CAN VLMS JUDGE MOTION? FROM CAPABILITY DIAGNOSIS TO RELIABLE DEPLOYMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Mainstream evaluation metrics for text-driven human motion generation (T2M) compute global similarity scores, offering no fine-grained assessment of text-motion alignment. Vision-language models (VLMs), with their capacity for multimodal semantic understanding, are a natural candidate to fill this gap. However, 3D human motion involves spatial depth, skeletal topology, and temporal continuity that lie beyond typical 2D visual reasoning, leaving VLM reliability in this domain unverified—and deploying an unverified evaluator risks producing misleading conclusions in practice. To investigate this, we present a four-part study. (1) We construct FineMo (Fine-Grained Motion Semantic Alignment Benchmark), a diagnostic benchmark comprising 14,400 questions across six data-driven semantic dimensions, three alignment levels, and three progressive task types. (2) We evaluate six mainstream VLMs on FineMo, revealing three failure modes that persist consistently across all models and tasks: Depth Perception Failure, Topology Discrimination Failure, and Temporal Resolution Failure. (3) Tracing each failure, we identify two root causes—rendering-stage information loss and model-stage processing limitation—each verified through controlled experiments. (4) Building on this analysis, we propose an adaptive evaluation framework that diagnoses failure modes from text semantics and activates targeted interventions accordingly, achieving 57% overall accuracy—16 points above baseline and 9 above blanket intervention—without modifying the model. Together, these contributions provide a systematic characterization of VLM capability boundaries in motion evaluation and an evidence-based framework for their reliable deployment.

1 INTRODUCTION

Text-driven human motion generation (T2M) has found broad application in gaming, film production, and embodied intelligence. Yet as models such as MDM (Tevet et al., 2023), MotionDiffuse (Zhang et al., 2022), and MLD (Chen et al., 2023) continue to improve, evaluation has not kept up. The prevailing metrics, R-Precision and MM-Dist (Petrovich et al., 2022; Guo et al., 2022), map text and motion into a shared feature space and compute global similarity scores, with no means to tell *where* a generated motion goes wrong.

Vision-language models (VLMs) (OpenAI, 2026; Comanici et al., 2025), with their capacity for multimodal semantic understanding, are a natural candidate to fill this gap. But 3D human motion poses challenges that go beyond typical 2D visual reasoning: inferring spatial depth, distinguishing skeletal topology, and resolving fine-grained temporal dynamics. Whether VLMs can handle these reliably has not been tested, and practitioners who trust VLM judgments without such testing risk unknowingly accepting flawed generations or rejecting good ones.

To address this, we present a four-part study. (1) We construct FineMo (Fine-Grained Motion Semantic Alignment Benchmark), a diagnostic benchmark for VLM reliability in motion-semantic understanding. We draw 1,600 text-motion pairs from four established motion-capture datasets, covering six data-driven semantic dimensions—Action Type (AT), Body Part (BP), Temporal Structure (TS), Spatial Direction (SD), Motion Frequency (MF), and Motion Intensity (MI). Each pair is expanded across three alignment levels (positive, hard negative, easy negative) and evaluated through three progressive task types (judgment, ranking, error correction), yielding 14,400 questions in total (Figure 1).

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.

Table 1: Comparison of existing benchmarks for human motion evaluation.

Benchmark	Motion input	Evaluated aspects	Dim. induction	Neg. sample levels	Task progression	Scale
CAR (Fujiwara et al., 2024)	3D motion	Temporal order	Experience-based	1	Single-level	—
MotionBench (Hong et al., 2025)	Natural video	Motion perception	Experience-based	0	Single-level	∼8,000 questions
MBench (Lin et al., 2026)	Rendered motion	Physical quality	Experience-based	0	Single-level	450 prompts
T2MBench (Yang et al., 2026)	Rendered motion	Overall generation quality	Experience-based	0	Single-level	1,025 prompts
FineMo (ours)	Rendered motion	**Semantic alignment** (6 dims: AT, BP, SD, TS, MF, MI)	**Data-driven**	**2** (hard + easy)	**3-level progressive**	**14,400 questions**

(2) We evaluate six mainstream VLMs on FineMo. Reliability diverges along semantic dimensions in a consistent pattern: Action Type is reasonably reliable, while Spatial Direction, Motion Frequency and Motion Intensity are systematically low. On Motion Frequency, for instance, even the best model achieves only 28% ranking accuracy against a 17% chance level. A closer look at the weak dimensions reveals three failure modes that persist across all models and tasks: Depth Perception Failure, Topology Discrimination Failure, and Temporal Resolution Failure.

(3) We trace each failure mode along the VLM information flow to two compounding sources: rendering-stage information loss and model-stage processing limitation. At the rendering stage, single-view projection discards depth-axis information, skeleton-line rendering strips body surface cues, and uniform temporal sampling loses inter-frame motion detail. At the model stage, VLMs show limited ability to infer 3D spatial relations from 2D inputs, struggle to recover skeletal topology from patch-based encodings, and fail to localize key dynamics amid redundant frames. Controlled experiments with staged interventions at each source confirm these attributions.

(4) Building on this analysis, we build an adaptive evaluation framework that diagnoses failure modes from the evaluation text and activates targeted interventions accordingly, without modifying the model. On FineMo, this achieves 57% overall accuracy, 16 points above baseline and 9 above blanket intervention, showing that targeted routing outperforms indiscriminate stacking.

Our main contributions are as follows:

- **FineMo Benchmark.** A diagnostic benchmark for VLM reliability in motion-semantic understanding, with data-driven dimensions, graded negatives informed by a survey of real T2M model errors, and a three-task progression from perception to error localization.
- **Three Failure Modes.** Depth Perception Failure, Topology Discrimination Failure, and Temporal Resolution Failure, shown to persist across six models and three tasks.
- **Two-Stage Root-Cause Attribution.** Each failure mode is traced to rendering-stage information loss and model-stage processing limitation, verified through controlled experiments.
- **Adaptive Evaluation Framework.** A diagnosis-driven framework that routes each sample to its matched intervention, outperforming both single interventions and blanket stacking.

2 Related Work

2.1 Text-to-Motion Evaluation

Prevailing evaluation metrics for T2M generation—primarily R-Precision and MM-Dist (Guo et al., 2022)—map text and motion into a shared latent space and compute aggregate alignment scores. While widely adopted, they collapse fine-grained semantic correspondence into a single scalar, offering no diagnostic signal for localized mismatches in body topology, temporal dynamics, or spatial direction. Preliminary explorations into LLM/VLM-based motion assessment (Fu et al., 2024; Yu et al., 2024) focus on aggregate performance or general video understanding, lacking controlled counterfactual constructions and multi-task probing tailored to motion-specific semantics. FAVOR-Bench (Tu et al., 2025) is the closest recent work, evaluating fine-grained motion understanding on natural videos with both close- and open-ended tasks. FineMo instead targets rendered 3D motion, pairing ground-truth motion-capture sequences with counterfactual texts that simulate typical T2M errors, and shifts the focus from measuring overall capability to diagnosing why VLMs fail and adjusting the evaluation configuration accordingly.

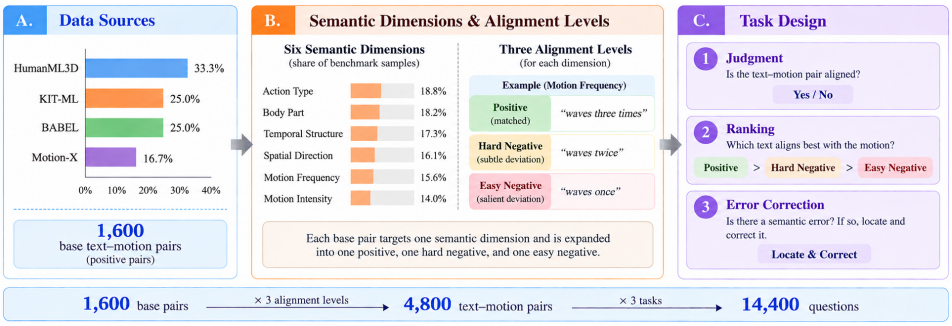

Figure 1: **Overview of the FineMo benchmark construction.** (A) Base text–motion pairs are collected from four T2M datasets. (B) Each pair targets one semantic dimension and is expanded into three alignment levels (positive, hard negative, easy negative). (C) All pairs are evaluated across three progressive tasks: judgment, ranking, and error correction.

2.2 Vision-Language Models for Motion Understanding

VLMs have demonstrated strong cross-modal reasoning in image and video domains (Liu et al., 2024a; Chen et al., 2024b), making them a natural candidate for fine-grained motion-text evaluation. However, their reliability degrades sharply beyond standard 2D perception: prior work reveals systematic weaknesses in 3D spatial inference (Chen et al., 2024a), skeletal topology recovery from patch-based encodings (Yuksekgonul et al., 2023), and key-dynamic localization amid redundant frames (Wang et al., 2025). Existing motion-language and video-VLM benchmarks (Fu et al., 2024; Yu et al., 2024) measure whether a model can answer, not where and why it fails across semantic dimensions.

