Weak Models or Insensitive Metrics? Auditing and Training Fine-Grained Motion Captioning

Anonymous authors
Paper under double-blind review

Abstract

Motion captions are evaluated by n-gram overlap, embedding similarity, or an LLM judge that scores the caption as a whole, but whether these metrics notice one wrong motion fact, such as the acting side or the direction of movement, has not been tested attribute by attribute. Without that test, a weak model cannot be distinguished from an insensitive metric. We build MotionADV, a Chinese–English challenge set of caption pairs that differ in exactly one fact, from 494 clips whose captions were corrected by professional motion artists. Under a single wrong fact, n-gram and embedding-based metrics move less than a fifth of the way to an unrelated caption, and n-gram metrics even prefer the wrong caption to a fact-preserving paraphrase. A whole-caption LLM judge cannot both detect and localize the error. We define MotionFact, a fixed rubric of motion attributes over a typed action schema, and build on it MotionJudge, a decomposed evaluation protocol that checks each attribute against a reference with its own questions. With Qwen3.5-27B, MotionJudge reaches a detection rate of at least 0.90 and a pass rate of at least 0.89 on every attribute, and an ablation that adds the rubric and then the schema-based protocol shows what each contributes. We further train Motion-VLM, a general video-language model that reads multi-view renders and motion tokens, and states MotionFact facts before its caption. Under both judges, Motion-VLM outperforms five published motion captioners on HumanML3D and demonstrates superior cross-dataset transferability.

1 Introduction

Motion-language models can now describe 3D human motion in natural language (Jiang et al., 2023; Zhu et al., 2026; Wu et al., 2025a;b; Chen et al., 2025), and their captions are evaluated by n-gram overlap (BLEU, ROUGE-L, CIDEr), text–motion retrieval (R-Precision), or an LLM judge (Zheng et al., 2023) that scores the caption as a whole. Yet a motion caption is useful only if its physical facts are correct: which arm moved, in which direction, how many times, and in what order. Current models struggle with exactly these facts: even strong vision-language models often misjudge the signed direction of a movement and remain well below human performance at distinguishing left from right (Lee et al., 2026; Jia et al., 2026; Kong et al., 2025). A low score on such facts has three possible causes: the model cannot perceive the evidence, its training target never asks it to state the fact, or the metric is insensitive to the error. Prior work has studied the first two, but unless the third is ruled out, a weak model cannot be distinguished from an insensitive metric. Figure 1 illustrates the problem. Against the reference "the person raises the **left** hand", the incorrect caption "the person raises the **right** hand" receives the same BLEU and ROUGE-L scores as the fact-preserving rewrite "the **woman** raises the **left** hand", and an even higher BERTScore. N-gram metrics register a single substituted word in each case, regardless of whether the substitution is wrong or harmless, while embedding-based metrics place *left* close to *right* because the two words occur in similar contexts.

Motion captions make this test feasible: although their wording is open-ended, the physical facts they can state fall into a small set of types, which we call *attributes*, such as the acting side, movement direction, rotation, speed, amplitude, count, and temporal order (Laban & Ullmann, 1971; Delmas et al., 2022; Yazdian et al., 2025). We formalize these facts as MotionFact, a fixed rubric of attributes defined over a typed action schema, and use it in three roles throughout the paper: to alter exactly one fact when auditing a metric, to check captions attribute by attribute, and to specify the facts as training targets.

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.

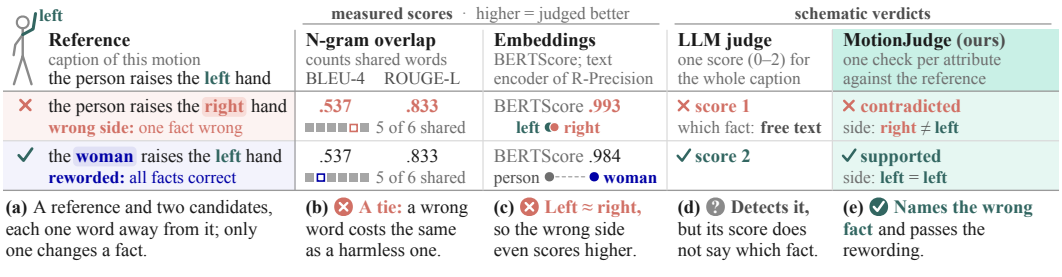


Figure 1: **Common caption metrics are insensitive to a wrong fact.** Rows are the two candidates for the reference in (a); columns (b)–(e) are evaluators. N-gram overlap ties the wrong side with a fact-preserving rewording, and embeddings place *left* beside *right* (Table 16). An LLM judge lowers its score, but only its free-text reason says which fact is wrong; asked to name the attribute, it often misses a wrong direction (Table 1). MotionJudge finds the side contradicted.

We first audit existing metrics with MotionADV, a challenge set built on 494 clips whose captions were corrected by professional motion artists (our *verified references*). It consists of Chinese and English minimal pairs whose labels are cross-verified by LLMs from multiple model families. For each attribute, we report the *detection rate*, i.e., the fraction of single-fact errors a metric flags, and the *pass rate*, i.e., the fraction of fact-preserving rewrites it leaves unflagged. Under a single-fact error, n-gram and embedding-based metrics change by less than one fifth of the change they show for an unrelated caption. A whole-caption LLM judge either cannot identify which attribute is wrong or, when asked to name it, misses errors on attributes such as direction, whose value must first be located. We therefore build MotionJudge, a decomposed evaluation protocol on top of MotionFact: it parses a caption into actions, aligns them with the verified reference, and checks each attribute with its own questions, distinguishing a contradicted fact from an omitted one. Adding first the rubric and then the typed schema, together with the decomposed protocol built on it, shows that the rubric enables the judge to name the erroneous attribute, while the schema-based protocol files each value under the correct attribute. With both, MotionJudge achieves a detection rate of at least 0.90 and a pass rate of at least 0.89 on all eleven attributes in both languages.

The remaining two causes lie in the model itself. We address them with Motion-VLM, a general video-language model that takes three rendered views of the motion together with the motion sequence as input. Before writing its caption, it predicts the azimuth of each camera and enumerates the MotionFact facts of each action, whose targets are computed from the motion data. On HumanML3D, Motion-VLM outperforms five published motion-specific captioners under both MotionJudge and a whole-caption LLM judge, even though R-Precision@1 ranks one of them first. On Motion-X++, it substantially outperforms zero-shot video-language models with up to 32B parameters. Given the right input and training target, a general video-language model thus states motion facts more accurately than models built specifically for motion, a conclusion that insensitive metrics would have obscured.

Our contributions are as follows:

- We introduce MotionFact, a fixed rubric of motion attributes defined over a typed action schema, and MotionADV, a per-attribute challenge set of minimal pairs for auditing motion-caption metrics.
- We propose MotionJudge, a decomposed evaluation protocol built on MotionFact that detects single-fact errors on every attribute, and ablate its rubric and its schema-based protocol.
- We present Motion-VLM, which states MotionFact facts before captioning and outperforms five published captioners on HumanML3D under both judges; ablations show that both multi-view input and structured targets contribute to its gains.

2 Related Work

LLM-based motion understanding. Building on motion–text translation (Guo et al., 2022b), language models for 3D human motion now caption motion, answer questions about it, or generate it (Jiang et al., 2023; Luo et al., 2024; Zhu et al., 2026; Wu et al., 2025a; Xu et al., 2026; Wu et al., 2025b; Chen et al., 2025). Most are trained on HumanML3D (Guo et al., 2022a), while Motion-X++ (Zhang et al., 2025b) offers a larger dataset of whole-body motion. Their captions are evaluated by

n-gram or embedding similarity to a reference caption and by text–motion retrieval, neither of which shows whether a caption states the correct side, direction, or count.

LLM judges and decomposed evaluation. LLMs are widely used as judges (Gu et al., 2026), either prompted (Zheng et al., 2023; Liu et al., 2023) or fine-tuned to score against a rubric (Kim et al., 2024). Splitting a judgment into checklist questions or atomic facts is established for text (Min et al., 2023; Lee et al., 2025; Cook et al., 2024) and for images and video (Hu et al., 2023; Cho et al., 2024; Dong et al., 2024; Chai et al., 2025; Wang et al., 2024). Hallucination metrics measure objects that a caption invents (Rohrbach et al., 2018; Petryk et al., 2024), and MotionHalluc tests directional, attributional, and temporal hallucinations in comparisons of two motion videos (Guo et al., 2026). MotionJudge decomposes motion captions into facts with the fixed rubric and typed action schema of MotionFact, distinguishing a contradicted fact from an omitted one.

Auditing metrics and judges. Challenge sets of minimal pairs are a standard way to test NLP models (Ribeiro et al., 2020), text-generation metrics (Sai et al., 2021; Karpinska et al., 2022; Amrhein et al., 2022), and vision-language models (Shekhar et al., 2017; Hsieh et al., 2023; Yuksekgonul et al., 2022). LLM judges have been audited for position and verbosity biases (Zheng et al., 2023) and on response pairs with objective correctness labels (Tan et al., 2025). For motion, the text-to-motion metrics in use correlate poorly with human ratings of individual samples (Voas et al., 2023), and motion-language models often fail to distinguish a description from one with reordered events (Fujiwara et al., 2024). We extend this test from temporal order to all eleven attributes of MotionFact, report detection and pass rates for each attribute, and validate the labels of the edits (Appendix G).

3 Auditing Motion-Caption Metrics

This section tests the third cause: whether the metric is insensitive to a wrong fact. Following challenge sets for machine translation (Amrhein et al., 2022; Karpinska et al., 2022), we build caption pairs that differ in exactly one motion fact and examine whether a metric detects the difference.

3.1 Detection and pass rates

A metric $h(c, r)$ compares a candidate c with a reference r and returns *error* or *pass*. For an attribute, T_e changes, deletes, or inserts one of its facts, and T_b keeps it (a paraphrase, an identical copy, or an edit of another attribute). Over the references x in C_e and $\mathcal{B}_b$ to which these edits apply, the detection and pass rates are

$$\mathrm{Det}_e(h) = \frac{1}{|C_e|} \sum_{x \in C_e} \mathbb{1}[h(T_e(x), x) = \text{error}], \qquad \mathrm{Pass}_b(h) = \frac{1}{|\mathcal{B}_b|} \sum_{x \in \mathcal{B}_b} \mathbb{1}[h(T_b(x), x) = \text{pass}]. \tag{1}$$

with indicator $\mathbb{1}[\cdot]$. In CheckList terms (Ribeiro et al., 2020; Sai et al., 2021), the detection rate is a directional test and the pass rate an invariance test. A metric that flags everything detects every error but passes nothing, so we report both. Both rates measure sensitivity to controlled edits, not accuracy on natural captions.

3.2 The MotionADV challenge set

Verified references. We start from an industrial dataset of 494 motion clips whose model-drafted captions were corrected by professional motion artists, clip by clip. The corrections were substantial: in most clips the artists rewrote the action, the direction, or the acting side. We parse each verified reference into MotionFact actions (Section 4.1), filling a slot only where the reference states the fact. The draft–correction pairs also provide real errors, which we use in Section 4.3.

Single-fact edits. Qwen3.5-27B rewrites the action phrase that states a value so that the value changes or disappears, leaving the rest of the caption unchanged. Each error sample thus differs from its source in one value. Temporal-order edits swap two adjacent actions by rule. MotionADV covers eleven attributes: side, direction, motion type, bilateral mode, amplitude, speed, oscillation axis, rotation, articulation, count, and temporal order. We edit a single action so that the change in caption length does not reveal the edit.

Label validation. The label of a controlled edit follows from its construction only if the edit changes the described motion. Not every edit does: an independent verifier judged more than half of a set of rule-generated action swaps to be semantically unchanged. Three validator LLMs from two families (Qwen3.5-27B, Qwen3-VL-32B, and gemma-4-31B) therefore re-derive the original value without being shown our label, state the edited value, and check that no other attribute changed. A label passes on a unanimous vote or on a majority that spans both families. When the two Qwen models outvote gemma, GPT-5.5, from a third family, arbitrates. Qwen3.5-27B is also the default base of the judges we test later, so a judge could share its blind spots with the labels. Yet on the labels that the other two validators pass without it, MotionJudge still detects at least 0.94 of all errors in both languages (Appendix F.5). Fact-preserving samples come from the same clips: paraphrases and identical copies of the reference (Chinese only), and the untouched attributes of each error sample. The English set is a paired translation of the Chinese set by Qwen3.5-27B, checked by the same three validators. The result is 1,658 Chinese error samples from 435 clips and 1,334 English error samples from 414 clips; per-attribute counts and examples are in Appendix D.

3.3 Audit results

We evaluate n-gram metrics (BLEU, ROUGE-L, METEOR, CIDEr), the text encoder of TMR, a text–motion retrieval model (Petrovich et al., 2023), and a whole-caption LLM judge on the same pairs. For the similarity metrics, which output a continuous score rather than an error flag, we report a *blindness ratio*: the median score change caused by a single-fact error, divided by the median change caused by an unrelated caption.

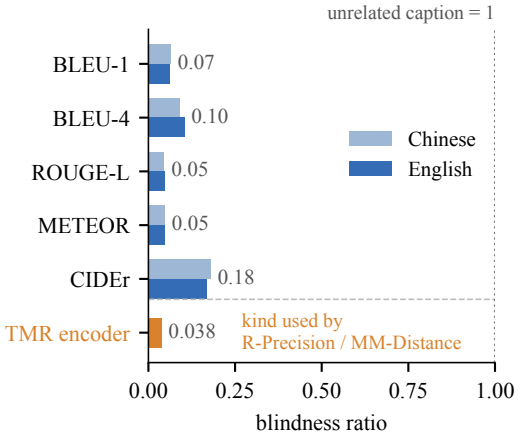


Figure 2: **Blindness ratio** (lower means less sensitive). R-Precision and MM-Distance read captions only through a text encoder; we test one such encoder, that of TMR (English only).

Similarity metrics are insensitive. Figure 2 shows that one wrong fact moves every similarity metric by a small fraction of what an unrelated caption does: roughly a twentieth to a tenth for BLEU, ROUGE-L, and METEOR, and under a fifth for CIDEr. On the full MotionADV, each of these metrics even scores the single-fact error above a fact-preserving paraphrase in at least 1,258 of 1,261 Chinese pairs (Appendix D.5). The TMR encoder, the kind of encoder through which R-Precision and MM-Distance read a caption, is the least sensitive: it places a caption with the wrong side almost on top of the correct one, while separating unrelated captions clearly. This extends to other attributes the insensitivity to temporal order found by Fujiwara et al. (2024): a higher R-Precision says little about whether the facts are correct. Embedding-based metrics fare no better: on a small probe of minimal pairs, BERTScore usually prefers the caption with the wrong fact to a fact-preserving rewrite.

A whole-caption LLM judge cannot both detect and localize. Given both the reference and the candidate, a whole-caption LLM judge does react to single-fact errors (Table 1, *One verdict*). However, a single verdict does not say which attribute is wrong, and when asked to name the attribute (+*Rubric*), the judge misses errors whose value must first be located in the sentence: it detects only 0.56 of the direction errors and 0.33 of the oscillation-axis errors in Chinese. Section 4 builds a judge that avoids both failures by giving each attribute a typed slot and its own questions.

4 MotionFact and MotionJudge

The audit calls for a judge that locates the fact each claim refers to, identifies which attribute is wrong, and leaves fact-preserving rewrites unflagged. We first formalize motion facts as MotionFact and then present MotionJudge, a decomposed evaluation protocol built on top of it. MotionFact later also structures the training target of Motion-VLM.

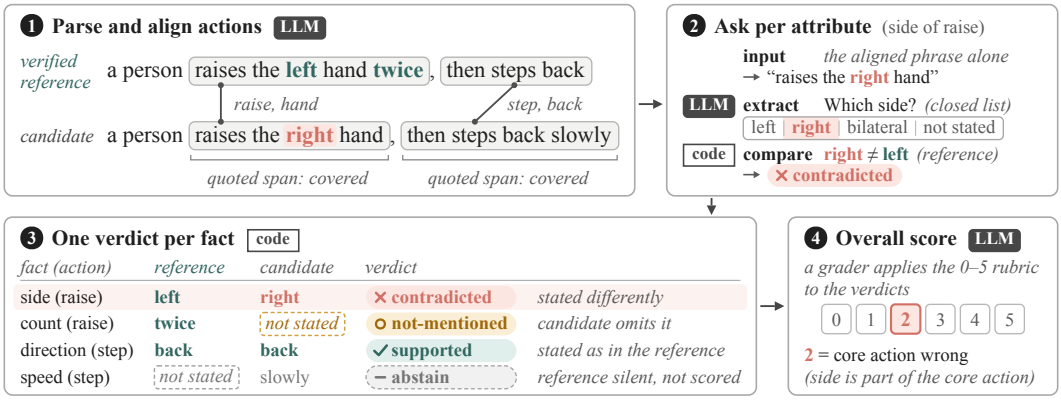


Figure 3: **MotionJudge checks a caption fact by fact and names the wrong one.** In this example, an LLM parses, aligns, extracts, and grades; code compares each value with the reference.

