MotionChain: Fine-Grained Video Motion Understanding via Structured Decomposition

Hao Yang[1,2,3*], Bo Xu[1,2*], Jun Dan[4], Sijia Chen[3]
Baigui Sun[1,2†], and Yang Liu[1,2†]

[1]IROOTECH TECHNOLOGY, China [2]Wolf 1069 b Lab, Sany Group, China
[3]The Hong Kong University of Science and Technology (Guangzhou), China
[4]Alibaba Tongyi Lab, China
haoyang.byte@gmail.com; {bo.xu,baigui.sun,yang.liu}@irootech.com

Abstract. Recent Vision-Language Models (VLMs) have demonstrated significant progress in video understanding. However, we notice that VLMs perform coarse perception and reasoning when answering fine-grained motion questions. We investigate the reasons for this phenomenon and discover that motions in video change rapidly, so fixed frame rate sampling easily misses inter-frame motion details. Moreover, when answering motion-centric questions, VLMs tend to introduce motion-irrelevant contextual information and lack a structured and precise representation of motions, making it difficult to distinguish what motion the subject performs and when. To address these problems, we propose MotionChain, a training-free method that decouples motion question answering into a sequence of ⟨time range, subject, motion, environment⟩ tuples. Concretely, MotionChain leverages optical flow variations to calibrate when motions occur and extracts motion-relevant key frames as supplementary input, effectively alleviating the problem of missing critical motion information. For fine-grained motion question answering (QA), each tuple element in MotionChain can be converted into a visually grounded claim, providing the model with visual evidence. Experiments on motion-centric video QA benchmarks demonstrate the superiority of our MotionChain.

Keywords: Fine-grained Motion Question Answering · Motion Frame Extraction · Vision-Language Models

1 Introduction

Motion information in video carries temporally sensitive, structured semantics, and capturing fine-grained motion details directly supports downstream tasks such as embodied intelligence [49] and imitation learning. Recent Vision-Language Models (VLMs) [1, 8, 22, 40] have been trained on large-scale multimodal datasets and perform well on video question answering and multimodal

* Equal contribution.
† Correspondence to: Baigui Sun <baigui.sun@irootech.com>, Yang Liu <yang.liu@irootech.com>.

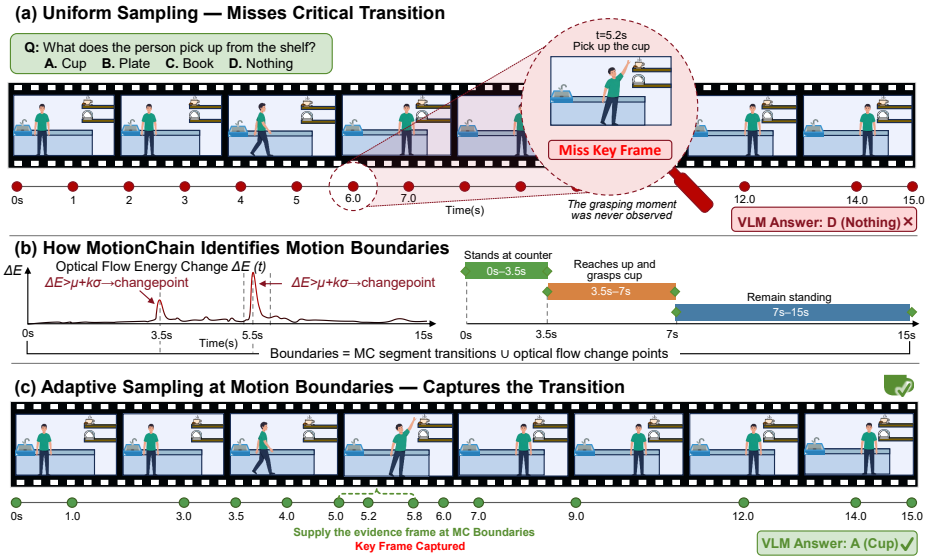

Fig. 1: Motivation and core idea of MotionChain. (a) Uniform 1 fps sampling misses the critical grasping moment ($t = 5.2$–5.8 s); the VLM never observes the motion and answers incorrectly. (b) MotionChain identifies motion boundaries by combining VLM segment transitions with optical-flow changepoints (timestamps where $|\Delta E|$ exceeds $\mu+k\sigma$). (c) MotionChain-adaptive sampling supplements the uniform grid with additional key frames at detected boundaries, capturing the transition that uniform sampling missed.