3 FineMo Benchmark

Distinct from existing motion evaluation benchmarks, FineMo induces six semantic dimensions from data and probes each through three alignment levels and three progressive task types, yielding 14,400 questions (Table 1). Construction proceeds in three steps (Figure 1), followed by validity verification.

3.1 Semantic Dimension Induction

To avoid relying on subjective judgment about which semantic facets matter, we induce dimensions directly from data by their frequency in real motion descriptions. Following the TnT-LLM framework (Wan et al., 2024), an LLM decomposes the text annotations of four datasets—HumanML3D (Guo et al., 2022), KIT-ML (Plappert et al., 2016), BABEL (Punnakkal et al., 2021), and Motion-X (Lin et al., 2023)—into open-ended semantic attributes, which are then clustered and filtered by frequency (full pipeline in Appendix A.1). Dimensions appearing in more than 20% of all annotations are retained.

Six dimensions meet this threshold: Action Type (what action is performed), Body Part (which body part is involved), Temporal Structure (in what order sub-actions unfold), Spatial Direction (along which spatial axis), Motion Frequency (how many repetitions), and Motion Intensity (at what amplitude or force). Coverage ranges from 97% (Action Type) to 21% (Motion Intensity).

3.2 Sample Construction

Based on the six dimensions, we construct text-motion pairs from the four datasets, covering three alignment levels.

Positive samples. Positive samples are drawn with balanced coverage across the six dimensions, each pairing a text description with its ground-truth motion sequence.

Negative samples. We surveyed the outputs of three representative models—MDM (Tevet et al., 2023), MotionDiffuse (Zhang et al., 2022), and MLD (Chen et al., 2023)—and catalogued their

Table 2: **Per-Dimension Accuracy of Six VLMs across Three Tasks.** Judgment chance = 50%; Ranking chance = 17%. The difficulty ordering across dimensions remains consistent across all models and tasks, with **Action type** being the easiest and *Motion Frequency* the hardest.

Task	Dimension	Model						Avg.
		GPT-5.4	Gemini 2.5 Pro	Qwen3.5-Plus	Qwen2.5-VL	InternVL2	LLaVA-1.6	
Judgment	**Action type**	**84%**	**82%**	**78%**	**77%**	**73%**	**62%**	**76%**
	Temporal Structure	77%	75%	74%	75%	65%	53%	70%
	Body part	75%	73%	68%	70%	63%	52%	67%
	Spatial Direction	63%	66%	57%	58%	52%	46%	57%
	Motion Intensity	56%	54%	52%	51%	49%	45%	51%
	Motion Frequency	*53%*	*52%*	*50%*	*49%*	*48%*	*44%*	*49%*
Ranking	**Action type**	**75%**	**72%**	**67%**	**66%**	**63%**	**52%**	**66%**
	Temporal Structure	61%	60%	58%	59%	51%	40%	55%
	Body part	59%	57%	51%	53%	48%	38%	51%
	Spatial Direction	39%	41%	31%	33%	28%	22%	32%
	Motion Intensity	33%	30%	27%	25%	24%	19%	26%
	Motion Frequency	*28%*	*26%*	*23%*	*22%*	*21%*	*18%*	*23%*
Error Correction	**Action type**	**58%**	**56%**	**51%**	**49%**	**44%**	**33%**	**49%**
	Temporal Structure	47%	46%	44%	45%	36%	24%	40%
	Body part	44%	42%	37%	39%	33%	22%	36%
	Spatial Direction	28%	31%	22%	23%	18%	12%	22%
	Motion Intensity	23%	21%	19%	18%	16%	11%	18%
	Motion Frequency	*19%*	*17%*	*14%*	*14%*	*13%*	*9%*	*14%*

most common error types per dimension (e.g., generating a semantically similar action instead of the target, activating an adjacent limb instead of the correct one). Based on this survey, we construct counterfactual negative texts by modifying only the target dimension's semantic attribute while holding all others constant. Each dimension yields two negatives per positive sample, graded by semantic distance: a *hard* negative applies a semantically close substitution (e.g., jogs $\rightarrow$ walks), and an *easy* negative applies a semantically distant one (e.g., jogs $\rightarrow$ crawls) (Appendix A.2). All negative texts are LLM-generated and manually verified for dimensional precision and naturalness, yielding 1,600 groups and 4,800 text–motion pairs.

3.3 Task Design

Based on the 4,800 text-motion pairs, we design three task types that progressively test perception, discrimination, and localization, each instantiated 4,800 times, yielding 14,400 test questions.

The **judgment task** asks the VLM whether a given text and motion are semantically consistent (Yes/No), probing basic perceptual ability.

The **ranking task** asks the VLM to rank the positive, hard negative, and easy negative texts for the same motion by alignment, probing fine-grained sensitivity to the target dimension. Each group is presented under three cyclic label rotations to control for position bias.

The **error-correction task** asks the VLM to either localize the mismatch between a given text and the motion on the target dimension or state that none exists, probing actionable error localization.

Ground truth for all three tasks is derived directly from the construction process: judgment GT is a binary label; ranking GT is the order positive $>$ hard negative $>$ easy negative; error-correction GT is the semantic change recorded during rewriting, or "no error" for positives, evaluated by semantic equivalence rather than literal match.

3.4 Validity Verification

Since both sample construction and ground-truth generation rely on automated pipelines, we verify both through human annotation on a 300-group subset. Inter-annotator agreement reaches Fleiss' $\kappa = 0.81$; human–GT agreement ranges from 89% to 95% on judgment and from 82% to 91% on error correction, with Kendall's τ of 0.72–0.86 on ranking. Full per-dimension breakdowns are reported in Appendix A.3.

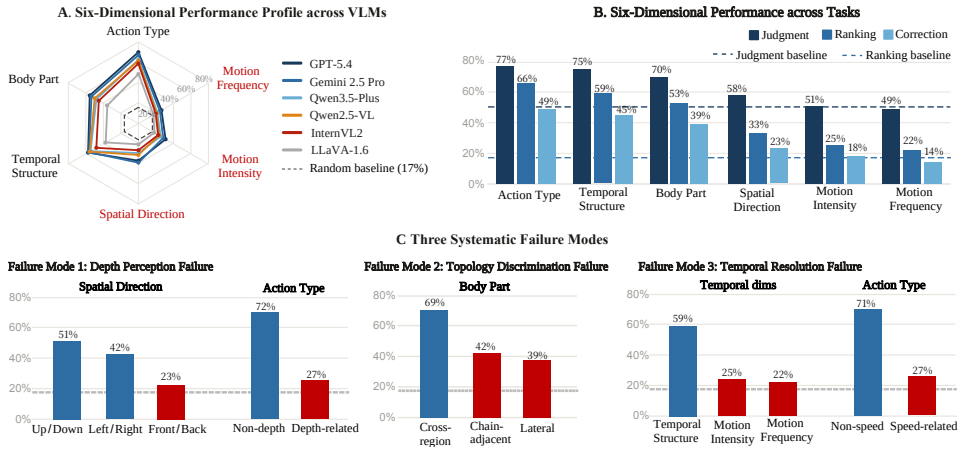

Figure 2: **Capability Profile and Failure Mode Analysis.** (A) Six-dimensional performance across six VLMs on ranking tasks; (B) per dimension performance across three tasks (Qwen2.5-VL); (C) subcategory breakdowns revealing three failure modes.

4 Capability Boundary Analysis

Experimental Setup. We evaluate six representative VLMs—GPT-5.4 (OpenAI, 2026), Gemini 2.5 Pro (Comanici et al., 2025), Qwen3.5-Plus (Qwen Team, 2026), Qwen2.5-VL (Bai et al., 2025), LLaVA-1.6 (Liu et al., 2024b), and InternVL2 (Chen et al., 2024b)—covering open- and closed-source systems across varying scales and temporal modeling capacities. All models share a unified input configuration: each motion is rendered as a skeleton-line animation from a single front view at 15 FPS, and a standardized output format (Appendix B.1–B.2). The judgment task reports accuracy; the ranking task reports exact-match accuracy, averaged over all three label rotations; the error-correction task reports semantic match rate over all 4,800 pairs, including positives whose correct response is "no error", where equivalence is judged by meaning rather than wording (Appendix B.3).

4.1 Consistent Divergence Across Models and Tasks

As shown in Figure 2A–B, reliability varies sharply across dimensions in a consistent pattern: Action Type is strongest, Body Part and Temporal Structure are intermediate, and Spatial Direction, Motion Frequency, and Motion Intensity are systematically lowest. Overall performance declines from judgment to ranking to error correction, as expected given increasing task difficulty, but the relative ordering of dimensions holds across all three tasks, pointing to genuine capability gaps rather than task-specific artifacts. Full results are in Table 2. We next look inside the weak dimensions to identify where the failures concentrate, and find three distinct failure modes. (see Figure 2C)

4.2 Three Failure Modes

Failure Mode 1: Depth Perception Failure. Within Spatial Direction, front/back accuracy (23%) is far below up/down (51%) and left/right (42%)—the only subcategory that requires depth reasoning (Figure 2C, left). Within Action Type, depth-dependent actions such as leaning forward or tilting backward score 27%, far below depth-independent actions such as raising an arm or kicking a leg (72%). The same weakness surfaces independently in two dimensions, pointing to a consistent failure on any assessment that requires depth information.