4.1 MotionFact: a fixed rubric over typed actions

MotionFact consists of two components. The *rubric* is a fixed set of named criteria, as in rubric-based LLM evaluation (Kim et al., 2024): the attributes of Section 3, and a severity scale that separates a wrong core action (action, direction, side, motion type) from a wrong detail (speed, amplitude, count). Unlike checklists generated per instance (Cook et al., 2024), the criteria are the same for every clip, so verdicts can be aggregated per attribute and compared across clips and systems. The *typed action schema*, a small domain-specific language, provides a typed slot for each fact. A description is an ordered sequence of N actions $U = (u_1, \ldots, u_N)$ with $u_i = (a_i, p_i, s_i, d_i, f_i, r_i, c_i, v_i, \tau_i)$: action, body part, side, displacement direction, facing, rotation, count, speed or amplitude, and an optional temporal span. Each filled slot of an action is thus a typed atomic fact (Min et al., 2023), much as a posecode is for a static pose (Delmas et al., 2022). Every slot has a finite domain, so an ill-typed value is rejected at parsing instead of being scored as a false claim. The order of the sequence makes temporal order a value in its own right.

The slots are designed to be orthogonal: each can change without changing another. This lets MotionADV edit exactly one fact and MotionJudge ask about one attribute at a time. A new body part or form of movement is added as a new slot or value with its own questions. Except for temporal order, each of the eleven attributes is a slot of the schema or a refinement of its action slot. On system outputs, MotionJudge scores the attributes that the reference states, and the training target uses the six that can be computed from the motion data.

References and verdicts. Each fact is judged against a reference: the verified reference on the 494 artist-corrected clips, or the original caption on public datasets. A reference is assumed to be correct in what it states but need not state every fact of the clip, so four cases are kept apart. A fact stated as the reference states it is *supported*, and a fact stated differently is *contradicted*. A fact that the reference states and the candidate omits is *not-mentioned*, so a model cannot raise its score by staying silent. A fact that the reference does not state is not scored (*abstain*).

4.2 MotionJudge: a decomposed evaluation protocol

Following question-based evaluation of text-to-image generation (Hu et al., 2023; Cho et al., 2024), MotionJudge fixes which questions a language model is asked, in what order, and how the answers are combined. Figure 3 walks through one example. The protocol is built to meet four requirements. **(i) Localization**: an LLM splits the candidate into actions and aligns each reference action with one of them or with none, so that every attribute question concerns a matched pair of actions. **(ii) Attribution**: each attribute has its own question templates, which ask the LLM to extract, from a closed list, the value stated by the matched candidate phrase, shown without the rest of the caption, so that a changed value is filed under the right attribute. **(iii) Deterministic comparison**: code compares the extracted values with the reference values, and the order that the LLM reads from the candidate for each pair of actions with the order in the reference. **(iv) Faithful coverage**: a reference action counts as covered only if the LLM quotes a span of the candidate that expresses it; code checks that

Table 1: **Adding the rubric and then the schema-based protocol on MotionADV** (detection / pass rate; red: detection below 0.90 or pass below 0.89). All judges use Qwen3.5-27B. *One verdict* and + *Rubric* are whole-caption judges that flag the caption or name the wrong attributes (Appendix F.2); + *Schema* is MotionJudge. *n*: MotionJudge's error samples (Chinese/English); *rewrites and copies*: 281 paraphrases and 281 identical copies (Chinese only).

		Chinese			English		
		Whole-caption judges			Whole-caption judges		
Attribute	*n*	One verdict	+ Rubric	+ Schema (MotionJudge)	One verdict	+ Rubric	+ Schema (MotionJudge)
side	241/217	.780 / .320	.884 / .970	.992 / .960	.765 / .352	.889 / .987	.982 / .972
direction	233/196	.722 / .309	.557 / .945	.923 / .938	.720 / .343	.640 / .951	.939 / .934
motion type	121/83	.775 / .312	.761 / .977	.950 / .934	.737 / .339	.677 / .964	.928 / .934
bilateral mode	125/71	.760 / .310	.712 / .995	.936 / .937	.676 / .334	.662 / .998	.901 / .894
amplitude	234/228	.477 / .268	.881 / .996	.979 / .976	.463 / .291	.769 / .999	.974 / .986
speed	192/183	.505 / .278	.980 / .997	.974 / .977	.482 / .302	.974 / .993	.989 / .977
oscillation axis	97/41	.588 / .298	.330 / .999	.969 / .956	.535 / .326	.352 / .998	.951 / .915
rotation	57/29	.768 / .309	.853 / .992	.965 / .993	.857 / .339	.952 / .992	.966 / .997
articulation	192/155	.854 / .326	.896 / .997	.948 / .974	.871 / .360	.877 / .997	.929 / .968
count	68/49	.918 / .315	1.00 / .994	1.00 / .963	.904 / .343	1.00 / .998	.980 / .970
temporal order	107/116	–	–	.991 / 1.00	–	–	.991 / 1.00
rewrites and copies	562	1.00	1.00	.976			
attributes in red		10/10	8/10	0/11	10/10	7/10	0/11
minimum		.477 / .268	.330 / .945	.923 / .934	.463 / .291	.352 / .951	.901 / .894

the quote occurs verbatim, and a second question confirms the match. Finally, an LLM grader applies the rubric to the per-attribute verdicts and assigns an overall score from 0 to 5. All prompts are listed in Appendix I.1.

4.3 Validation of MotionJudge

Ablation of the rubric and schema. In Table 1, we give a judge built on Qwen3.5-27B first the rubric of MotionFact and then its typed schema, which comes with the decomposed protocol; the last judge is MotionJudge itself. The whole-caption judge asked for one verdict detects many errors and never flags a paraphrase, but it cannot say which attribute is wrong, so whenever it flags an error sample, it flags every untouched attribute of that sample as well. Given the rubric and asked to name the wrong attributes, it recovers the pass rate and handles attributes whose value is a single word, such as speed and count, as well as MotionJudge does. It still misses errors whose value must first be located, above all on oscillation axis and direction, where it often names a neighboring attribute: a sway changed from left–right to front–back is filed under direction. The typed schema and the decomposed protocol built on it close this gap: MotionJudge detects at least 0.90 of the errors on all eleven attributes in both languages, at a small cost in pass rate. With gemma-4-31B as the base, MotionJudge still detects more oscillation-axis and direction errors than the + *Rubric* judge (Appendix F.8).

Real caption errors. Real errors are harder than single-fact edits. We score each draft of the artists' draft–correction pairs (Section 3.2) against its correction. Of the 54 drafts whose side words the artist rewrote and the 76 whose direction words were rewritten, MotionJudge contradicts a side fact in 30% and a direction fact in 20%, and marks most of the other 99 not-mentioned. In 65 of these, the artist replaced, added, or deleted the whole action, so not-mentioned is correct. Of the rest, 22 are misses, 13 of them because alignment rejects an action that differs only in side or direction; even so, the grader gives each miss a 2, the score for a wrong core action. On unchanged attributes MotionJudge contradicts at most 15%, and it flags most omissions on every attribute except action (Appendix F.9).

Agreement between judges. MotionJudge and a whole-caption LLM judge rank the HumanML3D systems of Section 6 almost identically (Spearman 0.98), and MotionJudge also identifies which fact is wrong.

5 Motion-VLM: Multi-View Input and Structured Targets

The audit and MotionJudge address the third cause, the metric. The remaining two causes lie in the model: it may not perceive the evidence, or its training target may never ask it to state the fact. We

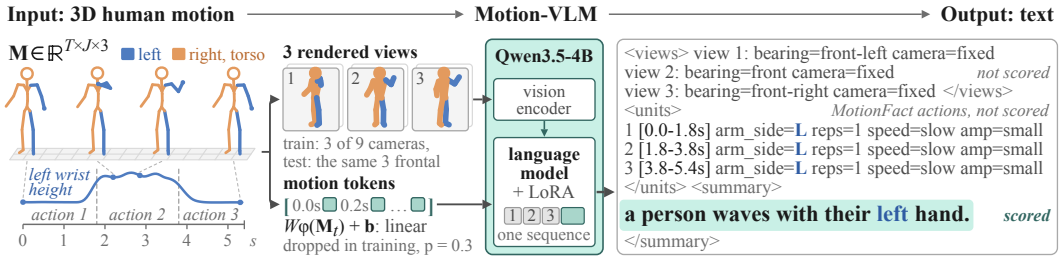


Figure 4: **Motion-VLM, a motion captioning model built on a video-language model.** It reads three rendered views and timestamped motion tokens in one sequence and writes camera azimuths and MotionFact actions before the caption, which alone is scored. Shown is its actual output on a HumanML3D test motion; the poses, wrist height, and views are drawn from the motion's joints.

therefore ask whether a general video-language model, given the right input and training target, can state motion facts. As Figure 4 shows, Motion-VLM reads rendered views of the motion together with the motion sequence and combines three choices. Multi-view renders prevent occlusion or a single viewpoint from hiding which side moved or in which direction. Camera azimuths in the target make the model state where each camera stands relative to the actor. A structured target lists the MotionFact facts of each action, computed from the motion data, before the caption. We fine-tune Qwen3.5-4B (Qwen Team, 2026) with LoRA.

5.1 Input: multi-view video and motion tokens

Each motion is rendered as an untextured video of the body; since the body is left–right symmetric, the left limbs are blue and the right limbs and torso orange. Training samples three views per example from a pool of nine cameras, eight fixed every 45° around the actor and one that tracks it; testing uses three fixed frontal cameras. The joint positions $\mathbf{M}_t$, sampled at the video frame times, enter as motion tokens $\mathbf{h}_t^M = W\,\phi(\mathbf{M}_t) + \mathbf{b}$, where ϕ returns normalized root-relative positions and W and $\mathbf{b}$ are the weight and bias of a single linear layer trained from scratch: as in Gemma 4 12B for vision and audio (Team et al., 2026), the motion input is encoder-free. Each token is preceded by the timestamp of its frame, so the model can match every pose to the frames that show it. Training drops the motion with probability 0.3, so the model also runs on video alone. Appendix B.1 gives the settings and the prompt.

5.2 Output: a structured training target

Beyond the dataset caption, we construct supervision from the motion data itself and train Motion-VLM to write

$$\mathbf{y} \;=\; \langle \texttt{views} \rangle\, \theta_{1:K}\;\; \langle \texttt{units} \rangle\, u_{1:N}\;\; \langle \texttt{summary} \rangle\, y^{\star}. \tag{2}$$

Here θ_k gives the azimuth of the camera of video k ($K = 3$) relative to the actor's initial facing and whether that camera is fixed or tracking, so that the model relates the views to one another in space. The ordered MotionFact actions $u_{1:N}$ carry six slots (arm side, displacement direction, rotation, count, speed, and amplitude) computed from the joint trajectories (Appendix B.3); they are noisy but caption-independent, and act as weak supervision for captioning. The summary $y^{\star}$ is the dataset's original caption. The model is trained on the whole target, but only the summary is scored; we conjecture that the actions act like a chain of thought (Wei et al., 2022), stating each fact before the summary.

6 Experiments

6.1 Experimental setup

Datasets. **HumanML3D** (Guo et al., 2022a) is the standard motion-captioning benchmark and the training dataset of every published baseline; its test split has 2,192 motions and their left–right mirrors, and the longest of each motion's captions serves as its reference. **Motion-X++** (Zhang et al., 2025b) provides 52-joint whole-body motion, largely estimated from video; we evaluate on 2,286 held-out clips.

Table 2: **Motion captioning on HumanML3D.** The summary-only model already matches the best motion captioner, and Motion-VLM scores highest under both judges, significantly ahead of MG-MotionLLM. All systems except the zero-shot model are trained on HumanML3D. Gray (×100): B@4 = BLEU-4, R-L = ROUGE-L, R-P@1 = R-Precision@1. Best in **bold**, second underlined.

			LLM judges		Reference-based			Retrieval
Method	Size	Input	MotionJudge↑	LLM judge↑	B@4↑	R-L↑	CIDEr↑	R-P@1↑
Motion captioning models								
MotionGPT	220M	motion	2.303	0.604	4.9	28.5	33.5	52.3
MotionGPT3	238M	motion	2.231	0.530	4.2	27.5	31.8	49.1
MG-MotionLLM	220M	motion	2.421	0.679	6.4	33.0	45.9	**59.8**
Motion-Agent	2B	motion	2.375	0.643	6.1	32.0	46.3	57.8
DEMO	8B	motion	1.925	0.321	2.9	18.4	23.2	35.1
Zero-shot video-language model								
Qwen3.5	4B	3-view video	1.743	0.164	1.4	16.5	4.9	9.3
Qwen3.5-4B fine-tuned on HumanML3D								
summary-only model	4B	3-view video	2.420	0.700	6.3	33.8	50.5	56.7
Motion-VLM	4B	3-view video + motion	**2.483**	**0.712**	**6.9**	**34.8**	**53.1**	57.9

Baselines. We compare against three groups. *Motion captioning models*, all trained on HumanML3D and run from their official checkpoints with repository-default decoding: MotionGPT (Jiang et al., 2023), MotionGPT3 (Zhu et al., 2026), MG-MotionLLM (Wu et al., 2025a) (coarse captioning checkpoint), Motion-Agent (Wu et al., 2025b), and DEMO (Xu et al., 2026). *Zero-shot video-language models*, which read renders of the same motions: Qwen3.5-4B (the base of Motion-VLM), Qwen3.5-27B, and Qwen3-VL-32B. A *summary-only* model: Qwen3.5-4B fine-tuned on the same renders, with no motion input, to output only the dataset's original caption, without azimuths or actions.

Implementation and evaluation. Motion-VLM is tested on three fixed frontal views and the motion sequence. MotionJudge (0–5) is our primary metric. As in prior motion-language work, we also report a whole-caption LLM judge on a 0–2 scale (the LLM judge). For comparability, we add the reference-based metrics BLEU-4, ROUGE-L, and CIDEr and, on HumanML3D, the retrieval metric R-Precision@1 (Guo et al., 2022a). Both judges use Qwen3.5-27B as their base model and score only the summary of a structured output. Prompts and setup details are in Appendices I.1 and B.

6.2 Main results

A general video-language model outperforms motion-specific captioners. Table 2 shows that on HumanML3D, Motion-VLM scores highest under both judges, ahead of the best published captioner, MG-MotionLLM. Both margins are significant and hold with a second base model (Appendix A). The summary-only model, the most direct baseline, already matches MG-MotionLLM on MotionJudge and exceeds it on the LLM judge, while its zero-shot base scores lowest. Once trained, a general video-language model thus captions motion as well as models built for it. The structured target adds a further gain here, and the motion tokens add theirs on Motion-X++ (Section 6.4).

In Table 2, only R-Precision@1 ranks MG-MotionLLM first, but retrieval text encoders barely react to a wrong fact (Section 3.3). BLEU-4 and ROUGE-L rank these systems almost as MotionJudge does, yet this agreement does not show that they detect wrong facts: HumanML3D's short references state few such facts.

6.3 Results on Motion-X++ and cross-dataset transfer

On Motion-X++, Table 3 shows that the summary-only model already scores far above zero-shot video-language models of up to 32B parameters, and Motion-VLM clearly above both: the motion facts come from training, not scale. Scored on the rear views given to the zero-shot models, Motion-VLM still leads (Appendix A). Unlike on HumanML3D, ROUGE-L and BERTScore here rank the zero-shot Qwen3.5-4B above every published captioner, whereas both judges place it near the bottom (Table 7). Table 4 shows that every trained model drops sharply off its training dataset, but the HumanML3D-trained Motion-VLM still scores highest on Motion-X++ under both judges.

Table 3: **Motion captioning on Motion-X++.** Summary-only: Qwen3.5-4B fine-tuned to output only the caption.

		LLM judges		Reference-based (×100)		
Method	Size	MotionJudge↑	LLM judge↑	B@4↑	R-L↑	CIDEr↑
Zero-shot video-language models						
Qwen3.5	4B	1.114	0.042	0.25	16.43	0.66
Qwen3.5	27B	1.248	0.053	0.55	17.92	1.15
Qwen3-VL	32B	1.349	0.070	0.75	18.11	2.66
Qwen3.5-4B fine-tuned on Motion-X++						
summary-only model	4B	1.592	0.259	10.23	27.34	20.58
Motion-VLM	4B	**1.805**	**0.349**	**11.13**	**28.00**	**23.81**

Table 4: **Cross-dataset transfer.** Source: the training dataset; target: the other dataset. Motion-VLM transfers best under both judges.

		LLM judge		MotionJudge	Reference-based (×100)	
Method	Size	Source↑	Target↑	Target↑	R-L↑	CIDEr↑
(a) HumanML3D → Motion-X++						
MotionGPT	220M	0.604	0.085	1.156	13.14	0.70
MotionGPT3	238M	0.530	0.069	1.216	13.49	0.78
MG-MotionLLM	220M	0.679	0.120	1.289	13.99	1.05
Motion-Agent	2B	0.643	0.133	1.216	15.45	1.16
DEMO	8B	0.321	0.043	0.938	8.45	0.32
Qwen3-VL (zero-shot)	32B	–	0.070	1.349	**18.11**	**2.66**
Motion-VLM	4B	**0.712**	**0.178**	**1.395**	15.49	1.11
(b) Motion-X++ → HumanML3D						
Qwen3.5 (zero-shot)	4B	0.042	0.164	1.743	**16.51**	**4.87**
Motion-VLM	4B	**0.349**	**0.204**	**1.862**	15.08	1.98

Table 5: **Ablations of the target and the motion input.** Top: training target and input, Δ against the row above. Bottom: Motion-VLM with its motion tokens changed at test time, Δ against Motion-VLM. Bold: significant under a paired bootstrap (Table 13).