reasoning. However, they often remain limited when perceiving and understanding fine-grained motions in video. This stems from their reliance on a fixed number of frames or a fixed frame rate to understand video, lacking the ability to extract keyframes at motion transition points. Due to insufficient attention to temporal and spatial precision, they often miss critical motion information, leading to performance degradation. As illustrated in Fig. 1, when a video is fed to a VLM under fixed-rate frame sampling, the model tends to miss rapidly changing critical motion moments and fails to derive correct answers.

Despite continuous progress in spatiotemporal modeling and long-context input capabilities for video understanding [5,7,16,33], motion perception and fine-grained motion understanding remain significant weaknesses of existing models. Fine-grained motion benchmarks such as MotionBench [21] and FAVOR-Bench [38] demonstrate that, even when VLMs perform well in understanding video scenes, they still exhibit non-negligible performance bottlenecks on object motion and camera motion recognition tasks. Several existing approaches strengthen temporal reasoning and localization through training [29,31,47]. However, these approaches aim to improve general video reasoning, and the temporal localization capabilities they acquire struggle to precisely capture the keyframe intervals corresponding to instantaneous motion transitions. Another category of methods [30, 48, 50] adopts training-free frameworks, leveraging agentic search, hierarchical memory, or retrieval augmentation to extend model coverage over

long video content. Nevertheless, these methods rely on semantic relevance as the retrieval signal and cannot perceive motion-specific cues such as abrupt inter-frame optical flow changes, making it easy to miss critical motion keyframes. Furthermore, MotionSight [18] attempts to enhance motion perception through input-level visual prompts, but it lacks structured constraints on motion understanding, making it difficult to precisely characterize what motion the subject performs and when. In summary, fine-grained motion-centric video QA faces two core challenges: (1) at the motion perception level, how to leverage video motion signals to calibrate motion occurrence time and retrieve motion-relevant keyframes; (2) at the motion understanding level, how to impose structured constraints on motion understanding so that model outputs precisely cover four key elements: time range, motion subject, motion content, and the spatial region where the motion occurs.

To this end, we propose MotionChain, a training-free framework for fine-grained motion-centric video QA. Video motions are temporally continuous and multi-phased; directly feeding a complete video to a model blurs motion subjects and temporal boundaries across phases. To address this issue, MotionChain decomposes continuous video motions into an ordered sequence of structured ⟨time range, subject, motion, environment⟩ tuples. Its design philosophy can be summarized as *"first organize the timeline, then collect precise evidence"*: instead of making the model receive the entire video and answer at once, we segment the video into temporally structured motion phases, and then targeted visual evidence is supplied for each phase. Specifically, MotionChain systematically addresses the two bottlenecks from complementary perspectives of perception and understanding.

On the perception side, to tackle the keyframe retrieval challenge, MotionChain employs a VLM–LLM collaborative loop inspired by ReAct [45] for coarse-to-fine video motion segmentation. The VLM first processes the full video and identifies major motion phases with approximate time boundaries guided by the question. The LLM then reviews the segments for coverage, granularity, and accuracy, and feeds back suggestions to refine them. Based on motion segmentation, we compute a temporal motion energy curve $E(t)$ via Farneback dense optical flow [19] and detect energy changepoints as physical calibration signals. VLM boundaries and optical-flow changepoints are complementary: VLM boundaries capture semantic-level motion transitions (e.g., "starts running"), while the optical-flow changepoints detect physically abrupt motion changes (e.g., a sudden camera pan); together they form a complete and reliable set of motion boundaries. Based on this boundary set, a boundary-aware non-uniform sampler allocates additional frames to motion transition moments, because these moments typically fall between the grid points of standard uniform sampling and are therefore missed by conventional methods. In the key frame processing stage, we leverage the information in MotionChain, and use a text-driven segmentation model [3] to localize the motion-relevant regions. The keyframes with highlighting guide the VLM to attend to the motion subject rather than the background. This joint temporal–spatial optimization redistributes frame budget from static