Failure Mode 2: Topology Discrimination Failure. Within Body Part, the finer the topological distinction required, the worse the accuracy (Figure 2C, middle): cross-region pairs (e.g., hand vs. foot) score 69%, chain-adjacent pairs within the same skeletal chain (e.g., upper arm vs. forearm) 42%, and lateral pairs (e.g., left vs. right hand) 39%. Cross-region assessment requires only coarse upper/lower body distinction; chain-adjacent assessment requires localization within a limb; lateral assessment requires telling which side of the body a part belongs to. Performance tracks this hierarchy directly.

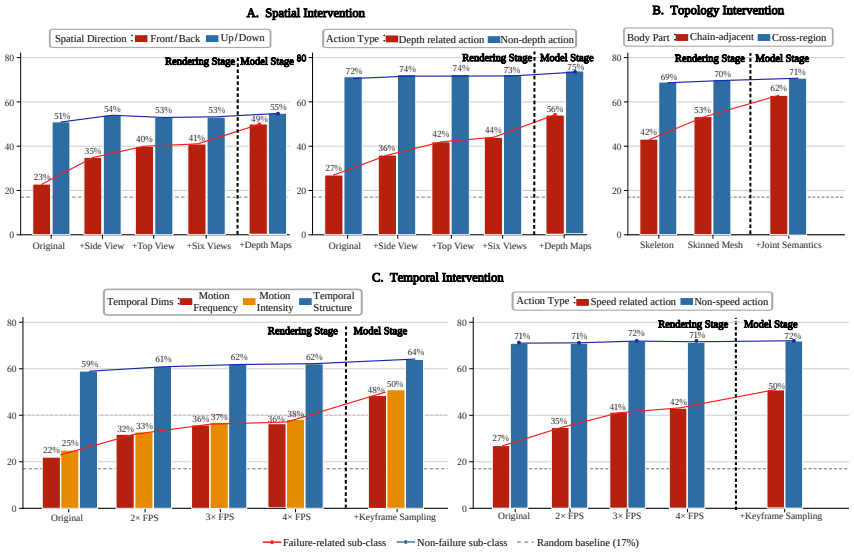

Figure 3: **Root-Cause Tracing via Targeted Interventions.** Progressive rendering-stage and model-stage interventions for each failure mode. Failure-related subcategories improve selectively under the corresponding intervention, while non-failure subcategories remain stable, confirming two-stage attributions.

Failure Mode 3: Temporal Resolution Failure. Among the three temporally dependent dimensions, Temporal Structure is relatively stable (59%), while Motion Frequency (22%) and Motion Intensity (25%) approach chance level (17%) (Figure 2C, right). The difference lies in temporal resolution demands: identifying the order of discrete events requires only coarse temporal localization, whereas counting motion cycles, measuring displacement magnitude, or detecting inter-frame velocity changes all require fine-grained temporal resolution. The speed subcategory within Action Type—walking vs. fast walking vs. jogging—scores 27%, well below non-speed actions (71%), providing converging evidence for the same limitation.

All three failure modes persist across models and tasks. Even Qwen2.5-VL, whose architecture includes dedicated temporal modeling components, still exhibits Failure Mode 3 under our frame-sequence input—pointing to systematic root causes at both the rendering and model stages, which Section 5 traces in detail.

5 Root-Cause Analysis

The three failure modes identified in Section 4 persist across all six models and all three tasks, suggesting systematic rather than model-specific causes. In this section, we trace each failure along the VLM information flow and identify two contributing stages: *rendering-stage information loss* and *model-stage processing limitation*. We disentangle the two through staged interventions: we first test whether enriching the rendering input (more views, richer body representation, or higher frame rate) can close the performance gap. If accuracy improves but eventually plateaus—while control subcategories remain flat—we attribute that portion to rendering-stage loss. We then apply a model-stage intervention (explicit depth cues, skeletal annotations, or keyframe selection) on top of the saturated rendering configuration. If this breaks through the plateau, again selectively, the model-stage attribution is confirmed.

All intervention experiments below use the ranking task. Qwen2.5-VL serves as the primary model; we replicate the full intervention ladder on InternVL2 to verify that the findings are not architecture-specific. Cross-model results are reported in Appendix C.4.

5.1 Depth Perception Failure

Rendering stage: depth-axis information loss. A single front-view orthographic projection collapses the 3D skeleton onto a 2D plane, discarding the depth axis entirely (Hartley & Zisserman, 2003). This yields a prediction: additional viewpoints should improve only front/back and depth-related action accuracy, with other subcategories unaffected.

We progressively add a side view and a top-down view, and finally replace these with six views. The results match the prediction: front/back and depth-related action accuracy rise substantially from their baseline values (23%→41% and 27%→44%, respectively), while all control subcategories shift by at most 3 percentage points (Figure 3A, rendering stage). Moving to six views yields no further gain—the rendering-stage benefit has saturated.

Model stage: 2D-to-3D spatial inference limitation. With rendering-stage information recovery exhausted, the remaining gap reflects the model's own ability to reconstruct 3D spatial relations from 2D projections. Prior work confirms that this capacity is weak even on natural images (Chen et al., 2024a; Liu et al., 2023); skeleton renderings, which lack perspective cues, make it worse. Providing the model with explicit depth information should therefore selectively lift the two depth-dependent subcategories further.

We augment the input with per-frame depth maps computed from the ground-truth 3D joint coordinates (Appendix C.1). Starting from the saturated rendering configuration, depth maps produce an additional 8 and 12 percentage points on front/back accuracy and depth-related actions, respectively, while control subcategories change by at most 2 percentage points (Figure 3A, model stage; InternVL2 replicates, Table 12).

5.2 Topology Discrimination Failure

Rendering stage: body surface cue loss. Skeleton-line rendering strips the surface cues that signal limb identity (Loper et al., 2015). Replacing it with a skinned mesh should therefore improve the fine-grained subcategories (lateral and chain-adjacent) while leaving cross-region unaffected.

Switching to skinned mesh rendering confirms this: lateral and chain-adjacent accuracy rise by 13 and 11 percentage points, respectively, while cross-region accuracy shifts by only 1 percentage point (Figure 3B, rendering stage). Both fine-grained subcategories, however, still fall well short of cross-region performance, indicating that rendering-stage recovery alone cannot close the gap.

Model stage: skeletal topology modeling limitation. VLMs encode images as independent fixed-size patches (Dosovitskiy et al., 2021), scattering the global relational structure of skeletal topology—left versus right, ordering along a kinematic chain—across discontinuous tokens. Prior work shows that such relational information is readily dominated by more salient local features (Yuksekgonul et al., 2023). To test whether this is the residual bottleneck, we convert the topological structure into a locally readable signal: each body segment is rendered in a distinct colour with a colour-to-part-name legend (Appendix C.2).

Starting from the skinned-mesh configuration, this annotation yields an additional 8 and 9 percentage points on lateral and chain-adjacent accuracy, respectively, while cross-region accuracy is unchanged (Figure 3B, model stage; InternVL2 replicates, Table 13).

5.3 Temporal Resolution Failure

Rendering stage: inter-frame motion detail loss. Uniform frame sampling at a fixed interval discards the continuous motion signal between snapshots (Hong et al., 2025). The three high-temporal-resolution subcategories are sensitive to this loss; Temporal Structure, which requires only coarse event ordering, is not. Increasing frame rate should therefore selectively improve the former group.

Progressively raising the frame rate to $2\times$, $3\times$, and $4\times$ the baseline confirms this: the three failure-related subcategories improve substantially, while Temporal Structure shifts by only 3 percentage points (Figure 3C, rendering stage). Between $3\times$ and $4\times$, the failure-related subcategories gain at most 1 percentage point—the rendering-stage benefit has saturated.

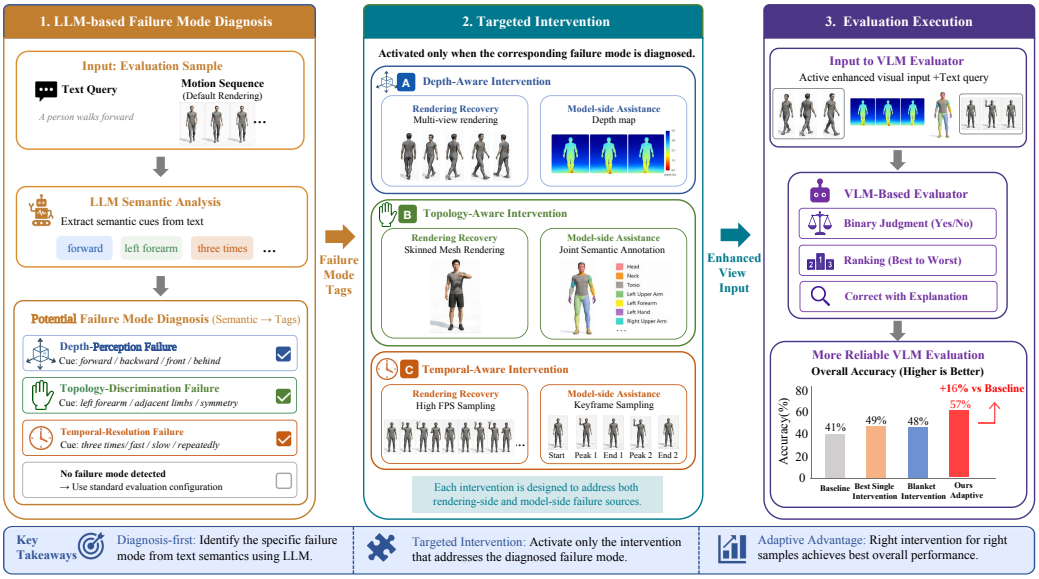

Figure 4: **Adaptive Evaluation Framework.** The framework diagnoses failure modes from text semantics (left), activates targeted interventions (middle), and executes VLM evaluation under the enhanced configuration (right).