	HumanML3D				Motion-X++			
	MotionJudge↑	Δ	LLM judge↑	Δ	MotionJudge↑	Δ	LLM judge↑	Δ
Training target and input								
video, summary-only target	2.420	–	0.700	–	1.592	–	0.259	–
video, structured target	2.465	**+0.045**	0.720	**+0.020**	1.584	−0.008	0.256	−0.003
video + motion, structured (Motion-VLM)	2.483	+0.018	0.712	−0.008	1.805	**+0.221**	0.349	**+0.093**
Motion-VLM, motion tokens at test time								
removed	2.474	−0.009	0.707	−0.005	1.575	**−0.230**	0.241	**−0.108**
zeroed	2.500	+0.017	0.717	+0.005	1.576	**−0.229**	0.240	**−0.109**
another clip's	2.479	−0.004	0.716	+0.004	1.485	**−0.320**	0.218	**−0.131**

The Motion-X++-trained Motion-VLM, whose motion projection expects the 52 Motion-X++ joints, runs on HumanML3D from video alone and still beats its zero-shot base under both judges.

6.4 Ablations

Table 5 tests the two model-side causes on both datasets, with the same data within each dataset. HumanML3D has short captions and 22 body joints that the renders show; Motion-X++ has paragraph-length references and 30 finger joints that the renders barely resolve.

Asking for the facts. Listing the camera azimuths and MotionFact actions before the summary raises both judges on HumanML3D, whose short captions omit such facts, and leaves Motion-X++ unchanged.

Seeing the evidence. On HumanML3D, three views instead of one give the largest gain of any ablation (Appendix B.4). In Table 5, motion tokens then add nothing significant there, as the renders already show every joint, but they raise both judges on Motion-X++. To check that the model reads the tokens, we remove them at test time, zero them, or feed another clip's motion (bottom of Table 5): on Motion-X++, every change lowers both judges significantly, and a wrong motion hurts more than none; on HumanML3D, nothing changes significantly. Appendix C.1 shows examples, and Appendix H discusses limitations.

7 Conclusion

Low scores on motion facts come from both weak models and insensitive metrics. On the metric side, n-gram and embedding-based metrics barely change under a wrong fact, whereas MotionJudge detects single-fact errors on every attribute. On the model side, Motion-VLM, a general video-language model that states MotionFact facts before its caption, outperforms five published motion captioners on HumanML3D under both judges. Each model-side choice helps where the data expose its cause: stating the facts helps on the short captions of HumanML3D, and motion tokens help on Motion-X++, where the renders barely resolve the hands.

Ethics Statement

This work involves no human-subject study. The only human input is the caption corrections that professional motion artists made on an industrial motion dataset, which contain no personal information. All motion data are skeleton sequences, and the videos we train on are untextured renders of them, so no identifying appearance is shown. HumanML3D and Motion-X++ are used according to their licenses. Fine-grained descriptions of human motion could support the monitoring of people in video; we do not target such applications.

Reproducibility Statement

The construction and validation of MotionADV are described in Section 3.2 and Appendix D, with examples in Appendix D.3; each sample comes with its validator votes and escalation record. We will release MotionADV and the code of MotionJudge and Motion-VLM. Appendix E gives the grammar and slots of MotionFact. Appendix F lists every prompt of MotionJudge and of the whole-caption LLM judge, the base model and its serving setup, and the cost, and Appendix F.8 repeats the evaluation with a second base model. For Motion-VLM, Table 11 gives the training, rendering, and input settings, Appendix I.2 the full prompt, Appendix B.2 the timestamp format, and Appendix B.3 how the slot values of the target are computed from the motion data. Baselines run from their official checkpoints with repository-default decoding (Section 6.1), and Appendix A reports every system with confidence intervals.

AI Use Statement

Large language models are part of the methods studied in this paper: they serve as judges (Sections 3 and 4), as validators of the MotionADV labels (Appendix D), and as the base of Motion-VLM (Section 5); their prompts are listed in Appendix I.1. In addition, we used generative AI assistants for three purposes: data synthesis (proposing candidate caption edits and translations for MotionADV, all of which pass the validation of Section 3.2), coding assistance for training, evaluation, and analysis scripts, and polishing the writing of the manuscript. We did not use generative AI tools to design the experiments or to produce the reported results. The authors reviewed all AI-assisted text, verified every reference against its source, and checked every reported number against the experiment records. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

References

Chantal Amrhein, Nikita Moghe, and Liane Guillou. Aces: Translation accuracy challenge sets for evaluating machine translation metrics. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pp. 479–513, 2022.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Yong Cao, Chuqiao Li, Xianghui Xie, Gerard Pons-Moll, and Andreas Geiger. Nextmotionqa: Benchmarking and judging human motion understanding with vision-language models. *arXiv preprint arXiv:2606.04773*, 2026.

Wenhao Chai, Enxin Song, Yilun Du, Chenlin Meng, Vashisht Madhavan, Omer Bar-Tal, Jenq-Neng Hwang, Saining Xie, and Christopher Manning. Auroracap: Efficient, performant video detailed captioning and a new benchmark. In *International Conference on Learning Representations*, volume 2025, pp. 16845–16889, 2025.

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. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Jaemin Cho, Yushi Hu, Jason Baldridge, Roopal Garg, Peter Anderson, Ranjay Krishna, Mohit Bansal, Jordi Pont-Tuset, and Su Wang. Davidsonian scene graph: Improving reliability in fine-grained evaluation for text-to-image generation. In *International conference on learning representations*, volume 2024, pp. 15625–15645, 2024.

Jonathan Cook, Tim Rocktäschel, Jakob Foerster, Dennis Aumiller, and Alex Wang. Ticking all the boxes: Generated checklists improve llm evaluation and generation. *arXiv preprint arXiv:2410.03608*, 2024.

Ginger Delmas, Philippe Weinzaepfel, Thomas Lucas, Francesc Moreno-Noguer, and Grégory Rogez. Posescript: 3d human poses from natural language. In *European Conference on Computer Vision*, pp. 346–362. Springer, 2022.

Hongyuan Dong, Jiawen Li, Bohong Wu, Jiacong Wang, Yuan Zhang, and Haoyuan Guo. Benchmarking and improving detail image caption. *arXiv preprint arXiv:2405.19092*, 2024.

Kent Fujiwara, Mikihiro Tanaka, and Qing Yu. Chronologically accurate retrieval for temporal grounding of motion-language models. In *European Conference on Computer Vision*, pp. 323–339. Springer, 2024.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, et al. A survey on LLM-as-a-judge. *The Innovation*, 7(6), 2026.

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 *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5142–5151. IEEE, 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 *European conference on computer vision*, pp. 580–597. Springer, 2022b.

Weile Guo, Shenghong He, Danying Mo, Chengdong Xu, Xuexun Liu, and Chao Yu. Motionhalluc: Diagnosing kinematic hallucinations in fine-grained motion reasoning. *arXiv preprint arXiv:2606.23061*, 2026.

Wenyi Hong, Yean Cheng, Zhuoyi Yang, Weihan Wang, Lefan Wang, Xiaotao Gu, Shiyu Huang, Yuxiao Dong, and Jie Tang. Motionbench: Benchmarking and improving fine-grained video motion understanding for vision language models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8450–8460. IEEE, 2025.

Cheng-Yu Hsieh, Jieyu Zhang, Zixian Ma, Aniruddha Kembhavi, and Ranjay Krishna. Sugarcrepe: Fixing hackable benchmarks for vision-language compositionality. *Advances in neural information processing systems*, 36:31096–31116, 2023.

Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A Smith. Tifa: Accurate and interpretable text-to-image faithfulness evaluation with question answering. In *2023 ieee/cvf international conference on computer vision (iccv)*, pp. 20349–20360. IEEE, 2023.

Mengdi Jia, Zekun Qi, Shaochen Zhang, Wenyao Zhang, Xinqiang Yu, Jiawei He, He Wang, and Li Yi. Omnispatial: Towards comprehensive spatial reasoning benchmark for vision language models. In *International Conference on Learning Representations*, volume 2026, pp. 35634–35670, 2026.

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.

Biao Jiang, Xin Chen, Chi Zhang, Fukun Yin, Zhuoyuan Li, Gang Yu, and Jiayuan Fan. Motionchain: Conversational motion controllers via multimodal prompts. In *European Conference on Computer Vision*, pp. 54–74. Springer, 2024.

Marzena Karpinska, Nishant Raj, Katherine Thai, Yixiao Song, Ankita Gupta, and Mohit Iyyer. Demetr: Diagnosing evaluation metrics for translation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 9540–9561, 2022.

Mert Kilickaya, Aykut Erdem, Nazli Ikizler-Cinbis, and Erkut Erdem. Re-evaluating automatic metrics for image captioning. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pp. 199–209, 2017.

Seungone Kim, Jay Shin, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Ryan Shin, Sungdong Kim, James Thorne, Minjoon Seo, et al. Prometheus: Inducing fine-grained evaluation capability in language models. In *International Conference on Learning Representations*, volume 2024, pp. 29927–29962, 2024.

Fei Kong, Jinhao Duan, Kaidi Xu, Zhenhua Guo, Xiaofeng Zhu, and Xiaoshuang Shi. Lrr-bench: Left, right or rotate? vision-language models still struggle with spatial understanding tasks. *arXiv preprint arXiv:2507.20174*, 2025.

Rudolf Laban and Lisa Ullmann. *The mastery of movement.* ERIC, 1971.

Jongseo Lee, Hyuntak Lee, Sunghun Kim, Sooa Kim, Jihoon Chung, and Jinwoo Choi. Which way did it move? diagnosing and overcoming directional motion blindness in video-llms. *arXiv preprint arXiv:2605.22823*, 2026.

Yukyung Lee, Joonghoon Kim, Jaehee Kim, Hyowon Cho, Jaewook Kang, Pilsung Kang, and Najoung Kim. Checkeval: A reliable llm-as-a-judge framework for evaluating text generation using checklists. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 15771–15798, 2025.

Lei Li, Sen Jia, Jianhao Wang, Zhongyu Jiang, Feng Zhou, Ju Dai, Tianfang Zhang, Zongkai Wu, and Jenq-Neng Hwang. Human motion instruction tuning. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17582–17591. IEEE, 2025a.

Zhe Li, Weihao Yuan, Yisheng He, Lingteng Qiu, Shenhao Zhu, Xiaodong Gu, Weichao Shen, Yuan Dong, Zilong Dong, and Laurence Yang. Lamp: Language-motion pretraining for motion generation, retrieval, and captioning. In *International Conference on Learning Representations*, volume 2025, pp. 84238–84250, 2025b.

Boda Lin, Yongjie Zhu, Xiaocheng Gong, Wenyu Qin, and Meng Wang. Kpm-bench: A kinematic parsing motion benchmark for fine-grained motion-centric video understanding. *arXiv preprint arXiv:2602.17768*, 2026.

Xinpeng Liu, Yong-Lu Li, Ailing Zeng, Zizheng Zhou, Yang You, and Cewu Lu. Bridging the gap between human motion and action semantics via kinematic phrases. In *European Conference on Computer Vision*, pp. 223–240. Springer, 2024.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-eval: Nlg evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 2511–2522, 2023.

Mingshuang Luo, Ruibing Hou, Zhuo Li, Hong Chang, Zimo Liu, Yaowei Wang, and Shiguang Shan. M^3GPT: An advanced multimodal, multitask framework for motion comprehension and generation. *Advances in Neural Information Processing Systems*, 37:28051–28077, 2024.

Thomas Markhorst, Zhi-Yi Lin, Jouh Yeong Chew, Jan van Gemert, and Xucong Zhang. Fine-grained human motion understanding with language models. *arXiv preprint arXiv:2606.20888*, 2026.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 12076–12100, 2023.

Jiawei Mo, Yixuan Chen, Rifen Lin, Yongkang Ni, Feng Liang, Min Zeng, Xiping Hu, and Min Li. Mochat: Joints-grouped spatio-temporal grounding multimodal large language model for multi-turn motion comprehension and description. *IEEE Journal of Biomedical and Health Informatics*, 2025.

Mathis Petrovich, Michael J Black, and Gül Varol. Tmr: Text-to-motion retrieval using contrastive 3d human motion synthesis. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9454–9463. IEEE, 2023.

Suzanne Petryk, David M Chan, Anish Kachinthaya, Haodi Zou, John Canny, Joseph E Gonzalez, and Trevor Darrell. Aloha: A new measure for hallucination in captioning models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pp. 342–357, 2024.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of nlp models with checklist. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pp. 4902–4912, 2020.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object hallucination in image captioning. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 4035–4045, 2018.

Ananya B Sai, Tanay Dixit, Dev Yashpal Sheth, Sreyas Mohan, and Mitesh M Khapra. Perturbation checklists for evaluating nlg evaluation metrics. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 7219–7234, 2021.

Ravi Shekhar, Sandro Pezzelle, Yauhen Klimovich, Aurélie Herbelot, Moin Nabi, Enver Sangineto, and Raffaella Bernardi. Foil it! find one mismatch between image and language caption. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 255–265, 2017.

Sijun Tan, Siyuan Zhuang, Kyle Montgomery, William Tang, Alejandro Cuadron, Chenguang Wang, Raluca Popa, and Ion Stoica. Judgebench: A benchmark for evaluating llm-based judges. In *International Conference on Learning Representations*, volume 2025, pp. 63277–63303, 2025.

Gemma Team, Sherif El Abd, Vaibhav Aggarwal, Robin Algayres, Alek Andreev, Olivier Bachem, Ian Ballantyne, Cormac Brick, Victor Cărbune, Michelle Casbon, et al. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026.

Aman Singh Thakur, Kartik Choudhary, Venkat Srinik Ramayapally, Sankaran Vaidyanathan, and Dieuwke Hupkes. Judging the judges: Evaluating alignment and vulnerabilities in llms-as-judges. In *Proceedings of the Fourth Workshop on Generation, Evaluation and Metrics (GEM^2)*, pp. 404–430, 2025.

Chongjun Tu, Lin Zhang, Peng Ye, Xianfang Zeng, Wei Cheng, Gang Yu, Tao Chen, et al. Favor-bench: A comprehensive benchmark for fine-grained video motion understanding. *Advances in Neural Information Processing Systems*, 38, 2026.

Jordan Voas, Yili Wang, Qixing Huang, and Raymond Mooney. What is the best automated metric for text to motion generation? In *SIGGRAPH Asia 2023 Conference Papers*, pp. 1–11, 2023.

Jiawei Wang, Liping Yuan, Yuchen Zhang, and Haomiao Sun. Tarsier: Recipes for training and evaluating large video description models. *arXiv preprint arXiv:2407.00634*, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Bizhu Wu, Jinheng Xie, Keming Shen, Zhe Kong, Jianfeng Ren, Ruibin Bai, Rong Qu, and Linlin Shen. Mg-motionllm: A unified framework for motion comprehension and generation across multiple granularities. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 27849–27858. IEEE, 2025a.

Qi Wu, Yubo Zhao, Yifan Wang, Xinhang Liu, Yu-Wing Tai, and Chi-Keung Tang. Motion-agent: A conversational framework for human motion generation with llms. In *International Conference on Learning Representations*, volume 2025, pp. 48023–48043, 2025b.

Shiyao Xu, Benedetta Liberatori, Gül Varol, and Paolo Rota. Dense motion captioning. In *2026 International Conference on 3D Vision (3DV)*, pp. 1750–1759. IEEE, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Payam Jome Yazdian, Rachel Lagasse, Hamid Mohammadi, Eric Liu, Li Cheng, and Angelica Lim. Motionscript: Natural language descriptions for expressive 3d human motions. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 21574–21581. IEEE, 2025.

Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? *arXiv preprint arXiv:2210.01936*, 2022.

Zhiyuan Zeng, Jiatong Yu, Tianyu Gao, Yu Meng, Tanya Goyal, and Danqi Chen. Evaluating large language models at evaluating instruction following. In *International Conference on Learning Representations*, volume 2024, pp. 40193–40219, 2024.

Pengfei Zhang, Pinxin Liu, Pablo Garrido, Hyeongwoo Kim, and Bindita Chaudhuri. Kinmo: Kinematic-aware human motion understanding and generation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11187–11197. IEEE, 2025a.

Yuhong Zhang, Jing Lin, Ailing Zeng, Guanlin Wu, Shunlin Lu, Yurong Fu, Yuanhao Cai, Ruimao Zhang, Haoqian Wang, and Lei Zhang. Motion-x++: A large-scale multimodal 3d whole-body human motion dataset. *arXiv preprint arXiv:2501.05098*, 2025b.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Bingfan Zhu, Biao Jiang, Sunyi Wang, Shixiang Tang, Tao Chen, Linjie Luo, Youyi Zheng, and Xin Chen. Motiongpt3: Human motion as a second modality. In *International Conference on Learning Representations*, volume 2026, pp. 12631–12657, 2026.

A Full Quantitative Results

This appendix supports the comparisons of Sections 6.2 and 6.3: it describes how the baselines are run, gives the paired margins on HumanML3D, and reports all metrics in Tables 6 and 7.

