	11.26
7	T2M	7.152	0.135	0.338	10.95

Table 7. Effectiveness of multi-task motion training on HuMo-t2m testbed using different configuration.

6. Conclusion

This paper introduces a practical VLMM for real-time, controllable motion generation, achieving state-of-the-art performance across motion benchmarks. We design a data curation pipeline to create the HuMo100M dataset, the largest of its kind with 100 million instructional instances, featuring part-level descriptions, long-term motion sequences, and aligned visual clips. Using this dataset, we train MotionCtrl, enabling controllable human motion generation. To enhance part-level control, we propose a novel part-aware residual quantization method (PRQ) to serve as our motion tokenizer. Experiments validate MotionCtrl's controllability across benchmarks, and we provide key design insights for developing such a practical VLMM.

Acknowledgements

This research is supported in part by the Artificial Intelligence-National Science and Technology Major Project (2023ZD0121200) and the National Natural Science Foundation of China (6243000159, 62102416, 62450001, 62476008), and the Natural Science Foundation of Jiangsu Province under Grant BK20243051.

References

[1] Chaitanya Ahuja and Louis-Philippe Morency. Language2pose: Natural language grounded pose forecasting. In *2019 International Conference on 3D Vision (3DV)*, pages 719–728. IEEE, 2019. 2

[2] Sadegh Aliakbarian, Fatemeh Sadat Saleh, Mathieu Salzmann, Lars Petersson, and Stephen Gould. A stochastic conditioning scheme for diverse human motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5223–5232, 2020. 2

[3] Changan Chen, Juze Zhang, Shrinidhi K Lakshmikanth, Yusu Fang, Ruizhi Shao, Gordon Wetzstein, Li Fei-Fei, and Ehsan Adeli. The language of motion: Unifying verbal and non-verbal language of 3d human motion. *arXiv preprint arXiv:2412.10523*, 2024. 2, 4, 5

[4] Ling-Hao Chen, Shunlin Lu, Ailing Zeng, Hao Zhang, Benyou Wang, Ruimao Zhang, and Lei Zhang. Motionllm: Understanding human behaviors from human motions and videos. *arXiv preprint arXiv:2405.20340*, 2024. 2

[5] Xin Chen, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, and Gang Yu. Executing your commands via motion diffusion in latent space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18000–18010, 2023. 6

[6] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24185–24198, 2024. 2

[7] Ginger Delmas, Philippe Weinzaepfel, Thomas Lucas, Francesc Moreno-Noguer, and Grégory Rogez. Posescript: Linking 3d human poses and natural language. *IEEE transactions on pattern analysis and machine intelligence*, 2024. 5

[8] Katerina Fragkiadaki, Sergey Levine, Panna Felsen, and Jitendra Malik. Recurrent network models for human dynamics. In *Proceedings of the IEEE international conference on computer vision*, pages 4346–4354, 2015. 2

[9] Anand Gopalakrishnan, Ankur Mali, Dan Kifer, Lee Giles, and Alexander G Ororbia. A neural temporal model for human motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12116–12125, 2019. 2

[10] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5152–5161, 2022. 1, 2, 4, 5, 6

[11] Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. Momask: Generative masked modeling of 3d human motions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1900–1910, 2024. 1, 2, 6, 7

[12] Yanli Ji, Feixiang Xu, Yang Yang, Fumin Shen, Heng Tao Shen, and Wei-Shi Zheng. A large-scale rgb-d database for arbitrary-view human action recognition. In *Proceedings of the 26th ACM international Conference on Multimedia*, pages 1510–1518, 2018. 1, 6

[13] Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. Motiongpt: Human motion as a foreign language. *Advances in Neural Information Processing Systems*, 36:20067–20079, 2023. 1, 2, 6

[14] Biao Jiang, Xin Chen, Chi Zhang, Fukun Yin, Zhuoyuan Li, Gang Yu, and Jiayuan Fan. Motionchain: Conversational motion controllers via multimodal prompts. *arXiv preprint arXiv:2404.01700*, 2024. 2

[15] Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. Autoregressive image generation using residual quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11523–11532, 2022. 2, 7

[16] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 1

[17] Yiheng Li, Ruibing Hou, Hong Chang, Shiguang Shan, and Xilin Chen. Unipose: A unified multimodal framework for human pose comprehension, generation and editing. *arXiv preprint arXiv:2411.16781*, 2024. 5

[18] Jing Lin, Ailing Zeng, Shunlin Lu, Yuanhao Cai, Ruimao Zhang, Haoqian Wang, and Lei Zhang. Motion-x: A large-scale 3d expressive whole-body human motion dataset. *Advances in Neural Information Processing Systems*, 36, 2024. 1, 5, 6

[19] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2023. 1

[20] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 2

[21] llamacpp project members. llamacpp. *https://github.com/ggml-org/llama.cpp*, 2024. 7

[22] Shunlin Lu, Ling-Hao Chen, Ailing Zeng, Jing Lin, Ruimao Zhang, Lei Zhang, and Heung-Yeung Shum. Humantomato: Text-aligned whole-body motion generation. *arXiv preprint arXiv:2310.12978*, 2023. 2

[23] Shunlin Lu, Jingbo Wang, Zeyu Lu, Ling-Hao Chen, Wenxun Dai, Junting Dong, Zhiyang Dou, Bo Dai, and Ruimao Zhang. Scamo: Exploring the scaling law in autoregressive motion generation model. *arXiv preprint arXiv:2412.14559*, 2024. 1, 2, 4, 6