Model stage: key-dynamic localization limitation. Once frame density is high enough, the bottleneck shifts to the model's ability to pick out informative frames from near-identical ones. Prior work on long-video reasoning shows that such redundancy overloads attention and dilutes key-frame signals (Wang et al., 2025). We replace uniform sampling with task-specific keyframe extraction computed from the 3D skeleton data, selecting only the frames most informative for each subcategory's criterion (Appendix C.3).

Starting from the saturated $4\times$ configuration, keyframe extraction produces an additional 12 percentage points on Motion Frequency, 12 on Motion Intensity, and 8 on the speed subcategory, while Temporal Structure and non-speed actions remain stable (Figure 3C, model stage; InternVL2 replicates, Table 14).

6 Adaptive Evaluation Framework

Section 5 shows that each intervention is effective for its target failure mode but has little effect elsewhere, so the key to deployment is routing each evaluation sample to the right one. We propose an adaptive framework that diagnoses the likely failure mode from the evaluation text and activates only the matched intervention, without modifying the model.

6.1 Framework Design

The framework consists of three sequential modules (Figure 4).

Failure mode diagnosis. Given an evaluation text, an LLM (Zheng et al., 2023; Ong et al., 2024) analyzes its semantic content and classifies it into one of four categories: depth-related, topology-related, temporal-resolution-related, or none (Appendix D.2).

Targeted intervention. Each identified failure mode activates the corresponding intervention from Section 5, combining rendering-stage recovery with model-stage assistance (depth-aware, topology-aware, or temporal-aware). Texts classified as none proceed under the standard configuration.

Evaluation execution. Assessment is performed under the activated intervention configuration.

Table 3: **Comparison of Intervention Strategies across Sample Categories.** Single interventions improve their target category selectively; blanket intervention introduces inter-intervention interference; the adaptive framework achieves the best overall accuracy (57%) by routing each sample to its diagnosis-matched intervention.

Configuration	Depth-Related Samples	Topology-Related Samples	Temporal-Resolution Samples	Non-Failure-Mode Samples	Overall Accuracy
Baseline	25%	40%	25%	60%	41%
Depth-Aware Intervention	**53%**	42%	25%	**64%**	48%
Topology-Aware Intervention	25%	**61%**	24%	61%	45%
Temporal-Aware Intervention	26%	41%	**50%**	63%	49%
Blanket Intervention	45%	53%	37%	55%	48%
Adaptive Framework (Ours)	**53%**	**62%**	**50%**	61%	**57%**

6.2 Experimental Validation

We compare five conditions on Qwen2.5-VL: a standard-input baseline, three single interventions (each applied to *all* samples), blanket intervention (all three applied simultaneously to all samples), and our adaptive framework. Failure-mode samples are defined by subcategory membership: depth-related samples cover front/back direction and depth-dependent actions; topology-related samples cover lateral and chain-adjacent body parts; temporal-resolution samples cover Motion Frequency, Motion Intensity, and speed-related actions; the remainder are non-failure-mode samples. The input specifications for each configuration—including view count, rendering type, frame sampling strategy, and total input volume—are reported in Appendix D.1.

Results are summarized in Table 3. **Single interventions** each produce large gains on their target category (+21–28 percentage points) but have negligible or inconsistent effects on non-target samples, confirming that the strategies are scope-limited. **Blanket intervention**, applying all three simultaneously, does not accumulate these gains: accuracy on each failure-mode category falls below the corresponding single intervention, and non-failure-mode accuracy drops by 5 percentage points. **The adaptive framework** avoids this by routing each sample to only its matched intervention. It matches the single-intervention ceiling for each failure-mode category (depth 53%, topology 62%, temporal 50%) while keeping non-failure-mode accuracy at the baseline level (61%), yielding 57% overall—16 percentage points above baseline and 9 above blanket intervention.

Since the framework routes each sample through an LLM-based diagnosis module, we also evaluate routing accuracy directly: on the 1,600 FineMo groups, the module achieves 93% overall accuracy with per-class F1 of at least 0.92 (Appendix D.3–D.4).

7 Conclusion and Limitations

Conclusion. The central question this work poses—whether VLMs can be trusted as fine-grained motion evaluators—turns out to have a structured rather than binary answer: reliability is not uniformly absent, but bounded by identifiable failure modes with traceable causes. FineMo, the three failure modes, the root-cause analysis, and the adaptive framework together provide both a diagnostic methodology and a deployment strategy, both extensible as understanding of VLM capability boundaries grows. We hope this work reframes the question from "can VLMs evaluate motion" to "under what conditions and with what configurations"—a framing more productive for both benchmark design and reliable deployment.

Limitations. FineMo's scope has four natural boundaries. (1) Evaluation uses rendered ground-truth motion paired with counterfactual texts; transfer to motions actually generated by T2M models and to natural video remains untested. (2) Root-cause attributions rest on intervention evidence rather than causal proof; underlying computational mechanisms warrant further investigation. (3) The adaptive framework's robustness depends on LLM-based failure mode diagnosis, which should be independently verified under out-of-distribution text semantics; the current routing module supports single-label diagnosis only, with multi-label extension left for future work. (4) The six dimensions were selected by a >20% frequency threshold, ensuring representativeness but not exhaustive coverage of motion semantics.

AI USE STATEMENT

In this work, we used generative AI tools as core components of the proposed methodology, in the following tasks with required disclosure. (1) *Qualitative and thematic data analysis*: an LLM decomposes motion descriptions into open-ended semantic attributes as part of the TnT-LLM-based dimension induction (Appendix A.1); the resulting clusters were inspected and semantically overlapping clusters were merged manually by the authors. (2) *Generation of synthetic data*: an LLM generates the counterfactual negative texts of FineMo (Section 3, Appendix A.2); every generated text was manually reviewed for dimensional precision and naturalness, and a 300-group subset was further validated by three human annotators (Appendix A.3). (3) *Implementation of methods*: an LLM performs failure-mode diagnosis within the proposed adaptive evaluation framework (Section 6, Appendix D.2), and an LLM judge assesses semantic equivalence in the error-correction task (Appendix B.3); we evaluated the diagnosis module against construction-derived labels (Appendix D.3) and validated the judge against 200 human-annotated pairs. All prompts used for these components are provided in full in the appendix. Tasks involving mathematical claims or proofs are not applicable to this work. Additionally, we used generative AI tools to edit the paper for readability and consistency. We have reviewed all AI-assisted work, and we take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work constructs a benchmark from publicly available motion datasets (HumanML3D, KIT-ML, BABEL, and Motion-X), used in accordance with their respective licenses, and evaluates publicly available VLMs. It involves no sensitive personal data and no new collection of human motion data. Human participation was limited to three annotators who validated benchmark quality; they were compensated at a standard research assistant rate. We see the primary societal value of this work as enabling more reliable VLM-based evaluation of text-driven motion generation, and as helping prevent the deployment of unreliable evaluators that could otherwise produce systematically misleading conclusions.

REPRODUCIBILITY STATEMENT

We have made the following efforts to support reproducibility. The benchmark construction pipeline, including the dimension-induction process, negative-sample rewriting strategies, and all LLM prompts, is described in Appendix A.1–A.2. Model versions, the unified input configuration, inference settings, evaluation metrics, and all task-specific prompts are given in Appendix B.1–B.3. The implementation of every intervention, including the complete definitions of the keyframe-extraction algorithms, is given in Appendix C.1–C.3, and the input specifications and diagnosis prompt of the adaptive framework are given in Appendix D.1–D.2. To preserve anonymity during review, code and benchmark data are not released with this submission; they will be released publicly upon acceptance.

REFERENCES

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian Ichter, Danny Driess, Pete Florence, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. SpatialVLM: Endowing vision-language models with spatial reasoning capabilities. *arXiv preprint arXiv:2401.12168*, 2024a.

Xin Chen, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, and Gang Yu. Executing your commands via motion diffusion in latent space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. How far are we to GPT-4V? closing the gap to commercial multimodal models with open-source suites. *arXiv preprint arXiv:2404.16821*, 2024b.

Gheorghe Comanici et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Caifeng Shan, Ran He, and Xing Sun. Video-MME: The first-ever comprehensive evaluation benchmark of multi-modal LLMs in video analysis. *arXiv preprint arXiv:2405.21075*, 2024.

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

Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5152–5161, 2022.

Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2003.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Jing Lin, Ailing Zeng, Shunlin Lu, Yuanhao Cai, Ruimao Zhang, Haoqian Wang, and Lei Zhang. Motion-X: A large-scale 3D expressive whole-body human motion dataset. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023.

Jing Lin, Ruisi Wang, Junzhe Lu, Ziqi Huang, Guorui Song, Ailing Zeng, Xian Liu, Chen Wei, Wanqi Yin, Qingping Sun, Zhongang Cai, Lei Yang, and Ziwei Liu. The quest for generalizable motion generation: Data, model, and evaluation. In *International Conference on Learning Representations (ICLR)*, 2026.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024a.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. LLaVA-NeXT: Improved reasoning, OCR, and world knowledge. `https://llava-vl.github.io/blog/2024-01-30-llava-next/`, January 2024b.

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, 2015.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M. Waleed Kadous, and Ion Stoica. RouteLLM: Learning to route LLMs with preference data. *arXiv preprint arXiv:2406.18665*, 2024.

OpenAI. Introducing GPT-5.4. `https://openai.com/index/introducing-gpt-5-4/`, 2026.

Mathis Petrovich, Michael J. Black, and Gül Varol. TEMOS: Generating diverse human motions from textual descriptions. In *European Conference on Computer Vision (ECCV)*, pp. 480–497, 2022.

Matthias Plappert, Christian Mandery, and Tamim Asfour. The KIT motion-language dataset. *Big Data*, 4(4):236–252, 2016. doi: 10.1089/big.2016.0028.

Abhinanda R. Punnakkal, Arjun Chandrasekaran, Nikos Athanasiou, Alejandra Quirós-Ramírez, and Michael J. Black. BABEL: Bodies, action and behavior with english labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 722–731, 2021.

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

Guy Tevet, Sigal Raab, Brian Gordon, Yoni Shafir, Daniel Cohen-Or, and Amit H. Bermano. Human motion diffusion model. In *International Conference on Learning Representations (ICLR)*, 2023.

Chongjun Tu, Lin Zhang, Pengtao Chen, Peng Ye, Xianfang Zeng, Wei Cheng, Gang Yu, and Tao Chen. FAVOR-Bench: A comprehensive benchmark for fine-grained video motion understanding. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2025.

Mengting Wan, Tara Safavi, Sujay Kumar Jauhar, Yujin Kim, Scott Counts, Jennifer Neville, Siddharth Suri, Chirag Shah, Ryen W. White, Longqi Yang, Reid Andersen, Georg Buscher, Dhruv Joshi, and Nagu Rangan. TnT-LLM: Text mining at scale with large language models. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2024.

Ziyang Wang, Shoubin Yu, Elias Stengel-Eskin, Jaehong Yoon, Feng Cheng, Gedas Bertasius, and Mohit Bansal. VideoTree: Adaptive tree-based video representation for LLM reasoning on long videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3272–3283, 2025.

Bin Yang, Rong Ou, Weisheng Xu, Jiaqi Xiong, Xintao Li, Taowen Wang, Luyu Zhu, Xu Jiang, Jing Tan, and Renjing Xu. T2MBench: A benchmark for out-of-distribution text-to-motion generation. *arXiv preprint arXiv:2602.13751*, 2026.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. MM-Vet: Evaluating large multimodal models for integrated capabilities. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 57730–57754, 2024.

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? In *International Conference on Learning Representations (ICLR)*, 2023.

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

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. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

A Benchmark Construction Details

A.1 Semantic Dimension Induction Pipeline

We follow the TnT-LLM framework (Wan et al., 2024) to induce semantic dimensions directly from text annotations, avoiding any predefined taxonomy. The pipeline consists of four sequential stages.

Stage 1: Open-ended attribute decomposition. Each text annotation in the four source datasets (HumanML3D, KIT-ML, BABEL, Motion-X) is processed independently by an LLM. The model is prompted to decompose the annotation into a set of *semantic attribute phrases* that capture distinct aspects of the described motion, without any predefined category list. The decomposition prompt is as follows:

Attribute Decomposition Prompt

```
Given the following motion description, extract a list of semantic
attribute phrases that each characterize a distinct aspect of the
described human motion.  Each phrase should be short (2-6 words),
self-contained, and describe one independent semantic property of
the motion (e.g., what body part is involved, how fast or intense
the motion is, in what direction it occurs, or how it unfolds over
time).  Do not use any predefined category list.  Output a JSON
list of strings.

Description:  {annotation}
```

Stage 2: Clustering and merging. The attribute phrases collected across all annotations are embedded using a sentence encoder and clustered via agglomerative clustering ($\cos$ similarity threshold $= 0.80$). Cluster labels are then assigned by prompting the LLM to produce a canonical category name for each cluster, and semantically overlapping clusters are manually merged.

Stage 3: Frequency filtering. For each candidate dimension, we compute the fraction of annotations in the combined corpus that contain at least one attribute phrase belonging to it. Dimensions with coverage $> 20\%$ are retained as evaluation dimensions. Table 4 reports the coverage of all candidate dimensions, including those below the threshold.

Stage 4: Retained dimensions. Six dimensions meet the 20% threshold: Action Type (AT, 97%), Body Part (BP, 55%), Temporal Structure (TS, 45%), Spatial Direction (SD, 38%), Motion Frequency (MF, 24%), and Motion Intensity (MI, 21%). Dimensions such as *Object Interaction* (14%) and *Facial Expression* (8%) were excluded due to insufficient coverage.

Table 4: Coverage of candidate semantic dimensions across the combined corpus (HumanML3D + KIT-ML + BABEL + Motion-X). Dimensions above the 20% threshold are retained for evaluation.

Dimension	Coverage (%)	Retained
Action Type	97	✓
Body Part	55	✓
Temporal Structure	45	✓
Spatial Direction	38	✓
Motion Frequency	24	✓
Motion Intensity	21	✓
Object Interaction	14	×
Facial Expression	8	×
Social Relation	6	×

A.2 Dimension-Specific Negative Sample Rewriting Strategies

Negative texts are constructed by modifying only the target dimension while leaving all other semantic attributes unchanged. Rewriting strategies are derived from a pilot study of generation

errors produced by three representative T2M models—MDM, MotionDiffuse, and MLD— ensuring that the deviations introduced reflect realistic model failure modes rather than arbitrary perturbations. Each dimension yields one **hard negative** (subtle deviation, low perceptual salience) and one **easy negative** (salient deviation, high perceptual salience) per positive sample. Table 5 summarises the rewriting strategy for each dimension with illustrative examples.

Table 5: Dimension-specific negative sample rewriting strategies. For each dimension, we list the T2M failure mode that motivates the strategy, the rewriting rule for hard and easy negatives, and a concrete example pair.

Dimension	T2M Failure Mode	Hard Negative	Easy Negative	Example
Action Type (AT)	Wrong verb: semantically similar action generated (e.g., walk instead of jog)	Replace the verb with a semantically similar one	Replace the verb with a semantically distant one	*jogs* → *walks* (hard); *jogs* → *crawls* (easy)
Body Part (BP)	Wrong limb: adjacent or bilateral body part activated instead of the target	Change a bilateral relation (*right* → *left*) or substitute an adjacent limb (knee → foot)	Substitute a cross-region limb	*raises right hand* → *raises left hand* (hard); *raises right hand* → *raises right foot* (easy)
Temporal Structure (TS)	Wrong order: sub-actions generated in incorrect sequence	Reverse the order of two adjacent sub-actions	Insert a sub-action absent from the original sequence	*steps forward then turns* → *turns then steps forward* (hard); add *pauses* between actions (easy)
Spatial Direction (SD)	Wrong direction: motion displaced along an incorrect axis	Substitute a nearby direction	Substitute the opposite direction	*steps backward* → *steps sideways* (hard); *steps backward* → *steps forward* (easy)
Motion Frequency (MF)	Wrong count: repetition count under- or over-generated	Change to an adjacent count value	Change to a substantially different count value	*waves three times* → *waves twice* (hard); *waves three times* → *waves once* (easy)
Motion Intensity (MI)	Wrong amplitude: motion executed with incorrect force or range	Change to an adjacent intensity level	Change to the opposite intensity level	*swings arm widely* → *swings arm slightly* (hard); *swings arm widely* → *barely moves arm* (easy)

Construction procedure. All negative texts are generated using an LLM assistant and subsequently reviewed manually. Each candidate negative text is evaluated on two criteria: (i) *dimensional precision*: only the target dimension is modified; (ii) *naturalness*: the resulting sentence reads as fluent English. Samples failing either criterion are either rewritten or discarded. The full rewriting prompt used for LLM-assisted generation is provided below.

Negative Sample Rewriting Prompt

```
You are given a motion description and a target semantic dimension.
Produce exactly two rewritten versions of the description:

1. Hard Negative: introduce a subtle change to the target
dimension (e.g., a nearby value or adjacent body part) while
keeping all other aspects identical. The change should be minimal
but semantically incorrect.

2. Easy Negative: introduce a salient change to the target
dimension (e.g., the opposite direction, a distant value, or a
cross-region body part) while keeping all other aspects identical.
The change should be clearly perceptible.

Rules: - Do NOT modify any dimension other than the specified
target. - Output must be fluent, natural English. - Return a JSON
object with keys "hard" and "easy".

Original description: {text}
Target dimension: {dimension}
Rewriting strategy: {strategy}
```

A.3 Full Validity Verification Results

Since both sample construction and ground-truth (GT) generation rely on automated pipelines, we conduct a systematic human validation study covering two complementary aspects: negative sample quality and GT reliability.