Baselines. Every published system runs from its officially released checkpoint with repository-default decoding, and none is retrained. For MotionGPT, our setup reproduces the released generations exactly on 176 clips. DEMO runs its stage-2 weights, which reproduce the authors' released outputs. MotionGPT3 keeps its default sampled decoding, and Motion-Agent runs on Gemma-2-2B. MG-MotionLLM has a coarse (m2t) and a fine (m2dt) captioning checkpoint. We report the fine checkpoint only in the full tables, since it emits a limb-level script rather than a sentence.

Margins on HumanML3D. In a paired bootstrap over motions, Motion-VLM exceeds MG-MotionLLM by +0.057 [+0.019, +0.096] on MotionJudge and +0.033 [+0.015, +0.052] on the LLM judge. With gemma-4-31B as the base, the MotionJudge margin is +0.100 [+0.058, +0.141] and the LLM-judge margin +0.039 [+0.020, +0.057]. Without the 171 motions whose skeletons are stored mirrored, the LLM-judge margin is +0.028 [+0.009, +0.047]. Motion-VLM exceeds the summary-only model by +0.061 [+0.030, +0.093] on MotionJudge (+0.043 [+0.009, +0.079] with gemma-4-31B). The video-only structured model of Table 5 exceeds MG-MotionLLM by +0.044 [+0.006, +0.082] on MotionJudge.

Ablations. On HumanML3D, three views instead of one raise MotionJudge from 2.252 to 2.505, and the structured target gains +0.045 [+0.014, +0.076] on MotionJudge and +0.020 [+0.004, +0.036] on the LLM judge. Predicting azimuths with the nine-camera pool raises the LLM judge from 0.701 to 0.720. On Motion-X++, removing the motion tokens at test time lowers the LLM judge by 0.077 [0.059, 0.096] (Appendix F.10).

Table 6: **All metrics on the HumanML3D test split.** Main-text Table 2 shows a subset. Reference-based and LLM-judge columns average each motion with its mirror (n = 2,192). Retrieval uses the T2M evaluator of Guo et al. (2022a), which admits clips of 40–199 frames (n = 4,198). Bold: best system per column; italic rows are not ranked. B@k = BLEU-k, R-L = ROUGE-L, R-P@k = R-Precision@k, MM-Dist = MM-Distance.

		Reference-based					T2M retrieval (n = 4,198)				LLM judges	
System	Input	B@1 ↑	B@4 ↑	R-L ↑	CIDEr ↑	BERTScore ↑	R-P@1 ↑	R-P@2 ↑	R-P@3 ↑	MM-Dist ↓	MotionJudge ↑	LLM judge ↑
Motion captioning models, trained on HumanML3D												
MG-MotionLLM (coarse)	motion	**0.2480**	0.0642	0.3299	0.4585	0.9023	**0.5984**	**0.7888**	**0.8708**	**2.517**	2.421	0.679
Motion-Agent	motion	0.2352	0.0613	0.3203	0.4633	0.9022	0.5779	0.7672	0.8514	2.618	2.375	0.643
MotionGPT	motion	0.2397	0.0490	0.2847	0.3351	0.8940	0.5234	0.7096	0.7976	2.975	2.303	0.604
MotionGPT3	motion	0.2133	0.0419	0.2753	0.3179	0.8921	0.4908	0.6736	0.7677	3.115	2.231	0.530
DEMO	motion, 20 fps	0.1835	0.0291	0.1845	0.2321	0.8714	0.3508	0.5256	0.6332	4.339	1.925	0.321
MG-MotionLLM (fine)	motion	*0.0179*	*0.0012*	*0.0347*	*0.0002*	*0.7983*	*0.0461*	*0.0885*	*0.1285*	*8.116*	*1.475*	*0.114*
Zero-shot video-language model												
Qwen3.5-4B (zero-shot)	3-view video	0.2030	0.0142	0.1651	0.0487	0.8504	0.0929	0.1687	0.2321	6.854	1.743	0.164
Qwen3.5-4B fine-tuned on HumanML3D												
Summary-only model	3-view video	0.2201	0.0625	0.3384	0.5049	0.9045	0.5674	0.7543	0.8402	2.758	2.420	0.700
Motion-VLM	3-view video + motion	0.2415	**0.0685**	**0.3484**	**0.5314**	**0.9066**	0.5792	0.7686	0.8534	2.673	**2.483**	**0.712**
Controls												
Reference as prediction (ceiling)	–	–	–	–	–	–	*0.5173*	*0.7085*	*0.8014*	*2.936*	*5.000*†	*2.000*
Random-clip reference (floor)	–	–	–	–	–	–	*0.0331*	*0.0623*	*0.0904*	*9.507*	*0.975*†	*0.040*
Repeat noise |Δ|	–	–	–	–	–	–	–	–	–	–	*0.005*†	*0.0025*

†MotionJudge controls use a fixed 200-clip subset, and LLM-judge controls use 400 clips. The retrieval controls reproduce the published real-caption row of Guo et al. (2022a) (0.511/0.703/0.797, MM-Dist 2.974), and the random-clip reference lands at chance. Empty predictions are scored, not dropped (MotionGPT3 emits 5; DEMO refuses 26 clips shorter than 20 frames). MotionJudge drops clips whose verdict fails to parse, so its n varies by system.

Viewpoints on Motion-X++. The test viewpoint does not change the Motion-X++ comparison. Motion-VLM is scored on the three frontal views throughout, whereas the scores of the zero-shot video-language models, under both judges, come from rear-view renders. Scored on the same rear views in the same judging session, Motion-VLM obtains 1.745 on MotionJudge, 0.062 [0.015, 0.115] below its frontal score.

Table 7: **All metrics on the Motion-X++ test set.** Similarity metrics are recomputed with one implementation and one reference set on the same 2,279 clips. Retrieval columns are a caption-to-caption proxy over 32 candidates, not the HumanML3D text–motion protocol. Bold: best per column; underline: best published or zero-shot system. Column names as in Table 6; R@k-T and MM-T are the proxy's R-Precision@k and MM-Distance.

		Reference-based					Text retrieval proxies			LLM judges	
System	Input	B@1 ↑	B@4 ↑	R-L ↑	CIDEr ↑	BERTScore ↑	R@1-T ↑	R@3-T ↑	MM-T ↓	MotionJudge ↑	LLM judge ↑
Motion captioning models, trained on HumanML3D											
Motion-Agent	motion	0.0405	0.0054	0.1545	0.0116	0.8686	0.1039	0.2359	0.9810	1.216	0.1325
MG-MotionLLM (coarse)	motion	0.0416	0.0031	0.1399	0.0105	0.8658	0.1175	0.2443	0.9808	1.289	0.1202
MotionGPT	motion	0.0466	0.0037	0.1314	0.0070	0.8613	0.0968	0.2139	0.9970	1.156	0.0851
MotionGPT3	motion	0.0367	0.0027	0.1349	0.0078	0.8578	0.1056	0.2227	1.0001	1.216	0.0693
DEMO	motion, 20 fps	0.0300	0.0011	0.0845	0.0032	0.8390	0.0761	0.1813	1.0474	0.938	0.0430
MG-MotionLLM (fine)	motion	0.0307	0.0007	0.0479	0.0001	0.7907	0.0506	0.1369	1.0444	1.047	0.0369
Zero-shot video-language models, renders of the same motions											
Qwen3-VL-32B	3-view video	0.1860	0.0075	0.1811	0.0266	0.8691	0.0722	0.1747	0.9753	1.349	0.0700
Qwen3.5-27B†	3-view video	0.1169	0.0055	0.1792	0.0115	0.8702	0.0599	0.1501	0.9912	1.248	0.0534
Qwen3.5-4B	3-view video	0.0681	0.0025	0.1643	0.0066	0.8710	0.0585	0.1496	0.9950	1.114	0.0420
Qwen3.5-4B fine-tuned on Motion-X++											
Motion-VLM (4B)	3-view video + motion	**0.3502**	**0.1113**	**0.2800**	**0.2381**	**0.8881**	**0.3539**	**0.5550**	**0.8333**	**1.805**	**0.3488**
Controls											
Reference as prediction (ceiling)	–	–	–	–	–	–	–	–	–	*4.999*	*2.00*
Neighboring-clip reference (floor)	–	–	–	–	–	–	–	–	–	*1.755*	*0.133*
Random-clip reference (floor)	–	–	–	–	–	–	–	–	–	*0.460*	*0.010*
Repeat noise |Δ|	–	–	–	–	–	–	–	–	–	*0.021*	*0.000*

†Same base model as both judges. MotionJudge was run on 2,286 clips ($n \approx 2{,}250$ per system after unparseable verdicts are dropped). Its controls use all 2,286 clips, and the LLM-judge controls use 400 clips (Appendix F.10). Two handicaps against the baselines are not repaired: MotionGPT3 truncates 6.6% of clips at 498 frames, and MG-MotionLLM was trained on motions of at most 199 frames.

Ranking of the captioning models. Table 8 compares how each metric and both judges rank the captioning models by their mean scores. On HumanML3D every metric ranks the nine models close to MotionJudge (Spearman 0.90 to 1.00). On Motion-X++ the agreement with MotionJudge ranges from 0.56 (R-Precision) to 0.92 (CIDEr, MM-Distance), and MotionJudge and the LLM judge agree at 0.72.

Per-attribute verdicts. Tables 9 and 10 break the MotionJudge verdicts down by attribute. The lead of Motion-VLM comes mainly from stating more reference facts correctly, while each fact it states is wrong no more often than for the baselines. On HumanML3D it contradicts about as often as the

Table 8: **How each metric ranks the captioning models, compared with both judges.** Each model enters with its mean score over the test set; the table gives the Spearman correlation of these scores between each metric and MotionJudge or the LLM judge, over the models of Table 6 (HumanML3D) and Table 7 (Motion-X++), controls excluded. *On HumanML3D, the T2M evaluator of Guo et al. (2022a) (R-Precision@2 gives the same values as @1 and @3); on Motion-X++, the caption-to-caption proxy.

	HumanML3D (9 models)		Motion-X++ (10 models)	
Metric	MotionJudge	LLM judge	MotionJudge	LLM judge
BLEU-1	0.90	0.83	0.77	0.43
BLEU-4	1.00	0.98	0.88	0.75
ROUGE-L	0.98	1.00	0.81	0.48
CIDEr	0.95	0.98	0.92	0.79
BERTScore	0.98	1.00	0.64	0.41
R-Precision@1*	0.97	0.92	0.56	0.85
R-Precision@3*	0.97	0.92	0.56	0.90
MM-Distance* (negated)	0.92	0.87	0.92	0.73
LLM judge	0.98	–	0.72	–

Table 9: **Per-attribute MotionJudge verdicts on both datasets.** Each cell gives contradicted / omitted in %. *Contradicted*: share of the motions whose reference states the attribute and whose caption contradicts a reference fact of it. *Omitted*: share of the same motions whose caption contradicts nothing but leaves out an action that carries the fact. A caption that states more facts has more chances to contradict one. *n* is the number of such motions for Motion-VLM and differs by at most 2% for the other systems. HumanML3D averages each motion with its mirror; on Motion-X++ the unit is a clip and, as in Table 3, the zero-shot models read the rear views. On Motion-X++, rotation is stated in only three references and is left out, also from *All*.

Method	Side	Direction	Rotation	Speed	Motion type	Bilateral mode	Oscillation axis	Articulation	All
HumanML3D									
n	1,580	1,637	248	281	2,186	1,196	401	152	7,681
MotionGPT	1.6 / 79.5	4.0 / 68.3	0.6 / 61.3	2.7 / 46.8	12.1 / 75.9	1.8 / 58.5	1.6 / 65.8	0.0 / 73.7	5.1 / 70.2
MotionGPT3	1.1 / 81.6	4.1 / 70.0	0.6 / 65.2	3.6 / 51.1	10.4 / 78.9	3.0 / 59.4	1.9 / 68.9	0.0 / 83.0	4.8 / 72.6
MG-MotionLLM	1.5 / 77.3	4.2 / 64.0	1.2 / 58.9	4.8 / 43.6	13.2 / 73.6	1.8 / 53.1	0.5 / 64.5	0.0 / 77.3	5.5 / 67.2
Motion-Agent	1.6 / 77.8	3.9 / 64.2	1.0 / 62.5	5.0 / 48.8	12.1 / 74.8	2.7 / 53.8	2.4 / 68.1	0.0 / 77.3	5.3 / 68.2
DEMO	2.1 / 85.3	3.3 / 75.9	2.0 / 76.8	4.8 / 56.1	8.0 / 83.2	0.8 / 65.8	3.5 / 72.7	0.0 / 87.2	4.0 / 77.7
Qwen3.5-4B (zero-shot)	1.8 / 91.6	2.2 / 84.9	0.0 / 97.0	0.0 / 76.3	2.8 / 93.3	2.6 / 80.4	0.1 / 94.6	0.0 / 85.5	2.0 / 88.6
summary-only model	1.5 / 77.2	4.0 / 63.9	0.4 / 61.1	3.4 / 45.0	13.0 / 73.5	3.4 / 54.1	0.6 / 66.1	0.0 / 80.9	5.6 / 67.5
Motion-VLM	1.6 / 76.0	3.8 / 62.0	0.6 / 58.3	2.5 / 44.7	13.5 / 72.8	4.2 / 52.4	0.9 / 65.0	1.3 / 79.9	5.8 / 66.2
Motion-X++									
n	1,540	1,194	3	85	2,265	1,226	957	421	7,688
Qwen3.5-4B (zero-shot)	2.4 / 95.7	0.3 / 94.5	–	0.0 / 94.0	2.6 / 97.1	0.0 / 94.5	1.4 / 95.8	0.0 / 97.6	1.5 / 95.8
Qwen3.5-27B (zero-shot)	4.2 / 93.6	0.7 / 92.5	–	0.0 / 95.3	3.0 / 96.8	0.0 / 92.5	0.3 / 95.4	0.0 / 97.4	1.9 / 94.7
Qwen3-VL-32B (zero-shot)	7.7 / 90.2	5.9 / 88.9	–	0.0 / 94.1	4.2 / 95.3	1.1 / 92.0	1.3 / 93.4	0.0 / 94.5	4.0 / 92.5
summary-only model	1.2 / 83.2	4.7 / 79.6	–	0.0 / 75.3	11.3 / 84.7	2.6 / 77.7	2.2 / 74.5	1.0 / 76.2	5.0 / 80.7
Motion-VLM	1.4 / 79.7	5.2 / 77.2	–	0.0 / 76.5	12.7 / 81.5	2.7 / 74.3	2.5 / 69.1	0.5 / 75.1	5.6 / 77.4

summary-only model and MG-MotionLLM (5.8% of motion–attribute pairs, against 5.6% and 5.5%) but omits less (66.2% against 67.5% and 67.2%). At the fact level it supports more reference facts than the summary-only model (23.9% against 22.5%; paired difference +1.5 [+0.9, +2.0] points), while its contradiction rate changes by +0.2 [−0.1, +0.4]. On Motion-X++ it likewise omits less than the summary-only model (77.4% against 80.7%) without contradicting less (5.6% against 5.0%). Per stated fact, its error rate is at or below that of the baselines (Table 10): 12.4% against 12.5% for the summary-only model under MotionJudge, 26.3% against 27.7% for MG-MotionLLM and the summary-only model with the gemma-4-31B base, and 12.8% against 13.3% on Motion-X++. Video models are not worse than motion models on side and direction. The summary-only model, which reads only video, contradicts and omits side in 1.5% and 77.2% of pairs and direction in 4.0% and 63.9%, against 1.5%, 77.3%, 4.2%, and 64.0% for MG-MotionLLM. The paired differences in side and direction contradiction between Motion-VLM and MG-MotionLLM, +0.1 [−0.6, +0.9] and −0.4 [−1.6, +0.8] points, both include zero.

Table 10: **Fact-level MotionJudge verdicts** for the systems whose per-fact records we kept. Each triple gives the share (%) of the reference facts of an attribute that the caption supports / contradicts / does not mention; the rest (at most 5%) states the value of another reference fact of the same attribute. HumanML3D facts are pooled over each motion and its mirror; on Motion-X++ the zero-shot model reads the rear views. *Errors among stated facts*: contradicted over supported plus contradicted, in %. The last row counts the scored reference facts per clip (on Motion-X++, per clip with at least one such fact: 2,265 of 2,286).