intervals to motion-critical moments and spatially emphasizes motion-relevant regions.

On the understanding side, to address the lack of structured constraints [13, 15] and the tendency to introduce motion-irrelevant environmental information, each MotionChain tuple essentially constitutes a chained, verifiable *motion claim*. When understanding video motions, the VLM compares each candidate option against the sub-tuples of MotionChain one by one, using the frames from the corresponding temporal interval as visual evidence to determine the truth value of each motion segment. Through this mechanism, the VLM's motion understanding becomes a structured, visually traceable reasoning chain that reduces irrelevant information and precisely conveys how motions evolve over time.

Our contributions are summarized as follows.

- To address the keyframe retrieval challenge at the motion perception level, we propose a motion boundary detection mechanism combining VLM and LLM coarse-to-fine grounding with Farneback optical flow calibration [19], boundary-aware non-uniform sampling, and text-driven spatial highlighting [3] for precise spatiotemporal keyframe retrieval.
- To address the lack of structured constraints and the tendency to introduce irrelevant information in motion understanding, we introduce MotionChain, which decomposes continuous video motions into an ordered sequence of ⟨time range, subject, motion, environment⟩ tuples. Each tuple serves as a visually verifiable motion claim, converting the model's understanding into a structured and traceable reasoning chain.
- Experimental results show that on MotionBench [21], MotionChain uses only 26 frames on average, fewer than the 32-frame baselines, yet improves Qwen3-VL [1] and InternVL3.5 [40] by 0.72% and 1.14%, respectively; on FAVOR-Bench [38], adding only 3.32 extra frames beyond the 1 fps grid yields a 1.35% overall gain, demonstrating the effectiveness and efficiency of MotionChain.

2 Related Work

2.1 Video Understanding with Vision-Language Models

With the rapid development of Transformer-based architectures [16, 17] and vision-language models, video understanding has gradually shifted from dedicated spatiotemporal networks toward unified modeling frameworks built on VLMs [1, 22, 26, 27, 40]. Mainstream approaches typically adopt keyframe sampling or frame-by-frame encoding strategies, feeding visual tokens into an LLM for reasoning. One line of work compresses and aligns video features through improved video Q-Formers or connectors [10–12, 14, 26, 27, 44], while another, exemplified by the Qwen-VL and InternVL series [1, 40], encodes each frame with a strong visual encoder before passing the result to a language model. Although these methods perform well on event-level understanding, they primarily rely on static frame modeling and offer limited capacity for explicitly capturing inter-frame dynamic differences. Fine-grained motion benchmarks [21, 38] show

that even leading VLMs exhibit significant performance bottlenecks on motion sequence recognition and camera motion estimation.

Several recent efforts attempt to strengthen temporal modeling. On the training side, ThinkingWithVideos [47] introduces reinforcement-learning-based tool-augmented reasoning, Open-o3 Video [31] enhances temporal grounding through explicit spatiotemporal evidence, and VideoMind [29] optimizes reasoning with stage-wise LoRA modules. Under the training-free paradigm, MotionSight [18] employs object-centric visual spotlight and motion blur as visual prompts to boost zero-shot motion understanding.