Annotators and protocol. Three annotators with backgrounds in computer vision and motion analysis independently complete all validation tasks on a randomly sampled subset of 300 groups (2,700 question instances across all dimensions and task types). Agreement is measured by Fleiss' κ before adjudication; the majority-vote label is used as the final reference. All three annotators are compensated at a standard research assistant rate.

A.3.1 Negative Sample Quality

Each negative sample is assessed on two binary criteria:

- **Dimensional precision**: the rewritten text modifies only the intended semantic dimension and leaves all others unchanged.
- **Visual discriminability**: the negative text, when paired with the motion, produces a visually detectable mismatch that a human observer can identify without domain expertise.

Inter-annotator agreement achieves Fleiss' $\kappa = 0.81$, indicating strong reliability. Using majority-vote labels, dimensional precision passes at **93%** and visual discriminability at **89%** overall. Per-dimension breakdown is reported in Table 6.

Table 6: Negative sample quality by dimension. DP = dimensional precision; VD = visual discriminability.

Dimension	**Hard Negative**		**Easy Negative**	
	DP (%)	VD (%)	DP (%)	VD (%)
Action Type	96	94	97	96
Body Part	94	88	95	93
Temporal Structure	92	85	94	90
Spatial Direction	93	87	95	92
Motion Frequency	91	84	93	89
Motion Intensity	90	83	92	87
Overall	**93**	**87**	**94**	**91**

A.3.2 GROUND-TRUTH RELIABILITY

We verify that automatically constructed GT aligns with human judgment across all three task types. The three annotators independently complete the judgment, ranking, and error-correction tasks on the same 300-group subset and their responses are compared against the GT.

- **Judgment task**: binary Yes/No agreement between annotator labels and GT, computed per instance. Overall agreement: **92%**.
- **Ranking task**: Kendall's τ between the annotator's ranked order (positive $>$ hard negative $>$ easy negative) and GT. Overall τ: **0.79**.
- **Error-correction task**: semantic match rate between the annotator's response and the GT (the semantic change recorded during rewriting for negatives, or "no error" for positives), judged by meaning equivalence rather than literal match (assessed by a separate LLM judge (Zheng et al., 2023)). Overall semantic match: **87%**.

All dimensions exceed 80% agreement on judgment and error correction, with Kendall's τ of at least 0.72 on ranking, confirming reliable GT quality across the full dimensional scope of the benchmark. Full per-dimension statistics are reported in Table 7.

Table 7: GT reliability by dimension across three task types. Judgment: accuracy (%); Ranking: Kendall's τ; Error Correction: semantic match rate (%).

Dimension	Judgment (%)	Ranking (τ)	Error Correction (%)
Action Type	95	0.86	91
Body Part	93	0.82	89
Temporal Structure	92	0.80	88
Spatial Direction	91	0.78	86
Motion Frequency	90	0.74	84
Motion Intensity	89	0.72	82
Overall	**92**	**0.79**	**87**

B EXPERIMENTAL DETAILS

B.1 IMPLEMENTATION DETAILS

Model selection. We evaluate six representative VLMs covering a range of model scales, training paradigms, and open/closed-source status: GPT-5.4 (OpenAI, 2026), Gemini 2.5 Pro (Comanici et al., 2025), Qwen3.5-Plus (Qwen Team, 2026), Qwen2.5-VL (Bai et al., 2025), LLaVA-1.6 (Liu et al., 2024b), and InternVL2 (Chen et al., 2024b). Models are selected based on three criteria: (1) recency and community adoption; (2) coverage of the capability spectrum, from strong reasoning-augmented closed-source models to lightweight open-source models; (3) architectural diversity, including models with dedicated temporal modeling mechanisms (e.g., Qwen2.5-VL) and standard image-encoder architectures. Table 8 summarises model-specific details.

Table 8: Details of the six evaluated VLMs. "Temporal modeling" indicates whether the model includes an architecture component specifically designed for video/temporal processing.

Model	Version / Checkpoint	Params	Open-source	Temporal Modeling
GPT-5.4	`gpt-5.4`	—	No	Standard
Gemini 2.5 Pro	`gemini-2.5-pro`	—	No	Standard
Qwen3.5-Plus	`qwen3.5-plus`	—	No	Standard
Qwen2.5-VL	`Qwen/Qwen2.5-VL-72B-Instruct`	72B	Yes	Dynamic resolution + temporal
InternVL2	`OpenGVLab/InternVL2-26B`	26B	Yes	Standard
LLaVA-1.6	`liuhaotian/llava-v1.6-34b`	34B	Yes	Standard

Unified input configuration. All models share a standardised input format to ensure comparability. Motion sequences are rendered as skeleton-line animations at a uniform frame rate of 15 FPS. Each frame is rendered at 512×512 pixels under a single front-facing orthographic projection. The rendered frames are provided to all models as a sequence of images in the order they appear in time; no audio or additional metadata is included.

Inference configuration. Commercial models (GPT-5.4, Gemini 2.5 Pro, Qwen3.5-Plus) are accessed via their official APIs using default generation settings (temperature $= 1.0$, top-$p = 1.0$). Open-source models are loaded in `bfloat16` precision and run with greedy decoding (temperature $= 0$) on NVIDIA A100 80GB GPUs. All models use a standardised output format enforced through the prompt; responses failing to match the expected format are treated as abstentions and scored as incorrect.

Evaluation metrics. The judgment task reports binary accuracy (correct Yes/No out of all instances). The ranking task reports exact-match accuracy (the fraction of instances where the predicted order exactly matches positive $>$ hard negative $>$ easy negative), computed per instance (each label rotation counts as one instance) and averaged across the dataset. The error-correction task reports semantic match rate over all 4,800 pairs, including positives whose correct response is "no error": equivalence between the model's response and the GT (the semantic change for negatives, or "no error" for positives) is judged by an LLM evaluator (OpenAI, 2026) rather than literal string match; details of the semantic equivalence judge are given in Appendix B.3.

B.2 Evaluation Prompts

We employ three task-specific prompts, one for each task type. All prompts require structured output and include a one-shot example to reduce format errors. Variables in braces (`{...}`) are filled at inference time.

Judgment Task Prompt

```
#Instruction: You are a motion-text alignment evaluator. You
are given a sequence of frames depicting a 3D human motion and a
text description. Determine whether the motion and the text are
semantically consistent.
#Output Format: Respond with a single JSON object: {"answer":
"Yes" | "No", "rationale": "<one sentence>"}
#Example:
Text: "A person jogs forward."
[motion frames]
Output: {"answer": "Yes", "rationale": "The figure moves forward
at a running pace consistent with jogging."}
#Now it is your turn.
Text: {text}
{image_frames}
```

Ranking Task Prompt

```
#Instruction: You are a motion-text alignment evaluator. You are
given a sequence of frames depicting a 3D human motion and three
text descriptions labelled A, B, and C. Rank the three texts from
best to worst alignment with the motion.

#Output Format: Respond with a single JSON object: {"ranking":
["<best>", "<second>", "<worst>"], "rationale": "<one sentence per
label>"}

#Example:
A: "A person jogs forward."
B: "A person walks forward."
C: "A person crawls forward."
[motion frames]
Output: {"ranking": ["A", "B", "C"], "rationale": {"A":
"Matches the pace and posture.", "B": "Too slow but correct
direction.", "C": "Incorrect posture entirely."}}

#Now it is your turn.
A: {text_A}
B: {text_B}
C: {text_C}
{image_frames}
```

Note. Each group is evaluated under three cyclic rotations of the label assignment (A/B/C $\rightarrow$ positive/hard/easy, hard/easy/positive, easy/positive/hard), so that every text occupies each position exactly once; each rotation is scored as a separate instance.