	HumanML3D, MotionJudge (Qwen3.5-27B)		HumanML3D, MotionJudge (gemma-4-31B)			Motion-X++, MotionJudge (Qwen3.5-27B)		
Attribute	summary-only	Motion-VLM	MG-MotionLLM	summary-only	Motion-VLM	Qwen3.5-27B (zero-shot)	summary-only	Motion-VLM
Side	17.0 / 0.8 / 82.0	18.4 / 0.9 / 80.3	20.8 / 2.2 / 75.2	20.7 / 2.5 / 75.2	23.0 / 2.3 / 73.0	0.5 / 1.8 / 97.6	10.5 / 0.5 / 88.8	12.1 / 0.6 / 87.1
Direction	33.5 / 2.2 / 63.6	35.5 / 2.1 / 61.7	42.0 / 6.0 / 49.4	40.9 / 5.5 / 51.2	43.8 / 5.6 / 48.1	15.6 / 0.3 / 84.0	17.3 / 2.2 / 79.7	19.1 / 2.3 / 77.7
Rotation	4.6 / 0.3 / 94.7	6.5 / 0.5 / 93.0	6.3 / 1.7 / 91.3	5.4 / 1.5 / 92.2	6.7 / 1.9 / 91.0	–	–	–
Speed	12.9 / 3.1 / 84.0	15.2 / 2.2 / 82.6	19.4 / 6.0 / 74.7	13.4 / 4.3 / 82.3	15.7 / 3.8 / 80.5	0.0 / 0.0 / 100.0	2.6 / 0.0 / 97.4	2.6 / 0.0 / 97.4
Motion type	27.7 / 5.5 / 65.2	29.0 / 5.7 / 63.6	28.5 / 19.6 / 49.2	27.6 / 18.8 / 50.7	29.4 / 18.9 / 48.8	7.3 / 0.7 / 91.2	18.7 / 3.5 / 73.7	21.2 / 3.8 / 70.2
Bilateral mode	2.3 / 2.5 / 94.9	3.4 / 3.3 / 93.0	2.1 / 0.6 / 97.1	1.8 / 0.4 / 97.6	2.0 / 0.4 / 97.4	0.2 / 0.0 / 99.8	11.9 / 1.8 / 85.9	13.8 / 1.7 / 84.4
Oscillation axis	11.8 / 0.6 / 87.7	13.5 / 0.8 / 85.7	16.3 / 2.3 / 81.4	12.8 / 3.9 / 83.4	12.5 / 3.6 / 83.9	0.2 / 0.2 / 99.6	5.8 / 1.4 / 92.2	6.5 / 1.6 / 91.7
Articulation	8.8 / 0.0 / 90.6	10.6 / 1.2 / 87.6	13.8 / 0.3 / 85.6	10.0 / 0.0 / 90.0	10.0 / 0.3 / 89.7	1.8 / 0.0 / 98.2	13.9 / 0.7 / 85.0	14.7 / 0.4 / 84.8
All	22.5 / 3.2 / 73.5	23.9 / 3.4 / 71.7	25.6 / 9.8 / 62.5	24.7 / 9.5 / 63.9	26.5 / 9.5 / 62.0	5.6 / 0.7 / 93.3	15.0 / 2.3 / 80.5	17.0 / 2.5 / 78.1
Errors among stated facts	12.5	12.4	27.7	27.7	26.3	11.5	13.3	12.8
Facts per clip	6.73	6.73	6.73	6.73	6.73	9.59	9.59	9.59

B Implementation and Ablations of Motion-VLM

This appendix specifies Motion-VLM (Section 5), gives the intervals behind its ablations (Section 6.4), and reports how training affects general video understanding.

B.1 Implementation details

Table 11 lists the settings of Motion-VLM. Video-only variants keep the same template with the motion input absent, so view sampling is identical between them.

Table 11: **Motion-VLM settings.**

Model	
Base model	Qwen3.5-4B (Qwen Team, 2026), the successor of Qwen3 (Yang et al., 2025); trained with ms-swift
LoRA	rank 32, scale 64; the four attention projections of the eight full-attention layers and the five projections of the 24 linear-attention layers
Motion projection	one linear layer trained from scratch, $66 \rightarrow 2560$ (HumanML3D, 22 joints) or $156 \rightarrow 2560$ (Motion-X++, 52 joints); saved with the adapter
New tokens	motion start and end tokens, trainable
Training	
Optimizer	AdamW, learning rate 10^{-4}, 40 warm-up updates, bfloat16
Schedule	at most three epochs; cosine decay to zero after the warm-up; length-grouped batches
HumanML3D	Motion-VLM: eight GPUs, 26,184 updates; ablation variants: six GPUs, 34,914 updates; one example per GPU, no gradient accumulation
Motion-X++	global batch of eight for every variant
Motion dropout	0.3 per example for Motion-VLM
Rendering	
Body	untextured, rendered in Blender; left limbs blue, right limbs and torso orange
Video	384×384, 10 fps, H.264
Camera pool	eight fixed cameras every 45° and one tracking camera
Training views	three sampled per example from the pool
Test views	front-left, front, and front-right of the actor's initial facing
Input	
Video	read at 5 fps, 4 to 128 frames, at most 100,352 pixels per frame
Motion	root-relative joint positions normalized by the mean and standard deviation of each dataset, sampled at the video frame times
Evaluation	
Intervals	HumanML3D: each motion averaged with its mirror (n = 2,192); Motion-X++: bootstrap over clips

B.2 Timestamps and motion dropout

Timestamps. Qwen3-VL aligns video with time through text: its processor writes a timestamp string such as <2.4 seconds> before each temporal patch of a video (Bai et al., 2025). We sample the motion at the video frame times and precede every motion token with a timestamp string of the same form, so the model reads poses and frames on one time axis, in a notation it already knows. Motion tokens and video frames do not share position ids, so only the timestamps link them. Motion keeps one token per sampled frame, whereas the video merges two frames into each temporal patch, so the motion tokens have twice the time resolution of the video.

Motion dropout. With probability 0.3 a training example loses the whole motion block together with its lead-in sentence, so the prompt never announces a stream that is absent. The same template, without the block, serves video-only inference.

B.3 Motion-derived slot values

The six slots of the structured target (Section 5.2) are computed from the joint trajectories. Side, direction, and rotation are derived geometrically, and count, speed, and amplitude from trajectory statistics, with speed and amplitude binned into per-dataset terciles. A slot is left empty when the computation cannot decide, and the side, direction, and rotation slots of clips stored mirrored are left empty (Appendix F.10). Predicting the azimuths requires the model to identify which viewpoint each video shows, and the actions give it noisy but caption-independent supervision from the motion data. On the evaluation split, the computed side is defined for 63% of eligible clips and agrees with the side stated in the human caption in 119 of 127 cases (0.94). This measures agreement between the motion data and the text, not accuracy against human labels of the motion.

B.4 Views and camera azimuths

A single view can hide a limb or leave left and right ambiguous; Table 12 varies the training views and whether the model predicts their azimuths. Three views instead of one give the largest gain. Sampling from the nine-camera pool pays off only with azimuths: on Motion-X++, whose clips include running and large-range movement, it clearly outperforms three fixed cameras, while sampling without azimuths does not. On HumanML3D, azimuths also help sampled views, but three fixed cameras score 2.505 on MotionJudge, above Motion-VLM (2.483). We keep the sampled pool and the azimuth target because they let one recipe handle the locomotion of Motion-X++, at a cost of 0.026 on the HumanML3D LLM judge. On Motion-X++, the azimuth target is tested only with the sampled pool. Motion alone, without renders, scores 2.399 and 0.657 on HumanML3D, below video alone (2.465 and 0.720). Because MotionJudge costs about 17 LLM calls per caption, the Motion-X++ view variants are scored only by the LLM judge, which ranks systems more reliably than individual samples (Thakur et al., 2025).

Table 12: **Training views and azimuths.** Input: video on HumanML3D, video and motion on Motion-X++.

	no azimuth		with azimuth	
Training views	MotionJudge↑	LLM judge↑	MotionJudge↑	LLM judge↑
HumanML3D				
1 fixed	2.252	0.559	–	–
3 fixed	2.505	0.727	2.504	0.735
3 sampled from 9	2.440	0.701	2.459	0.720
Motion-X++				
3 fixed	–	0.298	–	–
3 sampled from 9	–	0.273	1.805	**0.348**

Judging sessions. A model that appears in two tables can score slightly differently in each when the two tables were judged in separate sessions. Repeated sessions differ by at most 0.007 on HumanML3D and by about 0.02 on Motion-X++. The HumanML3D rows of Table 12 come from one judging session, in which the video-only structured model scores 2.459 as *3 sampled from 9, with*

azimuth, against 2.465 in Table 5. The Motion-X++ LLM-judge score of Motion-VLM is 0.348 in Table 12 and 0.349 elsewhere.

Training views and azimuths. On HumanML3D, three fixed cameras instead of one gain +0.167 [+0.147, +0.188] on the LLM judge, and sampling the three views from the nine-camera pool instead of fixing them loses 0.026 [0.010, 0.041]. On the same judge, the azimuth target gains +0.009 [−0.006, +0.023] with fixed cameras and +0.019 [+0.005, +0.034] with the sampled pool. Because the actor's initial facing is normalized, the azimuth is almost a constant function of the camera index (minimum modal share 0.9999), so it mainly tells the model which camera each view came from.

Test views. Both datasets use the same view sampling: three of nine cameras for training and three fixed frontal cameras for testing. The different effect of the motion tokens in Table 5 therefore does not come from the views. Moving the test input from rear to frontal cameras, without retraining, raises the LLM judge by 0.018 to 0.031 on six of the seven models we tried and by 0.006 on the seventh.

B.5 Intervals for the ablations

Table 13 gives the 95% interval of each change in Table 5, from a paired bootstrap over mirror-averaged motions on HumanML3D and over clips on Motion-X++. Another clip's motion also lowers the Motion-X++ scores significantly more than removing the motion does (−0.094 [−0.138, −0.051] on MotionJudge, −0.024 [−0.039, −0.008] on the LLM judge).

Table 13: **Intervals for Table 5.** Δ as in that table, with the 95% interval of the paired difference. Bold: the interval excludes zero.

Dataset	Change	MotionJudge	LLM judge
HumanML3D	structured target	**+0.045** [+0.014, +0.076]	**+0.020** [+0.004, +0.036]
	+ motion (Motion-VLM)	+0.018 [−0.014, +0.044]	−0.008 [−0.023, +0.006]
	motion removed	−0.009 [−0.029, +0.010]	−0.005 [−0.014, +0.003]
	motion zeroed	+0.017 [−0.003, +0.036]	+0.005 [−0.003, +0.013]
	another clip's motion	−0.004 [−0.019, +0.024]	+0.004 [−0.005, +0.012]
Motion-X++	structured target	−0.008 [−0.064, +0.040]	−0.003 [−0.025, +0.019]
	+ motion (Motion-VLM)	**+0.221** [+0.170, +0.281]	**+0.093** [+0.067, +0.112]
	motion removed	**−0.230** [−0.296, −0.182]	**−0.108** [−0.128, −0.082]
	motion zeroed	**−0.229** [−0.291, −0.178]	**−0.109** [−0.129, −0.084]
	another clip's motion	**−0.320** [−0.384, −0.262]	**−0.131** [−0.153, −0.104]

B.6 Effect on general video understanding

Table 14 measures what training costs on general video question answering. Both Motion-VLM models lose some accuracy against their zero-shot base: 1.5–2.0 points on MotionBench and 5.0–7.3 points on FAVOR-Bench. Since 95–100% of their answers follow the requested format, the losses are not a formatting artifact.

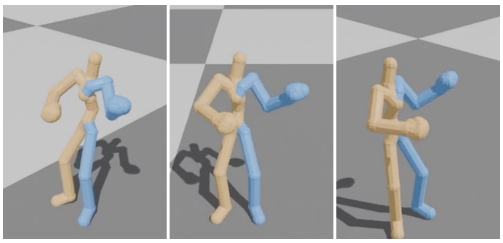
Table 14: **General video understanding after training.** Accuracy on MotionBench (dev, n = 4,018) and FAVOR-Bench (fixed 1,256-item subsample), with the same harness and greedy decoding for every row. Δ: change from the zero-shot base (shaded).

	MotionBench		FAVOR-Bench	
Model	Acc.	Δ	Acc.	Δ
Qwen3.5-4B (zero-shot base)	60.80	–	49.84	–
Motion-VLM trained on HumanML3D	58.79	−2.02	42.52	−7.32
Motion-VLM trained on Motion-X++	59.33	−1.47	44.82	−5.02

C Qualitative Results on Motion-X++

These examples show the captions behind the Motion-X++ scores of Motion-VLM in Table 3. Each example gives the middle frame of the three frontal views that Motion-VLM reads at test time, the reference caption, the output of Motion-VLM, and the reason given by the LLM judge. In the output, **green** marks content that matches the reference, blue marks detail that the reference does not state, and **red** marks an error. In the first four examples Motion-VLM states every key fact of the reference, including which hand does what and how the legs alternate, and adds detail about posture and stance that it reads from the renders.

Example 1: playing the viola.

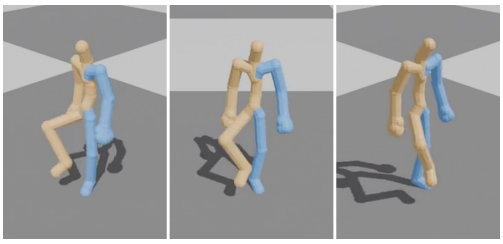


Reference. The person is playing the viola, moving the bow across the strings with her right arm while her left hand fingers the strings to produce different notes.

Motion-VLM. The person is **playing the viola, moving the bow across the strings with her right arm while her left hand fingers** press down on **the strings to create different notes**. Her movements are coordinated and precise, reflecting the technique required for playing the viola.

LLM judge. *The core action, the division of labor between the right and left hands, and the playing technique all match the reference; the added remark on coordination does not conflict with it.*

Example 2: kicking a shuttlecock.

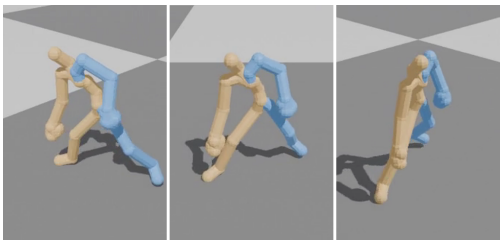


Reference. The person is performing a series of shuttlecock kicks, alternating between the left and right foot while maintaining balance and coordination.

Motion-VLM. The person is **performing a shuttlecock kicking motion**. They are standing with feet apart, **alternating kicking motions with each leg**, while keeping the upper body relatively stable and the head focused on the shuttlecock.

LLM judge. *It captures the shuttlecock kicks, the alternation between the left and right foot, and the balance; the description is complete and has no contradiction.*

Example 3: yoga triangle pose.

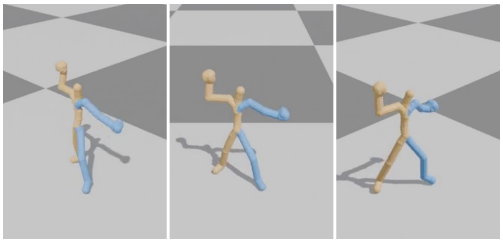


Reference. The person is performing a yoga sequence, specifically transitioning into and holding the triangle pose, involving lateral stretching of the torso and spine, engaging leg muscles, and extending the arms.

Motion-VLM. The person is performing **a yoga triangle pose**, maintaining a wide-legged stance with **arms extended** and **torso bent sideways**, creating a triangular shape with the body.

LLM judge. *It captures the triangle pose, the sideways bend of the torso, the extended arms, and the wide-legged stance, consistent with the reference.*

Example 4: a Shaolin sword form.

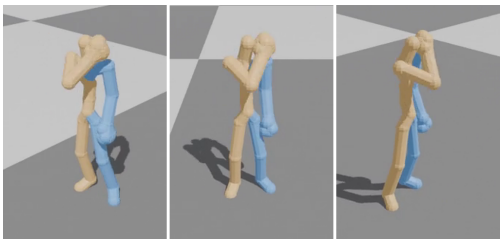


Reference. The person is performing a Shaolin Kung Fu Wushu sword form comprising a sequence of precise movements including stances, strikes, and blocks using a sword, accompanied by coordinated footwork and body positioning.

Motion-VLM. The person is performing **a sequence of movements from a Shaolin Kung Fu Wushu sword form**. They are demonstrating **precise, coordinated steps** and sword techniques, including thrusts, slices, and **stances**, with a focus on fluidity and control.

LLM judge. *It captures the core actions, their sequence, and the coordination of the sword form; the details agree with the reference.*

Failure case: taking off headphones.



Reference. The individual is standing and sequentially removing headphones with both hands, lifting them from the ears and bringing them down away from the head.

Motion-VLM. The person is **standing** and **repeatedly touching their eyes** with their hands. They start with their hands by their sides, **raise them** to **their eyes**, touch, and then **lower them back down**.

LLM judge. *The core action is wrong: the reference describes taking off headphones, while the output describes touching the eyes.*

The renders show only the body, so the headphones are invisible: in every view the hands rise to the head and come down again, which is also what touching the eyes looks like. Motion-VLM gets the body movement right but names the wrong purpose, a limit of body-only renders rather than of the viewpoints.

C.1 Contribution of the motion input

The two examples below show what motion adds on Motion-X++ (Table 5). Each compares Motion-VLM with the video-only structured model of Table 5, which has the same training and the structured target but no motion input, and adds what Motion-VLM writes when its motion is removed at test time. Both judges score Motion-VLM above the video-only model on 206 clips and below it on 61. In 111 of these 206 clips a musical instrument is played; the renders show only the body, so the instrument must be inferred from the arms and hands.

Example 5: cutting vegetables while standing.

Reference. The person is performing a simulated action of cutting vegetables while standing. He's making a chopping motion with his arms, moving them up and down rhythmically as if he were using a knife to cut through vegetables on a cutting board.

Motion-VLM (video and motion). The person is **standing** and performing **a repetitive motion of cutting vegetables**. They are using their hands to mimic the action of slicing, with **a slight downward movement**, and then returning to the starting position. The upper body remains relatively stable while the hands and forearms are engaged in the cutting motion.

Video only. The person is **standing** and performing **a sequence of hand gestures, likely as part of a presentation or explanation**. They are moving their hands and arms in various positions while maintaining an upright stance.