2.2 Video Grounding and Motion-Specific Understanding

Grounding-based video understanding methods provide stronger temporal or spatial evidence for VLMs. SeViLA [46] localizes relevant frames through a self-chained image-language model, TGB [42] learns a temporal grounding bridge for efficient temporal extrapolation, GCG [39] introduces weakly supervised Gaussian contrastive grounding for video question answering, and LLaVA-ST [25] trains spatial-temporal modules for fine-grained understanding. These methods rely on learned localizers, grounding bridges, Gaussian masks, or spatial-temporal heads, and therefore differ from our training-free use of a question-aware tuple chain. AKS [35] is training-free but focuses on adaptive keyframe selection for long-video understanding, where CLIP-style relevance is more central than sub-second motion boundary calibration. Flow4Agent [28] also uses optical flow, but treats it as a long-form motion prior for video agents rather than as a calibration signal for fine-grained motion QA.

Several works directly study motion or action understanding with multimodal models. MotionLLM [4] connects human motions and videos for behavior understanding, HAIC [41] improves human action understanding and generation with caption supervision, and LLaVAction [32] evaluates and trains MLLMs for action understanding. These supervised directions are complementary to MotionChain: they improve model capability through training or action-specific data, whereas our goal is to enhance frozen VLMs at inference time by organizing temporal evidence into visually verifiable motion claims.

Different from previous work, this work focuses on efficiently solving fine-grained motion-level video questions by fusing optical-flow boundary calibration with keyframe visual enhancement for motion perception, and imposing structured tuple visual claims for motion-centric video QA.

3 Method

We present MotionChain, a training-free framework for fine-grained motion-centric video QA (Fig. 2). MotionChain decomposes continuous video motions into a temporally ordered sequence of ⟨time range, subject, motion, environment⟩ tuples, and leverages this structure to jointly address the two core challenges identified in Sec. 1. For video motion perception, the temporal boundaries of each tuple,

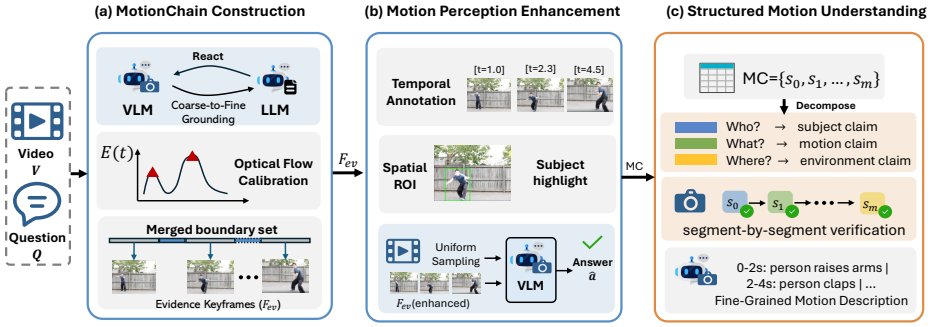

Fig. 2: Overview of the MotionChain pipeline. Given a video V and a question Q, MotionChain first constructs a temporally ordered sequence of structured tuples via VLM–LLM coarse-to-fine grounding calibrated by optical-flow changepoints (Sec. 3.1). The resulting boundary set $\mathcal{B}$ drives motion-aware perception enhancement (Sec. 3.2): boundary-adaptive sampling allocates additional frames at motion transition moments, while spatial region enhancement directs the VLM's attention to the motion subject. For structured motion understanding (Sec. 3.3), each tuple is converted into a visually grounded motion claim, enabling the VLM to verify candidate answers against segment-level visual evidence with structured reasoning.

combined with optical-flow changepoints, drive a non-uniform frame sampler that retrieves motion-critical keyframes missed by standard uniform sampling (Sec. 3.2). For motion question answering, the tuple fields are converted into a chain of visually verifiable claims, constraining the VLM to reason about each motion phase with grounded visual evidence rather than producing unstructured holistic descriptions (Sec. 3.3). The details of the algorithm are summarized in Algorithm 1.