Error-Correction Task Prompt

```
#Instruction: You are a motion-text alignment evaluator. You are
given a sequence of frames depicting a 3D human motion and a text
description that may contain at most one semantic error on the
{dimension} dimension. If the text and the motion are inconsistent
on this dimension, identify and describe the specific mismatch. If
they are consistent, report that there is no error.

#Output Format: Respond with a single JSON object:
{"has_error": true | false, "error_dimension": "{dimension}"
| null, "text_attribute": "<what the text says>" | null,
"motion_attribute": "<what the motion shows>" | null, "correction":
"<corrected phrase>" | null}. If "has_error" is false, set all
other fields to null.

#Example 1 (Spatial Direction, error present):
Text: "A person steps forward."
[motion frames showing the person stepping backward]
Output: {"has_error": true, "error_dimension": "Spatial
Direction", "text_attribute": "forward", "motion_attribute":
"backward", "correction": "A person steps backward."}

#Example 2 (Body Part, no error):
Text: "A person raises the right hand."
[motion frames showing the person raising the right hand]
Output: {"has_error": false, "error_dimension": null,
"text_attribute": null, "motion_attribute": null, "correction":
null}

#Now it is your turn.
Text: {text}
{image_frames}
```

B.3 Semantic Equivalence Judge for Error-Correction Evaluation

Because the error-correction task requires free-form textual output, ground-truth matching is performed by an LLM judge rather than exact string comparison. The judge receives the GT (the semantic change recorded during rewriting for negatives, or "no error" for positives) and the model's predicted response, and returns a binary equivalence label. We use GPT-5.4 as the judge with the following prompt:

Semantic Equivalence Judge Prompt

```
#Instruction: You are evaluating whether two descriptions of a
motion-text mismatch convey the same semantic content.

Reference (ground truth): {gt_change}
Prediction (model output): {pred_change}

Respond with a single JSON object: {"equivalent": true | false,
"reason": "<one sentence>"}

Two descriptions are equivalent if they identify the same incorrect
attribute in the text and the same correct attribute in the motion,
even if the phrasing differs. They are NOT equivalent if the
predicted dimension, attribute value, or direction of error differs
from the reference. If the reference is "no error", the prediction
is equivalent only if it also reports no error; any reported
mismatch is non-equivalent, and vice versa.
```

To validate the judge, we manually annotated 200 randomly sampled (prediction, GT) pairs and compared the judge's binary label to the human label; agreement was 94%, confirming reliable automated evaluation.

C Root-Cause Analysis: Intervention Details

This appendix provides the implementation details for the targeted interventions in Section 5, along with complete per-subcategory accuracy tables for all intervention conditions. Each intervention is applied to the ranking task on Qwen2.5-VL, consistent with the main-paper analysis; cross-model verification on InternVL2 is reported in Appendix C.4.

C.1 Spatial Interventions (Failure Mode 1)

Rendering-stage intervention: multi-view rendering. The baseline uses a single front-facing orthographic projection, which collapses the depth axis and discards all front/back information. We progressively augment the input with additional views: **+Side View** adds one lateral (90°) view; **+Top View** further adds one top-down (90° elevation) view; **+Six Views** replaces these with six uniformly distributed surrounding views (0°, 60°, 120°, 180°, 240°, 300° azimuth at 0° elevation). All views share the same orthographic camera distance and are rendered at the same resolution as the baseline. Each additional view is appended to the input frame sequence, and the prompt is augmented with a label indicating which view each frame belongs to.

Model-stage intervention: depth maps. After the rendering-stage benefit saturates at +Six Views, we augment the input with per-frame depth maps computed directly from the skeleton's 3D joint coordinates. Each depth map is produced by projecting the 3D skeleton onto the image plane and assigning each pixel a normalised depth value proportional to its z-coordinate in camera space; pixels not covered by any joint are filled via nearest-neighbour interpolation. The depth map is provided as an additional greyscale image appended after the corresponding RGB frame, with the prompt instructing the model to use it as an auxiliary spatial cue.

Results. Table 9 reports per-subcategory accuracy across all conditions. Failure-related subcategories (front/back, depth-related actions) improve selectively under the spatial intervention, while non-failure subcategories (up/down, non-depth actions) remain stable, confirming the two-stage attribution.

Table 9: Spatial intervention: per-subcategory ranking accuracy (%). Failure-related subcategories are shown in the upper block; non-failure subcategories in the lower block. Random baseline: 17%.

Subcategory	Original	+Side View	+Top View	+Six Views	+Depth Maps
Failure-related					
Front/Back (SD)	23	35	40	41	49
Depth-related Action (AT)	27	36	42	44	56
Non-failure (control)					
Up/Down (SD)	51	54	53	53	55
Non-depth Action (AT)	72	74	74	73	75

C.2 Topology Interventions (Failure Mode 2)

Rendering-stage intervention: skinned mesh rendering. The baseline uses skeleton-line rendering, which reduces the body to homogeneous joints and connecting line segments, discarding all surface cues (skin tone, clothing, limb shape) that signal limb identity. We replace skeleton-line rendering with a skinned mesh using the SMPL body model (Loper et al., 2015) fitted to each motion sequence. The mesh is rendered with a diffuse Lambertian shading model and a fixed directional light source. All other rendering parameters (viewpoint, resolution, frame rate) are identical to the baseline.

Model-stage intervention: joint semantic annotation. Since rendering-stage recovery alone leaves a substantial gap, we inject explicit skeletal topology information by replacing the uniform-colour skinned mesh with a **colour-coded body-part segmentation**: each body segment is rendered in a distinct colour, and a legend mapping colours to part names is appended to the input. The annotated parts include Head, Neck, Torso, Left/Right Upper Arm, Left/Right Forearm, Left/Right Hand, Left/Right Thigh, Left/Right Shank, and Left/Right Foot (15 segments in total). This converts the global relational structure of skeletal topology—which patch-based encoders struggle to recover from geometry alone—into locally discriminable colour tokens that the model can directly associate with part-name labels in the legend. No geometric or positional descriptions are provided; the model is expected to infer limb identity and bilateral relations from colour correspondence between the rendered figure and the legend.

Results. Table 10 reports per-subcategory accuracy across all conditions. Lateral and chain-adjacent subcategories improve selectively; cross-region accuracy is unaffected, confirming topology-specific attribution.

Table 10: Topology intervention: per-subcategory ranking accuracy (%). Failure-related subcategories are shown in the upper block; non-failure subcategories in the lower block. Random baseline: 17%.

Subcategory	Skeleton	Skinned Mesh	+Joint Semantics
Failure-related			
Lateral (BP)	39	52	60
Chain-adjacent (BP)	42	53	62
Non-failure (control)			
Cross-region (BP)	69	70	71

C.3 Temporal Interventions (Failure Mode 3)

Rendering-stage intervention: increased frame rate. The baseline uses uniform frame sampling at 15 FPS, consistent with the unified input configuration. We progressively increase the frame rate to $2\times$, $3\times$, and $4\times$ the original by reducing the inter-frame interval while keeping the total sequence duration fixed, thus increasing the total number of input frames proportionally.

Model-stage intervention: keyframe extraction. After the rendering-stage benefit saturates at $4\times$ FPS, we replace uniform high-rate sampling with task-specific keyframe extraction computed directly

from the 3D skeleton data. Three extraction strategies are used, one per high-temporal-resolution subcategory:

- **Motion Frequency** — *Boundary-frame extraction.* Keyframes are selected at motion-direction reversals: frames where the velocity vector of the primary joint (the joint with the highest cumulative displacement) changes sign along its dominant axis. This captures the start and end of each motion cycle, directly encoding repetition count. Formally, frame t is selected if $\text{sign}(v_t) \neq \text{sign}(v_{t-1})$, where $v_t = p_t - p_{t-1}$ is the joint displacement at frame t.
- **Motion Intensity** — *Peak displacement extraction.* Keyframes are selected at local maxima of the aggregate joint displacement magnitude: $D_t = \sum_j \|p_t^j - p_{t-1}^j\|_2$, where the sum is over all body joints. Frames where $D_t > D_{t-1}$ and $D_t > D_{t+1}$ are retained, capturing the moments of maximum motion amplitude.
- **Speed (Action Type)** — *Peak velocity extraction.* Keyframes are selected at local maxima of the inter-frame displacement rate of the root joint (pelvis), which most directly encodes locomotion speed: $R_t = \|p_t^{\text{root}} - p_{t-1}^{\text{root}}\|_2 / \Delta t$. Frames where R_t is a local maximum are retained.

In all three strategies, a minimum spacing of 3 frames between consecutive keyframes is enforced to avoid near-duplicate selections. The extracted keyframes are provided to the model in temporal order with their original frame indices appended to the prompt.

Results. Table 11 reports per-subcategory accuracy across all conditions. The three high-temporal-resolution subcategories improve selectively under temporal interventions; Temporal Structure and non-speed Action Type subcategories remain stable, confirming the attribution.

Table 11: Temporal intervention: per-subcategory ranking accuracy (%). Failure-related subcategories are shown in the upper block; non-failure subcategories in the lower block. Random baseline: 17%.

Subcategory	Original	2× FPS	3× FPS	4× FPS	+Keyframe
Failure-related					
Motion Frequency (MF)	22	32	36	36	48
Motion Intensity (MI)	25	33	37	38	50
Speed (AT)	27	35	41	42	50
Non-failure (control)					
Temporal Structure (TS)	59	61	62	62	64
Non-speed Action (AT)	71	71	72	71	72

Note on the $4\times$ FPS saturation. As reported in the main paper, further increasing from $3\times$ to $4\times$ FPS yields at most a marginal gain (no more than 1 percentage point) for Motion Frequency, Motion Intensity, and speed-related actions (Table 11, $3\times$ and $4\times$ columns), while Temporal Structure remains unchanged. This saturation indicates that the rendering-stage bottleneck for high-temporal-resolution assessment is not frame rate per se, but the model's ability to localise key dynamics amid the resulting redundant frames—a limitation that keyframe extraction directly addresses by removing uninformative inter-peak frames before they enter the model.