Motion-VLM, motion removed. The same caption as the video-only model, word for word.

LLM judge. *Motion-VLM: it captures the core actions of standing, simulated vegetable cutting, and repeated up-and-down arm movement, completely and without error. Video only: the core action is wrong, as the cutting is mistaken for the gestures of a speaker, and key details such as the rhythmic up-and-down movement are missing.*

Example 6: skipping rope while walking.

Reference. The person is performing a skipping rope exercise while walking forward. They are coordinating the swinging of the rope with their hands and jumping over it with their feet, all while taking steps in a forward motion. This action requires timing and dexterity to avoid tripping.

Motion-VLM (video and motion). The person is **performing a skipping rope exercise while walking forward**. They are **rotating the rope** with their wrists, **jumping over it** as it passes under their feet, and **taking steps in a forward direction**. The sequence involves **coordinating** the arm rotation with the leg jumps and steps.

Video only. The person is performing **a walking motion while simultaneously swinging their arms**. They take **steps forward** with alternating legs, and as one leg steps forward, the opposite arm swings forward in coordination. The arms return to a neutral position as the next step is taken with the other leg.

Motion-VLM, motion removed. The person is performing **a walking motion while simultaneously moving their arms in a circular motion, as if mimicking the action of grinding a spice**. . . .

LLM judge. *Motion-VLM: the core action, order, and direction all match the reference, and it conveys the coordination of skipping rope while walking forward. Video only: the core action is wrong; it describes ordinary walking with arm swings and misses the key action, skipping rope.*

In both clips the video-only model sees the arms move but not what they do. Motion-VLM names the action only when it reads the motion; without it, Motion-VLM falls back to a generic description.

D Construction and Validation of MotionADV

This appendix supports Section 3: it details how MotionADV is validated and composed, shows examples, and reports the minimal-pair probe cited in Section 3.3.

D.1 Label validation

Rules and language models generate candidate edits but do not label them. Qwen3.5-27B proposes the attribute edits by rewriting the action phrase that states the value, and temporal-order candidates swap two adjacent actions by rule. Each candidate goes to three validator models from two families: Qwen3.5-27B and Qwen3-VL-32B-Instruct from the Qwen family, and gemma-4-31B-it from the Gemma family. Without being shown our label, each validator independently states the original value of the edited action on the target attribute, then the edited value, and then whether any other attribute changed. Re-deriving the original value catches an edit filed under the wrong attribute. A validator asked only whether the direction flipped will confirm such an edit, whereas one asked which attribute "the figure turns to the right" states will reject it. A label passes on a unanimous vote or on a majority that spans both families. A majority of the two Qwen validators against gemma, where a shared blind spot could win, goes to GPT-5.5, an arbiter from a third family. GPT-5.5 decided the 353 escalations of the first round, accepting 269, and the 20 escalations of regenerated edits, accepting 14. Temporal-order candidates are kept only on a unanimous vote, without arbitration. The English set is a paired translation of the Chinese set by Qwen3.5-27B, checked by the same three validators on the English text; a pair whose check would need arbitration is dropped rather than arbitrated. Each sample is released with its votes and escalation record, and rejection rates are reported per attribute. Two limits remain. The base model of the evaluated judge is one of the three validators. Appendix F.5 reports MotionJudge on the labels that pass without its vote. The labels are also filtered by models, not by humans.

D.2 Edit size and composition

Editing one action keeps the length change of an error sample within the range of real predictions. References average 37 characters over 3.97 actions, so one action is about 9 characters, whereas deleting a whole caption changes length by −37. Across real prediction–reference pairs, the length difference has a 10th percentile of −12 and a 90th percentile of +18. A deletion of the whole caption, in contrast, can be caught from length alone. On natural output, 54.4% of the whole-caption judge's flags (71.3% on side) fall on predictions that say less rather than on predictions that say something false, and on an earlier version of the challenge set a condition that deleted the whole caption scored a spurious 1.000.

Table 15: **Composition of MotionADV.** Error samples per attribute and edit: *flip* changes one stated value, *drop* deletes it, and *add* inserts one. The lower rows give the totals, the controls (shaded), and the temporal-order pairs. –: the edit does not apply (*add*: the attribute is always stated already; rewrites and copies: Chinese only).

	Chinese			English		
Attribute	flip	drop	add	flip	drop	add
side	126	115	–	116	101	–
direction	97	68	72	86	61	53
motion type	124	14	–	86	13	–
bilateral mode	112	13	–	61	10	–
amplitude	117	118	–	113	116	–
speed	104	96	–	98	93	–
oscillation axis	83	39	–	48	30	–
rotation	57	38	–	29	13	–
articulation	145	47	–	113	42	–
count	63	10	–	42	10	–
total error samples	1,658 (435 clips)			1,334 (414 clips)		
length control: pass (short / long)	400 / 400			390 / 390		
length control: matched errors (flip / drop)	542 / 258			540 / 240		
rewrites and copies: paraphrase / identical copy	281 / 281			–		
temporal order: reordered / original	139 / 139			138 / 138		

Table 15 gives the composition of MotionADV by attribute and edit family. Drop edits of motion type and rotation, and of oscillation axis in English, are released but excluded from the detection rates of MotionJudge, because the verb itself carries these attributes and a fluent edit cannot delete them. Oscillation-axis flips that keep the old axis in the verb or leave the axis domain (25 in Chinese, 7 in English) are released but excluded from every judge's rates. The English set translates the same clips with the same labels and edit sites, so the two sets have the same content. A native-English set is left to future work. No combination of attribute and edit family reaches the 200-sample target, because the validated pools are smaller. Samples cluster (the 73 Chinese count samples fall in 59 clips), so intervals come from a bootstrap over clips. Two value domains are skewed: 56 of 73 count samples state a count of one, and amplitude is binary (141 small, 94 large). The noise floor, 0.024 per cell, is measured on the identical-copy samples of MotionADV.

D.3 Representative minimal pairs

The boxes below show English samples from MotionADV. Each edit changes one stated value (*flip*) or deletes it (*drop*). The original value is shown in **green** and the edited text in **red**. Every label passed the validator vote of Section 3.2. For the first example, all three re-derived the original value *right* and the edited value *left*, and confirmed that nothing else changed.

Side, flip (right → left)

Reference. The character raises their **right** hand to scratch their head while maintaining a standing posture.
Edited. The character raises their **left** hand to scratch their head while maintaining a standing posture.

Count, flip (3 → 5)

Reference. The character continues walking forward, moving both hands from their sides to in front of their abdomen **three times** during the movement.
Edited. . . . moving both hands from their sides to in front of their abdomen **five times** during the movement.

Speed, drop (slow → not stated)

Reference. The character **slowly** rises from a squat to a standing position, bringing the left hand from touching the ground back to the side of the body.
Edited. The character rises from a squat to a standing position, bringing the left hand from touching the ground back to the side of the body.

Table 16: **Similarity scores for the example pairs.** Red marks a failure: the caption with the wrong fact scores at least as high as the fact-preserving rewrite. B@k = BLEU-k, R-L = ROUGE-L.

		B@1↑	B@4↑	R-L↑	METEOR↑	BERTScore↑
Reference:	"The person raises the left hand."					
wrong fact	"The person raises the right hand."	.833	.537	.833	.807	.993
rewrite	"The *woman* raises the left hand."	.833	.537	.833	.807	.984
Reference:	"The person walks forward and stops."					
wrong fact	"The person walks backward and stops."	.833	.000	.833	.807	.982
rewrite	"As can be seen in the video, the person walks forward and then comes to a stop."	.294	.214	.435	.682	.935

Oscillation axis, flip (left–right → front–back)

Reference. . . . the upper body sways slightly **left and right** to the rhythm with minor vertical bouncing, conveying a state of listening to music.
Edited. . . . the upper body sways slightly **forward and backward** to the rhythm with minor vertical bouncing, conveying a state of listening to music.

Rotation, flip (counter-clockwise → clockwise)

Reference. . . . with legs rapidly circling and weaving beneath the body, rotating **counterclockwise** around the hands as the axis.
Edited. . . . with legs rapidly circling and weaving beneath the body, rotating **clockwise** around the hands as the axis.

Bilateral mode, flip (alternate → mirror)

Reference. . . . crouching and sneaking at a constant speed toward the left, **alternating leg steps** about nine times, with arms slightly spread to maintain balance.
Edited. . . . crouching and sneaking at a constant speed toward the left, **alternating leg steps in opposite directions** about nine times, with arms slightly spread to maintain balance.

D.4 Similarity metrics on minimal pairs

Table 16 scores the example pair of Figure 1 and a second pair with each similarity metric. A typical verified reference reads: "The character bends forward with the head lowered, dribbles the basketball in place rapidly with the left hand, then holds the ball with both hands." Each such caption makes several claims (body part, side, speed, object, order), any one of which a fluent caption can get wrong.

We also scored 14 minimal pairs of this kind with every similarity metric. In 10 pairs the wrong fact and the fact-preserving rewrite each alter one word (e.g., "left" → "right" against "person" → "individual"), and in 4 pairs the fact-preserving rewrite changes more of the sentence. Table 17 counts how often each metric ranks the wrong caption at least as high as the rewrite. The n-gram metrics mostly give both captions the same score, whereas BERTScore and an embedding similarity (Qwen3-0.6B) rank the wrong caption *higher* in almost every pair.

Table 17: **Minimal pairs: wrong fact versus fact-preserving rewrite** (14 pairs). Counts of pairs where the wrong caption scores higher (reversed), the same (tied), or lower (correct).

Pairs	Outcome	BLEU-1	BLEU-4	ROUGE-L	METEOR	BERTScore	Embedding
equal edits	reversed	0/10	2/10	0/10	2/10	8/10	9/10
(one word each)	tied	9/10	4/10	9/10	7/10	0/10	0/10
	correct	1/10	4/10	1/10	1/10	2/10	1/10
larger	reversed	4/4	2/4	4/4	4/4	4/4	4/4
rewrite	tied	0/4	1/4	0/4	0/4	0/4	0/4
	correct	0/4	1/4	0/4	0/4	0/4	0/4

D.5 Similarity metrics on the full MotionADV

Table 18 extends the probe to all of MotionADV. Every n-gram metric scores the single-fact error above a validated paraphrase of the reference in at least 1,258 of 1,261 Chinese pairs. When the fact-preserving sample instead carries an unrelated prepended sentence, the error again scores higher in at least 89% of pairs, except under exact-match METEOR with a short sentence (.875 in Chinese, .900 in English). Only when both captions carry the same sentence and differ in the edited value alone (*Matched*) do the metrics prefer the correct caption, and CIDEr still prefers the error in 237 of 800 Chinese and 227 of 780 English pairs. BERTScore has no per-sample scores on the full set, so it is compared only on the 14 pairs above.

Table 18: **Similarity metrics on the full MotionADV.** Pairwise accuracy: the fraction of pairs in which a metric scores a fact-preserving sample above a single-fact error sample with the same reference, ties counted as one half (chance 0.5). *Paraphrase*: a validated rewrite of the reference (Chinese only). *Length control*: the reference with an unrelated sentence prepended, short or long, against a plain error sample. *Matched*: the length control against its matched error, which carries the same sentence, so that only the edited value differs. 95% intervals from a bootstrap over clips lie within ±0.045 of every entry. Chinese is split into characters and English into words, punctuation dropped. ‡Exact-match METEOR.

	Chinese				English		
		Length control			Length control		
Metric	Paraphrase	short	long	Matched	short	long	Matched
BLEU-1	.001	.010	.000	1.000	.019	.000	.992
BLEU-4	.000	.046	.000	1.000	.105	.002	1.000
ROUGE-L	.000	.035	.000	1.000	.059	.002	1.000
METEOR‡	.000	.875	.024	.998	.900	.037	.996
CIDEr	.002	.006	.000	.704	.010	.000	.709
Pairs	1,261	1,240	1,240	800	589	589	780
Clips	280	270	270	270	254	254	254

D.6 Verified references and draft–correction pairs

In evaluation, we assume that a fact stated by a verified reference is correct and that a reference need not state every fact, so MotionJudge scores only the facts the reference states and abstains where it is silent. MotionADV does not rest on this assumption: an edit only needs to leave a plausible caption, and its label follows from the construction, so an *add* edit may insert a value that the source caption does not state. In the draft–correction pairs, the direction of each edit gives the label: a rewritten value is a wrong value, an added clause is an omission, and a deleted clause is an unsupported claim. These pairs test the judge and do not rank captioners. Their references are edits of one production captioner's outputs, and that captioner scores 0.955, against 0.064 for the best other system and 0.019 for an unrelated reference. This advantage cannot be separated from where the references came from.

E Specification of MotionFact

This appendix specifies the typed action schema of MotionFact (Section 4.1): its grammar, the verdicts MotionJudge returns, and how MotionADV, MotionJudge, and the training target of Motion-VLM share its slots.

E.1 Grammar, types, and verdicts

Grammar. A MotionFact description is generated by a finite, non-recursive production system:

$$\textit{desc} ::= \texttt{<units>}\ \textit{unit}^{+}\ \texttt{</units>}\ \texttt{<summary>}\ \textit{text}\ \texttt{</summary>}$$
$$\textit{unit} ::= \textit{action}\ \textit{slot}^{*}$$
$$\textit{slot} ::= \texttt{part=}w \mid \texttt{side=}w \mid \texttt{dir=}w \mid \texttt{rot=}w \mid \texttt{count=}w \mid \texttt{speed=}w \mid \texttt{span=}w$$

A unit (one action) cannot contain another unit, so every claim is about one action.

Types. Each slot j has a finite domain D_j, for example $D_{\text{side}} = \{\text{left}, \text{right}, \text{bilateral}\}$, extended with $\perp$ for a slot that is left open. Ill-typed strings are rejected at parsing, before any comparison, so a malformed output is never scored as a false physical claim.

Verdicts. For a claimed value q_j and the value r_j stated by the reference, MotionJudge returns $h_j(q_j, r_j) \in \{\text{supported}, \text{contradicted}, \text{not-mentioned}, \text{abstain}\}$: *abstain* when $r_j = \perp$ (the reference does not state the fact), *not-mentioned* when $q_j = \perp$ but $r_j \neq \perp$, and otherwise supported or contradicted. Verdicts on an action combine under a partial order in which *contradicted* dominates and *abstain* is neutral.

Overall score. An LLM grader receives both captions and the per-attribute verdicts and assigns a 0–5 score from a fixed rubric that weighs severity. A contradiction on action, direction, side, or motion type is most severe, and an omission is weighted by the importance of the omitted content. The grader may flag a verdict it disagrees with but must score the verdicts as given.

E.2 Slots shared by MotionADV, MotionJudge, and the target

Side, direction, rotation, and count are the slots s, d, r, c. Speed and amplitude share the graded slot v with disjoint domains and are edited as separate attributes. Motion type, oscillation axis, articulation, and bilateral mode refine the action slot a: motion type determines which of the other three are defined for an action. Temporal order is the order of the actions. MotionADV edits all eleven attributes (Table 19). On system outputs MotionJudge scores the attributes the reference states on both datasets: side, direction, rotation, speed, motion type, bilateral mode, oscillation axis, and articulation, plus amplitude and count, which enter only the overall score. Action is checked only on the real draft–correction pairs and temporal order only on MotionADV, because the references of the public datasets lack the action options and temporal-order pairs these checks need. The training target of Motion-VLM uses six flat slots (arm side, direction, rotation, count, speed, and amplitude), written `arm_side`, `dir`, `turn`, `reps`, `speed`, and `amp` (Appendix I.2), with values computed from the motion data (Section 5.2). Facing f is defined but has no MotionADV attribute, so no result uses it.

The action slot is refined because a flat schema lost facts that had no slot of their own: the LLM compressed them into neighboring slots or dropped them. Under a flat schema, the LLM returned a rotation value on 19 of 1,937 actions. The same loss occurred on four attributes, and each received its own slot.

Table 19: **The eleven attributes of MotionADV**, grouped into slots of the schema, refinements of its action slot, and temporal order. ✓: also scored by MotionJudge on system outputs.

Attribute	Values	Example edit	Scored
side	left, right, bilateral	left hand → right hand	✓
direction	front, back, left, right, up, down	forward → backward	✓
rotation	clockwise, counter-clockwise	clockwise → counter-clockwise	✓
speed	slow, moderate, fast	slowly → quickly	✓
count	integer	twice → three times	✓
amplitude	small, large	small → large steps	✓
motion type	static, linear, reciprocating, rotation, circular	swings → circles	✓
oscillation axis	front–back, left–right, up–down	back and forth → side to side	✓
articulation	flex, extend	bends → straightens the elbow	✓
bilateral mode	parallel, mirror, alternate, joint	together → alternately	✓
temporal order	order of actions	A then B → B then A	

F Protocol and Validation of MotionJudge

This appendix supports the design and validation of MotionJudge in Section 4 and its use as the primary metric in Section 6.

F.1 Inputs to the judges

Both judges read prose on both sides. A structured output or a structured reference is reduced to its summary before scoring, so the MotionFact actions of Motion-VLM are never scored. A structured target is about 1.6× the length of its summary, and scoring prose against it would charge a model for content that a summary cannot carry. On the LLM judge, this setting moves every fine-tuned model up by 0.010–0.029 and every zero-shot model down by 0.004–0.011. It leaves the published baselines unchanged, since their references are already prose.