[24] Mingshuang Luo, Ruibing Hou, Hong Chang, Zimo Liu, Yaowei Wang, and Shiguang Shan. M^{3}gpt: An advanced multimodal, multitask framework for motion comprehension and generation. *arXiv preprint arXiv:2405.16273*, 2024. 1, 2

[25] Zhengyi Luo, Jinkun Cao, Kris Kitani, Weipeng Xu, et al. Perpetual humanoid control for real-time simulated avatars. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10895–10904, 2023. 4

[26] Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, and Michael J Black. Amass: Archive of motion capture as surface shapes. In *Proceedings of*

the IEEE/CVF international conference on computer vision, pages 5442–5451, 2019. 5, 6

[27] Fabian Mentzer, David Minnen, Eirikur Agustsson, and Michael Tschannen. Finite scalar quantization: Vq-vae made simple. *arXiv preprint arXiv:2309.15505*, 2023. 2

[28] Mathis Petrovich, Michael J Black, and Gül Varol. Temos: Generating diverse human motions from textual descriptions. In *European Conference on Computer Vision*, pages 480–497. Springer, 2022. 2

[29] Matthias Plappert, Christian Mandery, and Tamim Asfour. The kit motion-language dataset. *Big data*, 4(4):236–252, 2016. 5

[30] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 4

[31] Soyong Shin, Juyong Kim, Eni Halilaj, and Michael J Black. Wham: Reconstructing world-grounded humans with accurate 3d motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2070–2080, 2024. 4

[32] Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024. 5

[33] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 2

[34] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 5

[35] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 2

[36] Timo Von Marcard, Roberto Henschel, Michael J Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *Proceedings of the European conference on computer vision (ECCV)*, pages 601–617, 2018. 6

[37] Yuan Wang, Di Huang, Yaqi Zhang, Wanli Ouyang, Jile Jiao, Xuetao Feng, Yan Zhou, Pengfei Wan, Shixiang Tang, and Dan Xu. Motiongpt-2: A general-purpose motion-language model for motion generation and understanding. *arXiv preprint arXiv:2410.21747*, 2024. 2, 6

[38] Ye Wang, Sipeng Zheng, Bin Cao, Qianshan Wei, Qin Jin, and Zongqing Lu. Quo vadis, motion generation? from large language models to large motion models. *arXiv preprint arXiv:2410.03311*, 2024. 1, 2, 4, 7

[39] Zhenyi Wang, Ping Yu, Yang Zhao, Ruiyi Zhang, Yufan Zhou, Junsong Yuan, and Changyou Chen. Learning diverse stochastic human-action generators by learning smooth latent transitions. In *Proceedings of the AAAI conference on artificial intelligence*, pages 12281–12288, 2020. 2

[40] Qi Wu, Yubo Zhao, Yifan Wang, Yu-Wing Tai, and Chi-Keung Tang. Motionllm: Multimodal motion-language learning with large language models. *arXiv preprint arXiv:2405.17013*, 2024. 6

[41] Liang Xu, Shaoyang Hua, Zili Lin, Yifan Liu, Feipeng Ma, Yichao Yan, Xin Jin, Xiaokang Yang, and Wenjun Zeng. Motionbank: A large-scale video motion benchmark with disentangled rule-based annotations. *arXiv preprint arXiv:2410.13790*, 2024. 4

[42] Tackgeun You, Saehoon Kim, Chiheon Kim, Doyup Lee, and Bohyung Han. Locally hierarchical auto-regressive modeling for image generation. *Advances in Neural Information Processing Systems*, 35:16360–16372, 2022. 2, 7

[43] Lijun Yu, José Lezama, Nitesh B Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Agrim Gupta, Xiuye Gu, Alexander G Hauptmann, et al. Language model beats diffusion–tokenizer is key to visual generation. *arXiv preprint arXiv:2310.05737*, 2023. 2

[44] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023. 2

[45] Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Yong Zhang, Hongwei Zhao, Hongtao Lu, Xi Shen, and Ying Shan. Generating human motion from textual descriptions with discrete representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14730–14740, 2023. 6

[46] Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. Motiondiffuse: Text-driven human motion generation with diffusion model. *arXiv preprint arXiv:2208.15001*, 2022. 6

[47] Mingyuan Zhang, Daisheng Jin, Chenyang Gu, Fangzhou Hong, Zhongang Cai, Jingfang Huang, Chongzhi Zhang, Xinying Guo, Lei Yang, Ying He, et al. Large motion model for unified multi-modal motion generation. In *European Conference on Computer Vision*, pages 397–421. Springer, 2024. 1, 2, 3, 5, 6, 7

[48] Yaqi Zhang, Di Huang, Bin Liu, Shixiang Tang, Yan Lu, Lu Chen, Lei Bai, Qi Chu, Nenghai Yu, and Wanli Ouyang. Motiongpt: Finetuned llms are general-purpose motion generators. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 7368–7376, 2024. 2, 5, 6

[49] Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Video instruction tuning with synthetic data. *arXiv preprint arXiv:2410.02713*, 2024. 2

[50] Zixiang Zhou, Yu Wan, and Baoyuan Wang. Avatargpt: All-in-one framework for motion understanding planning generation and beyond. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1357–1366, 2024. 6