C.4 Cross-Model Verification: InternVL2

To verify that the two-stage attribution is not specific to Qwen2.5-VL's architecture, we replicate the full intervention ladder on InternVL2 (Chen et al., 2024b). All experimental settings (rendering configurations, intervention implementations, evaluation prompts) are identical to those used for Qwen2.5-VL in Appendices C.1–C.3. Tables 12–14 report the complete results.

Across all three failure modes, the same structure holds: rendering-stage interventions selectively improve failure-related subcategories until saturation, model-stage interventions break through the saturation ceiling, and control subcategories remain stable throughout. The two models differ in absolute accuracy levels—InternVL2 starts lower on most subcategories, consistent with its smaller scale (26B vs. 72B)—but the saturation-then-breakthrough pattern and the selectivity of each intervention are consistent.

Table 12: Spatial intervention results (InternVL2, ranking task, %). Failure-related subcategories in the upper block; control subcategories in the lower block. Random baseline: 17%.

Subcategory	Original	+Side View	+Top View	+Six Views	+Depth Maps
Failure-related					
Front/Back (SD)	19	27	32	34	43
Depth-related Action (AT)	23	31	36	38	49
Control					
Up/Down (SD)	44	45	44	46	47
Non-depth Action (AT)	67	68	67	68	69

Table 13: Topology intervention results (InternVL2, ranking task, %). Failure-related subcategories in the upper block; control subcategories in the lower block. Random baseline: 17%.

Subcategory	Skeleton	Skinned Mesh	+Joint Semantics
Failure-related			
Lateral (BP)	34	46	54
Chain-adjacent (BP)	37	47	56
Control			
Cross-region (BP)	63	64	65

D Adaptive Framework Details

D.1 Input Specifications per Intervention Configuration

Table 15 reports the input specifications for each intervention configuration, using a typical 4-second motion sequence in FineMo as an example (baseline frame rate: 15 FPS, yielding approximately 60 frames). The adaptive framework activates only one intervention per sample based on the diagnostic result, so its input scale matches that of the selected configuration.

D.2 Failure Mode Diagnosis Prompt

The adaptive evaluation framework (Section 6) relies on an LLM-based diagnosis module to route each evaluation sample to its matched intervention. The per-category results in Table 3 provide indirect evidence that routing is reliable—since the adaptive framework matches the single-intervention ceiling for each failure-mode category, mis-routing would necessarily produce lower per-category numbers. This appendix evaluates routing accuracy directly and provides the full diagnosis prompt.

Single-label routing design. Since each evaluation text in FineMo is constructed by modifying a single semantic dimension, each text is associated with at most one failure mode—the one corresponding to its target dimension. The diagnosis module is therefore treated as a single-label four-class classification problem: given a text, the LLM identifies the most salient failure mode among {Depth-Related, Topology-Related, Temporal-Resolution-Related, None} and activates the corresponding intervention (or no intervention if None is selected). This design is consistent with the benchmark's single-dimension construction and avoids the inter-intervention interference that blanket activation introduces (Section 6.2).

The full diagnosis prompt is as follows. The LLM is instructed to select exactly one failure mode label, or `none` if the text does not involve any of the three known failure-prone dimensions.

Table 14: Temporal intervention results (InternVL2, ranking task, %). Failure-related subcategories in the upper block; control subcategories in the lower block. Random baseline: 17%.

Subcategory	Original	2× FPS	3× FPS	4× FPS	+Keyframe
Failure-related					
Motion Frequency (MF)	21	28	32	31	43
Motion Intensity (MI)	24	31	34	33	46
Speed (AT)	23	32	36	35	44
Control					
Temporal Structure (TS)	51	52	53	53	55
Non-speed Action (AT)	67	67	68	67	68

Table 15: Input specifications per intervention configuration (4-second sequence, baseline ≈60 frames at 15 FPS).

Configuration	Views	Rendering	Frame Sampling	Frames/View	Additional Input
Baseline	1	Skeleton-line	Uniform 15 FPS	∼60	—
Depth-Aware	6	Skeleton-line	Uniform 15 FPS	∼60	Front-view depth maps (∼60)
Topology-Aware	1	Skinned mesh + colour coding	Uniform 15 FPS	∼60	Colour–part legend
Temporal-Aware	1	Skeleton-line	Keyframe extraction	∼8–15	Frame index labels
Blanket	6	Skinned mesh + colour coding	Keyframe extraction	∼8–15	Depth maps + legend + frame indices

Failure Mode Diagnosis Prompt

```
#Instruction: You are analysing a motion description to identify
the single most salient failure mode that would challenge a
vision-language model when evaluating this text against a 3D motion.
Select exactly one label from the four options below.

A -- Depth-Related:
The text involves front/back spatial direction or an action whose
correct execution requires reasoning about depth (e.g., leaning
forward, stepping backward, facing the camera).

B -- Topology-Related:
The text refers to a body part whose identity requires
distinguishing bilateral symmetry (left vs. right limb) or
chain-adjacent limbs within the same skeletal chain (e.g., upper
arm vs. forearm, thigh vs. shin).

C -- Temporal-Resolution-Related:
The text specifies motion repetition count, speed/pace, or
amplitude/intensity that requires counting motion cycles or
measuring inter-frame displacement magnitude (e.g., waves three
times, jogs quickly, swings arm widely).

D -- None:
The text does not involve any of the above failure-prone dimensions
and should be evaluated under the standard configuration.

#Rules:
- Select exactly ONE label (A, B, C, or D).
- If multiple failure modes seem applicable, select the one most
directly expressed in the text.
- Do not select A, B, or C based on weak or ambiguous cues; prefer D
when uncertain.

#Output Format:
{"label": "A" | "B" | "C" | "D",
"rationale": "<one sentence explaining the choice>"}

Text to diagnose: {evaluation_text}
```

D.3 Routing Accuracy Evaluation

Protocol. We evaluate routing accuracy on all 1,600 FineMo groups, following the same procedure used in the adaptive framework: for each group, the original (positive) description is passed through the diagnosis module, and the predicted label is compared against the ground-truth class of the group, derived directly from sample construction: groups in the front/back or depth-dependent-action subcategories are labelled Depth-Related; groups in the lateral or chain-adjacent subcategories are labelled Topology-Related; groups in the Motion Frequency, Motion Intensity, or speed subcategories are labelled Temporal-Resolution-Related; all remaining groups are labelled None. We treat this as a standard four-class classification problem and report per-class precision, recall, and F1, as well as overall accuracy.

Results. Table 16 reports per-class routing accuracy. The diagnosis module achieves F1 scores of at least 0.92 across all four classes, with overall accuracy of **93%**. The high routing accuracy confirms that the per-category adaptive results in Table 3 reflect near-ceiling intervention matching rather than incidental gains from mis-routed samples.

Table 16: Failure mode diagnosis routing accuracy (four-class classification). P = precision, R = recall. Overall accuracy: 93%.

Class	**P (%)**	**R (%)**	**F1**
Depth-Related	92	96	0.94
Topology-Related	91	95	0.93
Temporal-Related	94	96	0.95
None	95	89	0.92
Overall accuracy		**93%**	

D.4 Routing Error Analysis

To characterise the failure cases of the diagnosis module, we manually inspect a random sample of 60 mis-routed texts (20 per failure-mode class) and categorise the error type. Table 17 summarises the findings.

Table 17: Routing error analysis (20 sampled mis-routings per failure-mode class). FN = false negative (labelled None instead of the correct mode); FP = false positive (labelled as this mode instead of None or another mode).

Class	**Error**	**Count / 20**	**Representative Example & Cause**
Depth-Related	FN	11 / 20	"The person leans slightly." — Implicit depth cue without an explicit direction word; the LLM classifies it as None.
	FP	9 / 20	"The person raises their arm overhead." — *overhead* loosely implies a vertical direction; the LLM over-interprets it as depth-related.
Topology-Related	FN	8 / 20	"The person extends the forearm." — No bilateral marker present; the LLM misses the chain-adjacent cue and labels it None.
	FP	12 / 20	"The person swings both arms widely." — *both arms* superficially resembles a bilateral reference, but no left/right distinction is required.
Temporal-Related	FN	13 / 20	"The person performs the motion briskly." — Figurative speed language not directly associated with temporal resolution by the LLM.
	FP	7 / 20	"The person walks three steps." — *three* is interpreted as a repetition count, but refers to locomotion steps rather than motion cycles.

Key observations. False negatives dominate for Depth and Temporal modes, where failure-mode semantics are expressed through implicit or figurative language that the LLM does not reliably associate with the corresponding failure mode (e.g., "leans slightly" vs. "leans forward"; "briskly" vs. "quickly"). False positives are more prevalent for Topology, where surface-level bilateral constructions (*both arms*) mislead the LLM into selecting Topology-Related even when no left/right discrimination is required. Both error types point to limits of semantic generalisation in the diagnosis LLM rather than rule-based vocabulary gaps, and are consistent with the broader known weaknesses of LLMs on implicit spatial and quantitative reasoning. The current routing accuracy of 93% is sufficient to achieve the per-category ceilings reported in Table 3; improving the diagnosis module and extending it to handle descriptions that genuinely involve multiple failure modes simultaneously are left as future work (see Section 7).