F.2 Protocol steps

MotionJudge asks a fixed sequence of small questions (Figure 3). Table 20 lists the steps, and Appendix I.1 gives each prompt. Every question offers a "not stated" or "none of the above" answer. The alignment question (Step 2) is also asked with an action from another clip, for which the correct answer is "none of the above". Extracted values are compared with the reference values in code.

Table 20: **The questions MotionJudge asks**, in order. On system outputs a caption takes about 17 calls in total (Appendix F.4).

Step	Question	Answer	Asked
1	Split a caption into actions	list of verbatim spans	once per caption
2	Which candidate action matches this reference action?	one letter	per reference action
3	What value does this phrase state for an attribute?	one letter	per matched action and attribute
4	Is a quantity stated, and what number?	A/B/C, then a number	per matched action
5	Which of two actions happens first?	A/B/C	per pair of actions
6	Is this reference action covered? Quote it	yes + quote, then yes/no	per reference action
7	Overall score from the verdicts	0–5	once per caption

The *One verdict* and + *Rubric* judges of Table 1 are whole-caption judges on the same base model with their own prompts; each reads the reference and the candidate once. The first answers whether the candidate contains an error, and its answer is applied to every attribute of the sample: a flag on an error sample counts as a detection on the edited attribute and as a false alarm on each of the other attributes, so its pass rate is close to one minus its flag rate. The second names the wrong attributes, and only the named attributes count as flagged. On the paraphrases and identical copies, both pass a sample that they do not flag. Both whole-caption judges also score the drop edits of motion type and rotation, and the English oscillation-axis drop edits, which MotionJudge excludes.

F.3 Quotation-based coverage check

Asked directly whether each reference action is covered, the LLM tends to say yes. We therefore require it to quote the candidate span that expresses the action. The quotation must occur verbatim in the candidate and pass a pairwise check (Step 6, Appendix I.1). On a separate synthetic set, built with the same edit operations and judged by Qwen3-8B, the direct question detected 0.38 of dropped actions and 0.21 of added ones. The quoted version detected 0.99 of both and flagged 0.01–0.04 of identical copies, reorderings, and paraphrases. Most of the gain comes from rejected quotations: a lenient claim of coverage must produce a span that survives the pairwise check.

F.4 Computational cost

On system outputs, MotionJudge makes about 17 LLM calls per caption: one to split the caption into actions, about 15 per-attribute questions (alignment, value extraction, and coverage), and one for the overall score. Scoring the 2,286 Motion-X++ captions of two systems took 81,367 calls (including a second aggregation mode we do not report) and about 20M prompt and 1.5M generated tokens. With Qwen3.5-27B served by vLLM on one 80 GB GPU, this took 74 minutes, or about half an hour per system. On MotionADV, where all eleven attributes are checked, the count rises to about 25 calls per sample. The whole-caption LLM judge makes one call per caption. We therefore run MotionJudge

on the main comparisons and on the HumanML3D ablations, and use the LLM judge alone where many variants are compared and both judges rank systems alike.

F.5 Uncertainty of the MotionADV rates and labels without the judge's base model

Table 21 gives 95% intervals for the rates of MotionJudge in Table 1, from a bootstrap over clips. That run stored counts per attribute rather than per-sample verdicts, so the bootstrap places each attribute's misses at random among its scored facts. On the earlier run whose per-sample verdicts survive, the exact detection bounds lay within 0.01 of this placement. The exact pass-rate bounds lay up to 0.02 lower on the other attributes and 0.15 lower on rotation, because false alarms cluster in clips. The lowest lower bound of detection is .887 in Chinese (direction) and .828 in English (bilateral mode, 71 samples); for the pass rate it is .924 and .876 (both bilateral mode). The interval of the lowest detection rate, English bilateral mode at .901 [.828, .962], covers 0.90. Placing every miss in the clips with the most facts, the most clustered arrangement, lowers these bounds to .845 and .784 for detection and .827 and .794 for the pass rate. On the earlier run, this arrangement gave a bound at or below the exact one for every attribute.

Qwen3.5-27B is one of the three validators. For 1,060 of the 1,560 Chinese and 894 of the 1,252 English error samples that MotionJudge scores, the label passes on a unanimous vote of the other two validators alone, one from each family, on all three questions and, in English, on both translation checks. MotionJudge misses 59 Chinese and 52 English error samples in total. Even if every miss fell in this subset, its detection there would be .944 and .942 (Table 21). The subset covers the smallest share of bilateral mode (34 of 125 and 22 of 71) and motion type (41 of 121 and 31 of 83), the attributes on which the validators agree least. An earlier MotionJudge kept its per-sample verdicts on an earlier version of the challenge set. On the 1,435 of those error samples that the current challenge set keeps unchanged, detection was higher where both other validators approved the label than where at least one of them objected: .938 (95% interval .923–.953) against .893 (.863–.921). The labels that needed the base model's vote or the arbiter were harder for the judge, not easier.

Table 21: **Uncertainty of MotionJudge's rates in Table 1.** Detection and pass rate with 95% intervals from a bootstrap over clips (2,000 resamples; 20,000 for the last row). The run behind Table 1 stored counts per attribute, so each attribute's misses are placed at random among its scored facts (median bounds over 200 placements); temporal order and *rewrites and copies* stored per-sample verdicts and use the exact bootstrap. The last row places every miss in the clips with the most facts, the most clustered arrangement. *Without Qwen*: error samples whose label passes on a unanimous vote of gemma-4-31B and Qwen3-VL-32B alone, in English also on both translation checks (n), and the detection rate on them if every miss of the attribute fell among them. Temporal-order labels passed a unanimous vote of all three validators, so that row is exact.

	Chinese				English			
			Without Qwen				Without Qwen	
Attribute	Detection	Pass	n	Det. ≥	Detection	Pass	n	Det. ≥
side	.992 [.979,1.00]	.960 [.954,.965]	205	.990	.982 [.962,.996]	.972 [.967,.977]	183	.978
direction	.923 [.887,.955]	.938 [.930,.946]	168	.893	.939 [.904,.970]	.934 [.924,.944]	148	.919
motion type	.950 [.908,.984]	.934 [.928,.939]	41	.854	.928 [.868,.976]	.934 [.928,.940]	31	.806
bilateral mode	.936 [.891,.975]	.937 [.924,.949]	34	.765	.901 [.828,.962]	.894 [.876,.912]	22	.682
amplitude	.979 [.959,.996]	.976 [.964,.987]	201	.975	.974 [.952,.991]	.986 [.975,.995]	193	.969
speed	.974 [.949,.995]	.977 [.964,.989]	166	.970	.989 [.972,1.00]	.977 [.962,.990]	162	.988
oscillation axis	.969 [.931,1.00]	.956 [.940,.971]	77	.961	.951 [.875,1.00]	.915 [.892,.938]	33	.939
rotation	.965 [.909,1.00]	.993 [.985,1.00]	30	.933	.966 [.889,1.00]	.997 [.990,1.00]	16	.938
articulation	.948 [.915,.977]	.974 [.961,.985]	90	.889	.929 [.887,.967]	.968 [.953,.982]	70	.843
count	1.00 [1.00,1.00]	.963 [.941,.982]	48	1.00	.980 [.935,1.00]	.970 [.948,.990]	36	.972
temporal order	.991 [.973,1.00]	1.00 [1.00,1.00]	107	.991	.991 [.971,1.00]	1.00 [1.00,1.00]	116	.991
all but temporal order			1,060	.944			894	.942
rewrites and copies		.976 [.970,.980]						
lowest lower bound	.887	.924			.828	.876		
misses packed in the largest clips	.845	.827			.784	.794		

F.6 Attribution errors of the + *Rubric* judge

The + *Rubric* judge of Table 1 often sees that something changed but files it under a neighboring attribute. On an error sample it either names the edited attribute (a detection), names only other

attributes (a misattribution), or names none. We count these cases in its per-sample records for a 1,471-sample version of the Chinese set, taken before the final label cleanup. The judge misattributed 46 of 80 missed oscillation-axis edits, 28 of them to direction and most others to motion type. It misattributed 39 of 93 missed direction edits, mostly to side and motion type. On speed (4 misses) and count (0), no miss was misattributed. The typed slots of MotionFact prevent such misattribution: a value such as *front–back* can only fill the oscillation-axis slot.

The two samples below, from the same records and translated from Chinese, show the difference. In each, the + *Rubric* judge names a neighboring attribute and misses the edited one, while MotionJudge contradicts only the edited fact and supports the other facts of the caption.

Oscillation axis, flip (left–right → front–back)

Reference. The character holds a submachine gun with both hands in a slight crouch, swivels their head slightly **left and right**, and conducts a vigilant search.
Edited. . . . swivels their head slightly **forward and backward**, and conducts a vigilant search.

+ Rubric judge. Names *direction*; the oscillation axis is not named.
MotionJudge (head action). Oscillation axis: left–right vs. front–back, **contradicted**; amplitude: small vs. small, supported; motion type: reciprocating vs. reciprocating, supported. No other fact is contradicted.

Direction, flip (left → up)

Reference. The character bends down to dribble forward, quickly changes direction to the right, feints to the **left**, then turns right to jump and shoot.
Edited. . . . quickly changes direction to the right, feints **straight upward**, then turns right to jump and shoot.

+ Rubric judge. Names *side*; the direction is not named.
MotionJudge. Feint: direction left vs. up, **contradicted**. Dribble: direction forward vs. forward, supported. Change of direction: side right vs. right and direction right vs. right, supported. Turn: rotation clockwise vs. clockwise, supported.

F.7 Agreement with the whole-caption judge

On HumanML3D, the six published checkpoints, the zero-shot base, and Motion-VLM in Table 6 are ranked identically by MotionJudge and the LLM judge (Spearman 1.00); the summary-only model and MG-MotionLLM, 0.001 apart on MotionJudge, swap places. On Motion-X++ the ten systems of Table 7 give Spearman 0.72. Most of them score between the random-clip and neighboring-clip controls under both judges (Appendix F.10), where small differences reorder the ranking. With gemma-4-31B as the base of the LLM judge, the eight HumanML3D systems keep their order (Appendix F.8).

F.8 Replication with a second base model

Re-running MotionADV with gemma-4-31B as the base (Table 23) keeps the advantage of MotionJudge over the + *Rubric* judge where the fact must first be located: on direction, motion type, bilateral mode, and oscillation axis, MotionJudge detects +0.13 to +0.52 more in Chinese and +0.10 to +0.20 more in English. On the other six attributes the two judges are comparable, with the + *Rubric* judge at most 0.06 higher. The + *Rubric* judge reads the two captions side by side, an easier task when they differ in a single word, whereas MotionJudge first extracts each caption into typed slots. Under gemma, MotionJudge does not reach a detection rate of 0.90 and a pass rate of 0.89 on every attribute. Its bilateral-mode detection rate falls to 0.888 and 0.831, its oscillation-axis detection rate to 0.829 in English, and its motion-type pass rate to 0.833 and 0.788. Rebuilding the LLM judge on gemma for 13 HumanML3D systems and variants keeps the ranking (Spearman 0.99, one order flip in 78 pairs, between two systems that neither base can separate). On 281 fact-preserving paraphrases and 281 identical copies, MotionJudge with Qwen3.5-27B keeps a pass rate of 0.976 with no difference between the two kinds, so the remaining flags are noise.

We also run MotionJudge itself with gemma-4-31B on the five strongest HumanML3D systems of Table 2, reusing the reference facts. Gemma parses the candidates, answers every question, and assigns the overall score. Table 22 shows that Motion-VLM ranks first under both bases, and its margin over MG-MotionLLM grows from +0.057 to +0.100. Its margin over the summary-only model is +0.061

and +0.043, both significant. Motion-VLM also leads Motion-Agent and MotionGPT under gemma (+0.122 [+0.078, +0.166] and +0.207 [+0.159, +0.256]). The five systems keep their order except that the summary-only model, tied with MG-MotionLLM under Qwen3.5-27B, scores above it under gemma. Gemma scores every system lower, so absolute scores are comparable only under one base.

Table 22: **MotionJudge with two base models** on HumanML3D (mirror-averaged, n = 2,186 motions under gemma). Differences are taken over the motions both systems score, with paired bootstrap 95% intervals in brackets.

System	Qwen3.5-27B	gemma-4-31B
MotionGPT	2.303	2.047
Motion-Agent	2.375	2.132
MG-MotionLLM	2.421	2.154
Qwen3.5-4B, summary-only model	2.420	2.210
Motion-VLM	**2.483**	**2.253**
Motion-VLM − MG-MotionLLM	+0.057 [+0.019, +0.096]	+0.100 [+0.058, +0.141]
Motion-VLM − summary-only	+0.061 [+0.030, +0.093]	+0.043 [+0.009, +0.079]

F.9 Side and direction errors in the draft–correction pairs

We read every side and direction rewrite in Table 24 with its per-fact verdicts and the reason code of each (Table 25). A clip counts as a rewrite when the reference both drops and adds side (or direction) words relative to the draft, anywhere in the caption, so the label also covers clips where the artist replaced a whole action. Alignment answers "none of the above" when no draft action matches a reference action, and the fact becomes not-mentioned; its prompt says that a difference in side or direction does not block a match, and that an unsure case gets "none of the above". When the draft states a value that is the reference value of another fact in the same clip, a post-rule turns the contradiction into not-mentioned. The rule removes false alarms when a draft groups clauses differently from the reference.

In most of the rewrites that are not contradicted, the reference fact has no counterpart in the draft (19 of 38 for side, 46 of 61 for direction), and the action verdict contradicts 17 and 42 of these clips. The 22 misses have four causes. Alignment rejects a draft action that differs only in side or direction (13 clips). The post-rule reassigns a contradiction the value step (Step 3) had found (4). The value step reads a draft phrase that does not state the side (1). The reference parse has no usable direction fact for the edited clause (4): two edits are turns, which the parse files under rotation, where they are missed as well; one is a fall with no direction fact; and one reference value is too coarse, since "steps back to the right" is stored as *back*, so the draft's "steps back to the left" is supported. The four direction contradictions outside the edit come from compound directions. For example, a forward lean is stored as *down* in the reference and read as *forward* in the draft.

The overall score still penalizes these drafts. Side rewrites score 1.85 (53 of 54 at 2 or below) and direction rewrites 1.86 (72 of 76). The 94 drafts the artists left untouched are the reference scored against itself, and they score 5.00. All 22 misses score 2, and the stated reason names the wrong side or direction.

On unchanged attributes, the low *meets* rate comes from not-mentioned verdicts, not from contradictions. The 239 clips with unchanged side words hold 547 side facts. Of these, 331 are supported and 4 contradicted. Of the 100 facts that alignment leaves unmatched, 98 lie in a clause that the artists reworded or added. Because the label compares only the set of side words, six of these clips still flip a side in one clause; MotionJudge contradicts one of them and calls the others not-mentioned. Of the 104 facts for which the value step answers "not stated", 91 lie in a clause that the draft repeats word for word. In 61 of them the side carries over from the previous clause ("raises both arms, turns the palms up"), which the value step cannot see because it is shown only the aligned phrase. In the other 30 the phrase states the side itself ("stands with both legs apart"). The same holds on the 94 untouched pairs: MotionJudge supports 137 of 191 side facts and 88 of 122 direction facts, and contradicts 1 and 5.

The three samples below are translated from Chinese.

Table 23: **MotionADV with a second base model.** Detection / pass rates with gemma-4-31B in place of Qwen3.5-27B and the same prompts and code (compare Table 1). Bold: the higher detection rate of the two judges.

	Chinese, gemma-4-31B		English, gemma-4-31B	
Attribute	MotionJudge	+ Rubric	MotionJudge	+ Rubric
side	.946 / .983	**.967** / .944	**.977** / .986	.959 / .949
direction	**.910** / .926	.717 / .884	**.913** / .920	.795 / .914
motion type	**.983** / .833	.855 / .932	**.952** / .788	.848 / .930
bilateral mode	**.888** / .970	.632 / .998	**.831** / .943	.662 / .995
amplitude	**.996** / .998	.957 / .980	.965 / .998	**.974** / .989
speed	.964 / .987	**.985** / .998	.967 / .991	**.969** / .996
oscillation axis	**.969** / .981	.454 / .999	**.829** / .951	.634 / .994
rotation	**.982** / .995	.884 / .993	.966 / .991	**1.00** / .990
articulation	.911 / .993	**.969** / .996	.923 / .961	**.942** / .997
count	1.00 / .994	1.00 / .996	.980 / .983	**1.00** / .998

Table 24: **MotionJudge on real errors of a production captioner (494 Chinese pairs).** Labels come from the artists' corrections: a rewritten value is a wrong value, an added clause is an omission, and a deleted clause is an unsupported claim (counted only; *n* counts labeled clips). Verdicts are per clip and attribute, and every other *n* counts the labeled clips that MotionJudge scores on the attribute, the denominator of the rates beside it. *value*: a fact of the attribute is contradicted. *any*: not every fact of the attribute is supported. *detected*: a fact is not-mentioned and none is contradicted. *no-contra*: no fact is contradicted. *meets*: every fact is supported. Table 1 scores one edited fact per sample, so *any* and *no-contra* correspond to its detection and pass rates only approximately.