3.1 MotionChain Construction

MotionChain decomposes a video V of duration T into a temporally ordered sequence of segments $\mathcal{MC} = \{s_j\}_{j=1}^{M}$, where each segment is a structured tuple:

$$s_j = \big([t_j^{\mathrm{s}}, t_j^{\mathrm{e}}],\ \mathrm{sub}_j,\ \mathrm{mot}_j,\ \mathrm{env}_j\big), \tag{1}$$

recording the time range, primary motion subject, motion understanding, and environment context, respectively. Different from free-form video captioning, this decomposition maintains strict temporal contiguity ($t_j^{\mathrm{e}} \le t_{j+1}^{\mathrm{s}}$) and assigns each moment in the video to exactly one motion phase, casting video motions into a temporally structured representation.

Coarse-to-Fine Video Grounding. The construction of MotionChain follows a coarse-to-fine loop between a VLM and an LLM, inspired by the ReAct paradigm [45]. Fig. 3 visualizes this construction process and the subsequent optical-flow boundary calibration. In the coarse stage, the VLM observes the full video and outputs an initial segmentation that identifies major motion phases with approximate time ranges. Crucially, the grounding is question-aware: the question Q is provided as a focus directive, so the same video yields different

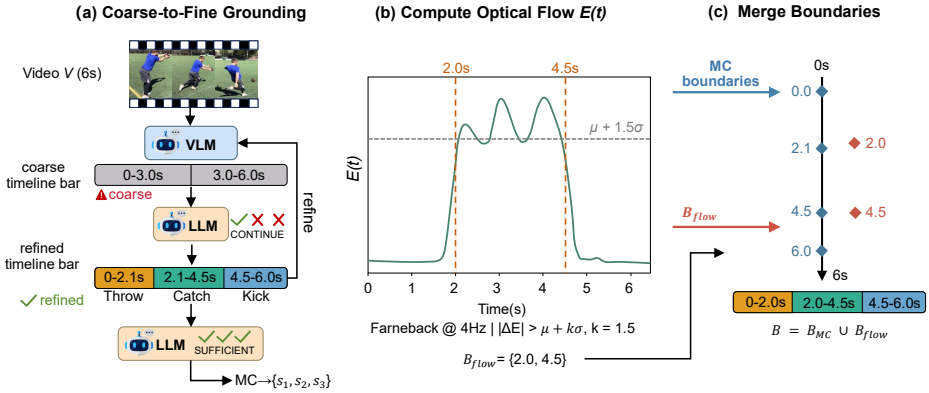

Fig. 3: MotionChain construction process. **Left**: the VLM produces a question-aware coarse motion segmentation; the LLM evaluates it for coverage, granularity, and accuracy, then the VLM refines the segmentation with the feedback. **Right**: optical-flow changepoints (red) complement VLM-predicted boundaries (blue), forming the merged boundary set $\mathcal{B}$.

segmentations for different questions (*e.g.*, querying about the ball's trajectory versus counting a person's repetitions produces different motion phases). In the refinement stage, the LLM evaluates the initial segmentation along three axes—temporal coverage, granularity, and accuracy—and generates concise feedback. The VLM then re-examines the video with this feedback and produces a refined segmentation with corrected boundaries and more detailed motion labels.

Optical Flow Boundary Calibration. VLM-predicted segment boundaries are constrained by the model's temporal resolution, which depends on the input frame rate and internal temporal attention. To obtain sub-second boundary accuracy, we compute a temporal motion energy profile $E(t)$ via Farneback dense optical flow [19] at 4 Hz:

$$E(t) = \frac{1}{HW} \sum_{x,y} \sqrt{u(x,y)^2 + v(x,y)^2}, \tag{2}$$

where $u(x,y)$ and $v(x,y)$ are the horizontal and vertical flow components between consecutive frames. Changepoints—timestamps where the motion undergoes abrupt physical change, are identified by thresholding the energy difference:

$$\mathcal{B}_{\text{flow}} = \left\{ t \;\middle|\; |\Delta E(t)| > \mu_\Delta + k\sigma_\Delta \right\}, \quad k = 1.5, \tag{3}$$

where μ_Δ and σ_Δ denote the mean and standard deviation of the energy differences across all consecutive frame pairs. The final boundary set merges MotionChain segment transitions and optical-flow changepoints:

$$\mathcal{B} = \{t_j^{\text{s}}, t_j^{\text{e}}\}_{j=1}^{M} \cup \mathcal{B}_{\text{flow}}. \tag{4}$$