	Rewrites			Omissions		Deletions	Pass (unchanged)		
Attribute	*n*	value	any	*n*	detected	*n*	*n*	no-contra	meets
action	169	**.947**	.970	79	.190	88	145	.855	.821
side	54	.296	**1.00**	134	.978	21	239	.983	.456
direction	76	.197	.908	130	.823	70	117	.897	.513
speed	1	0/1	1/1	4	4/4	43	78	1.00	.872
amplitude	2	0/2	2/2	11	.909	46	74	1.00	.959
count	7	3/7	6/7	6	6/6	26	10	10/10	10/10

Table 25: **Causes behind the side and direction rewrites of Table 24.** Each clip is counted once, by its edited fact. The causes were assigned by one reader from the verdicts and the two captions.

	Side	Direction
Rewrites scored	54	76
contradicted on the edited fact	16	11
contradicted only on a compound direction outside the edit	0	4
action replaced, added, or deleted (not-mentioned is correct)	19	46
draft states no value for the action	2	1
borderline (action partly rewritten)	5	4
miss: alignment rejects the matching action	7	6
miss: contradiction reassigned by the post-rule	4	0
miss: value read from a phrase without the side	1	0
miss: reference parse has no usable fact for the edit	0	4
Contradicted among contradicted or missed rewrites	16/28	11/21
Mean overall score of the rewrites (untouched drafts: 5.00)	1.85	1.86
Overall score of each miss	2	2

Side, action replaced: not-mentioned is correct

Reference. The character operates the vehicle from the driver's seat, **holds the steering wheel with both hands**, the body sways violently, and finally the head tilts back.
Draft. The character keeps operating the vehicle from the driver's seat, **waves the left hand upward**, and

the body sways more.

MotionJudge. Side: alignment matches "holds the steering wheel with both hands" to no draft action, *not-mentioned*. Action: holding the wheel vs. waving, **contradicted**. Score 2: "The reference has 'holds the steering wheel with both hands' and the candidate 'waves the left hand upward', which changes the form of the action and the position of the hands [. . .]; the candidate also omits the final head tilt."

Side, alignment miss (left → right)

Reference. The character stands in place and raises the **left arm**.
Draft. The character stands in place, raises the **right arm**, with the fingers of the right hand slightly spread and pointing to the front of the abdomen, then lowers the arm back to standing.

MotionJudge. Side: alignment answers "none of the above" for "raises the left arm", although "raises the right arm" is one of the draft actions it is offered, so the fact is *not-mentioned*. This is a miss. Action: **contradicted**. Score 2: "The candidate describes the reference's 'raises the left arm' as 'raises the right arm', a serious conflict in a core attribute (left/right). [. . .] The added details do not make up for the wrong side."

Side, contradiction reassigned by the post-rule (right → left)

Reference. The character stands at ease, feet apart, **right foot forward**. The body turns to the **left**, the arms hang naturally, slightly away from the body [. . .]
Draft. The character stands at ease, feet apart, **left foot forward**. The head turns to the **right**, the arms hang naturally, slightly away from the body [. . .]

MotionJudge. "Right foot forward" is aligned with "left foot forward" and the value step reads *left*, a contradiction. The post-rule turns it into *not-mentioned*, because *left* is the reference value of "the body turns to the left". That fact matches no draft action and is also *not-mentioned*. Feet apart and arms hanging: both sides, supported. Score 2: "The reference has 'right foot forward' and 'the body turns to the left', the candidate 'left foot forward' and 'the head turns to the right'. The picture is reversed in left/right and in facing, an error in the core action."

F.10 Controls and data checks

Controls on Motion-X++. MotionJudge scores a reference against itself at 4.999, against a random clip's reference at 0.460 [0.425, 0.496], and against the neighboring clip's reference at 1.755 [1.685, 1.828]. Re-running the whole chain shifts the mean by 0.021. Clip identifiers are exercise names, so 27.7% of neighbors record the same exercise and score 3.447, and the others score 1.105. For the LLM judge the same two controls give 0.010 and 0.133 on 400 clips. On HumanML3D, identifiers are numeric and the two controls coincide (0.975 for MotionJudge, 0.04 for the LLM judge).

Same model family. Both judges use Qwen3.5-27B and Motion-VLM is built on Qwen3.5-4B, whereas the published baselines use T5, Gemma-2, GPT-2, and LLaMA backbones. With gemma-4-31B, from another family, as the base of MotionJudge, Motion-VLM still ranks first and its margin over MG-MotionLLM grows from +0.057 to +0.100 (Appendix F.8).

Train–test overlap. In HumanML3D, 11.9% of test captions appear verbatim in the training split. Motion-VLM and the published captioners are all trained on this split and evaluated on the same test split with the same protocol, so the overlap favors no system. We keep these captions for comparability with prior work. For the published systems we use the released checkpoints and cannot control how they treated the overlapping captions. HumanML3D and our Motion-X++ training data do not overlap.

Mirrored skeletons. Body side is meaningless if a retargeting pipeline has mirrored the skeleton, and a render does not show the mirroring. We detect it with the sign of the scalar triple product of the up vector, the shoulder vector, and the facing direction, which flips under reflection. On 400 clips per dataset the test is decidable for 341 Motion-X++ and 340 Fit3D clips, all with the expected sign, and a mirrored control passes none. In HumanML3D, 8.2% of clips (from HumanAct12) are stored mirrored. Their motion-derived side, direction, and rotation slots are left empty in the training targets, while both judges still read the side stated by their captions, which match the renders.

Motion at test time. Removing, zeroing, or swapping the motion tokens of the trained model changes neither judge significantly on HumanML3D and lowers both on Motion-X++ (Table 13). Each condition lowers per-clip LLM-judge scores on Motion-X++ 2.2–2.4 times as often as it raises them, so the model uses its motion tokens on the dataset with richer references.

G Extended Related Work

This section extends Section 2 with the work closest to Motion-VLM, to MotionJudge, and to our audit.

Motion models that read video. A few motion-language models read video together with motion (Chen et al., 2025; Jiang et al., 2024; Li et al., 2025a). Others improve the motion representation with language–motion pretraining, body-part or kinematics-aware encoders, and timestamped poses (Li et al., 2025b; Mo et al., 2025; Zhang et al., 2025a; Markhorst et al., 2026). Motion-VLM instead renders the motion data as multi-view video for a general video-language model, adds motion tokens that carry the video timestamps, and states MotionFact facts before its caption. Such facts have long been used to describe motion (Laban & Ullmann, 1971; Delmas et al., 2022; Yazdian et al., 2025; Liu et al., 2024). We use them as training targets.

Evaluating motion captions. Video benchmarks for fine-grained motion ask questions about the video, and FAVOR-Bench and NextMotionQA also score captions (Hong et al., 2025; Tu et al., 2026; Cao et al., 2026). Closest to MotionJudge, KPM-Bench extracts action, order, and direction from captions without an LLM (Lin et al., 2026), and MotionHalluc tests directional, attributional, and temporal hallucinations in comparisons of two motion videos (Guo et al., 2026). MotionJudge covers eleven attributes with a fixed rubric, checks each against the reference, and separates a wrong fact from an omitted fact and from a fact the reference does not state.

Audits of caption metrics and judges. Our audit differs from earlier audits of caption metrics and LLM judges (Kilickaya et al., 2017; Sai et al., 2021; Karpinska et al., 2022; Zeng et al., 2024; Tan et al., 2025) in three ways. It reports detection and pass rates separately for each motion attribute rather than one agreement score. Each edit changes exactly one fact and keeps the caption length close to that of real captions. The labels of the edits are themselves validated: unvalidated negatives can be solved from text alone (Hsieh et al., 2023), and an independent verifier judged 58% of 230 rule-generated action swaps to leave the described motion unchanged.

H Limitations and Outlook

MotionJudge has not been validated against human raters and cannot penalize claims that the reference does not cover. On real errors, it can score a wrong side or direction as not-mentioned rather than contradicted, mostly because alignment rejects an action that differs from the reference only in that value, and it reads each value only from the aligned phrase, so it misses a side carried over from the previous clause (Appendix F.9). The English set of MotionADV is a machine translation of the Chinese set, and on HumanML3D every system is scored against a single reference caption per motion.

Beyond evaluation, MotionFact and MotionJudge are already used in an industrial motion-data pipeline to clean, synthesize, and rewrite captions, including for text-to-motion training. The insensitivity we find in caption metrics also bears on text-to-motion evaluation. R-Precision and MM-Distance read the prompt through the kind of text encoder that places a wrong side almost on top of the correct one (Section 3.3), and evaluating a generated motion by captioning it and comparing the caption with the prompt through a similarity metric inherits that metric's insensitivity. Checking each fact of such a caption against the prompt with MotionJudge is a natural next step.

I Prompts

This appendix gives the prompts of MotionJudge and of the whole-caption LLM judge (Appendix F) and the input prompt of Motion-VLM (Appendix B.1).

I.1 MotionJudge and the whole-caption LLM judge

The boxes below give the prompt of each step in Table 20, translated from the Chinese originals, with fields filled at run time shown as `[field]`, followed by the prompt of the whole-caption LLM judge used in Section 6.

Step 1 · Split a caption into actions

Input Below is a description of a human motion. It has been verified by a person, and its content is taken as correct.
Description: "`[caption]`"

Task Split it into a **list of actions in narrative order**. Rules:
(1) One action = one claim of the form "which body part did what"; "A, then B" must become two items.
(2) Every item must be a **contiguous span copied verbatim** from the description; do not change, add, or reorder any word.
(3) Only split what the description states; do not infer or complete omitted subjects.
(4) Holding a posture (e.g., "keeps standing") also counts as an item.
(5) Pure temporal connectives ("then", "finally") and pure manner modifiers ("quickly", "widely") are not separate items; they stay with the action they modify.

Answer Only a JSON array, e.g. ["span 1", "span 2"].

Step 2 · Align a reference action with the candidate's actions

Input Below is a candidate description of a human motion, split into a list of actions.
Candidate actions:
A. `[action 1]` B. `[action 2]` ...
`[last]`. None of the above: the candidate has no corresponding action.
An action from the reference: "`[reference action]`"

Task Which item in the candidate list is **the same action** as this reference action? Synonyms and different levels of detail count as the same action; **differences in left/right, direction, speed, amplitude, or count do not matter here**, since they are compared separately later. Pair only if it really is the same action; if unsure, choose "None of the above".

Answer One letter.

Step 3 · Extract a value from the aligned phrase

Input Below is one action phrase from a description of a human motion.
Action phrase: "`[aligned phrase]`"

Task What does this phrase say its `[attribute]` is? `[criterion]`
`[options, one letter each]`
`[last]`. The phrase does not state its `[attribute]`.

Answer One letter.

This question is used for side, direction, speed, and amplitude. For motion type and bilateral mode, which need context, the same question is asked about the full description, restricted to the aligned action.

Step 4 · Count (two questions)

Input Below is a description of a human motion. It mentions "`[action]`".

(a) Answering only about **this action**: which case describes its **quantity**?
A. Explicit repetition: it is said to be done several times ("repeatedly", "three times in a row").
B. A quantity inherent to the action, not an emphasis on repetition ("takes one step back", "turns two circles").
C. No quantity is given.
Answer A, B, or C.

(b) What **number** does the description give for this action (steps, turns, swings, repetitions)? If a number is given, answer only the number ("two circles" → 2); otherwise answer "not stated".

Step 5 · Temporal order of two actions

Input Below is a description of a human motion: "[candidate]"

Task According to this description, which of the two actions happens first?
X: "[action a]" Y: "[action b]"
A. X first B. Y first C. The order cannot be told (or at least one action is not mentioned).

Answer A, B, or C.

Step 6 · Coverage with a quotation (two questions)

(a) The reference contains this action span: "[reference span]". The full candidate is: "[candidate]".
Does the candidate contain something that expresses this action? If yes, answer "yes" on the first line and on the second line **copy verbatim** the part of the candidate that expresses it; if no, answer "no".

(b) Span A (reference action): "[reference span]". Span B (quoted from the candidate): "[quote]".
Does span B express the action of span A? Different wording or synonyms count as yes; a different action, or a description of scene, emotion, or style, counts as no. Answer yes or no.

Check The quotation must also appear verbatim in the candidate, which is checked in code.

Step 7 · Overall score from the per-attribute verdicts

Input The reference, the candidate, and the per-attribute verdicts, taken as facts.

Task Give one score:
5 = a reader forms the same picture of the motion as from the reference (core action, direction, side, and motion type all match);
4 = differences that do not change what action happened (wording, extra detail where the reference is silent, an unstated peripheral attribute);
3 = a reader would get a **detail** wrong (speed, amplitude, or count), but the core action is right;
2 = a reader would get the **core action** wrong (action type, direction, side, or motion type), or a whole action is missing, or a misleading claim is added;
1 = only part of the actions correspond;
0 = a reader would be led to a completely different motion.
An extra claim where the reference is silent is scored 2 only if it asserts an attribute that changes the understanding of the action; otherwise 4.

Answer One integer from 0 to 5.

Whole-caption LLM judge (0–2), used in Section 6

Input The reference description (GT) and a model description (PRED) of one motion, both reduced to their summary (Appendix F.1).

Task You are a strict reviewer of motion descriptions. Judge whether PRED conveys the motion described by GT accurately and completely:
2 = accurate and complete: the core action, order, and direction/left–right relations all agree with GT, with no clear error or omission;
1 = partly correct: the main action is captured, but something is omitted, the order is wrong, a direction or side is wrong, or details absent from GT are added;
0 = wrong or unrelated: the core action is wrong, contradicts GT, or cannot be matched.

Answer Only JSON with two fields: "quality" (0, 1, or 2) and "reason" ([short reason]).

I.2 Input prompt of Motion-VLM

The box below shows the user turn of Motion-VLM; there is no system prompt. The role and the view description come before the media: attention is causal, so only text placed ahead of the visual tokens takes part in encoding them. The question comes last, next to the answer. Each view enters as a separate video labeled `Video i:`, following the multi-video convention of the Qwen chat template, with i counting the sampled views. The prompt states neither the azimuths nor the camera indices, which would reveal them: the model predicts the azimuths in the `<views>` block.

User turn of Motion-VLM (three views, with motion)

Role You are a motion analyst. You are given renderings of a motion-capture clip together with the underlying 3D pose stream, and you describe what the person does as an ordered list of motion units in a fixed structured format.

Views 3 synchronized camera views of the clip, rendered as an untextured simplified body: the LEFT limbs are blue, the RIGHT limbs and the torso are orange. Left and right always mean the person's own anatomical sides, never sides of the screen.

Video Video 1: [video] Video 2: [video] Video 3: [video]

Motion The raw motion-capture pose data for the same clip, one marker per sampled frame:
<|motion_start|><0.0 seconds><|motion_pad|><0.2 seconds><|motion_pad|>
... <t_T seconds><|motion_pad|><|motion_end|>

Task Question: first say where each camera stands relative to the person, then split the clip into motion units in time order, write one line per unit with the fields in the order shown, then summarise. A unit boundary falls where the body's configuration changes most between the ∼1.2 s before it and the ∼1.2 s after it, and no unit is shorter than about 1.5 s; a clip whose configuration never changes that much is one single unit. Output only the three blocks below and nothing else:

```
<views>
view i:  bearing=<front|front-left|left|back-left|
                  back|back-right|right|front-right> camera=<fixed|moving>
</views>
<units>
i [t0-t1s] arm_side=<L|R> dir=<front|back|left|right> turn=<CW|CCW>
reps=<int> speed=<slow|moderate|fast> amp=<small|medium|large>
</units>
<summary>one or two sentences describing the same motion the units describe</summary>
```

Fields What the fields mean. The values shown above are the only ones allowed:
`view i` the camera that produced Video *i* above, one line each, in that order
`bearing` where that camera stands relative to the way the person faces at the START of the clip: front means the camera is in front of the person and sees the face, back means it sees the back, left means it stands off the person's own left side; omit the key when you cannot tell which way the person faces
`camera` fixed if the camera holds one position for the whole clip, moving if it travels with the person; unlike every other field it is never omitted
`i` the unit's position in time order, starting at 1
`[t0-t1s]` the unit's start and end time, in seconds from the start of the clip
`arm_side` which of the person's own arms is the moving one (L is the person's own left arm); the arms are the only body part this format names
`dir` which way the whole body travels over the ground during the unit, relative to the way the person faces; omit it when the person does not travel
`turn` which way the body turns about its own vertical axis, seen from above; omit it unless the turn is clear and stays in one direction
`reps` how many times the unit's own motion repeats; write reps=1 for a motion done once
`speed` how fast the person travels over the ground during the unit
`amp` amplitude, how far the moving arm reaches away from the hips

Rules Write the fields on one line separated by single spaces. Every field except the unit number i and the view lines' camera is optional. Omit the whole key for any field you cannot determine; do not guess, do not write none, and do not leave an empty value. Write [t0-t1s] only when the unit boundaries are clear. A unit for which you can determine nothing is its number followed by nothing_determined, and nothing else on that line. Do not name the action or the object; the <summary> is where the action is described. Example, in which the second unit omits what is uncertain and the third determines nothing:

```
<views>
view 1:  bearing=front-left camera=fixed
view 2:  bearing=back camera=fixed
view 3:  bearing=front-right camera=moving
</views>
<units>
1 [0.0-2.0s] arm_side=R dir=front reps=1 speed=fast amp=medium
2 [2.0-3.8s] turn=CCW speed=moderate
3 [3.8-5.6s] nothing_determined
</units>
<summary>The person walks forward taking pictures, then turns around.</summary>
```