These two sources are complementary: VLM boundaries capture semantically meaningful motion transitions, while optical-flow changepoints detect physically abrupt motion changes that the VLM may not perceive at its input frame rate.

3.2 Motion Perception Enhancement

With the boundary set $\mathcal{B}$ and the structured fields of each segment s_j, we enhance the VLM's visual perception from both temporal and spatial dimensions, directly addressing the challenge that fixed-rate sampling misses critical motion transitions.

Boundary-Aware Non-Uniform Sampling. Standard uniform sampling at 1 fps distributes frames evenly across the video, which often misses motion transitions that occupy only a fraction of a second. We observe that the most informative moments, where motions change, typically fall between the integer-second grid points of the 1 fps grid. To capture these moments, we introduce a boundary-aware sampler that selects additional evidence frames at non-grid timestamps. Specifically, for each candidate boundary timestamp $t \in \mathcal{B}$, we compute its distance from the nearest uniform grid point:

$$d(t) = |t - \mathrm{round}(t)|. \tag{5}$$

Timestamps satisfying $d(t) \geq \tau$ (*i.e.*, sufficiently far from the 1 fps grid) are retained as evidence frame candidates. These candidates are ranked by $d(t)$ in descending order and deduplicated with a minimum spacing of 0.3 s, yielding at most K supplementary evidence frames $\mathcal{F}_{\mathrm{ev}}$. The final frame set sent to the VLM combines the uniform baseline with the evidence frames:

$$\mathcal{F} = \mathcal{F}_{\mathrm{uniform}} \cup \mathcal{F}_{\mathrm{ev}}. \tag{6}$$

The boundary-aware sampler does not replace the uniform baseline; it supplements it with frames at motion-critical moments that uniform sampling is structurally guaranteed to miss, improving temporal information density under the same VLM context budget.

Temporal Annotation. When VLMs process multi-image inputs, the chronological ordering among visually similar frames can be ambiguous. To provide explicit temporal cues, we annotate each frame with a text marker $[\mathtt{t} = \tau\mathtt{s}]$ embedded before the frame in the VLM input, where τ is the frame's timestamp in seconds. This allows the VLM to infer temporal ordering, duration, and speed of motions (*e.g.*, two frames 0.3 s apart depict successive moments of the same motion) without injecting any semantic content from MotionChain.

Spatial Region Enhancement. Fine-grained motion QA often requires the VLM to focus on a specific region of the frame—such as the hand performing a gesture, rather than the full scene. We leverage the subject field sub_j from each segment to localize the motion-relevant spatial region. Given a text-driven segmentation model [3], sub_j serves as a natural language query to detect the motion subject in each evidence frame, producing a per-frame bounding box $\mathbf{b}_{j,t}$. Bounding boxes across frames are aggregated into a unified region of interest $\mathbf{R}$ with padding:

$$\mathbf{R} = \mathrm{Pad}\Big(\bigcup_{j,t} \mathbf{b}_{j,t},\ \alpha\Big), \tag{7}$$

where α controls the padding ratio to accommodate inter-frame positional variation. Evidence frames are then highlighted around $\mathbf{R}$, reducing background distraction and effectively increasing the spatial resolution of the motion-relevant area within the VLM's fixed input resolution. When the segmentation model is unavailable, we fall back to an optical-flow-based motion region computed from the accumulated flow magnitude map.

Extension to Camera Motion. The enhancements above target *object motion*; for *camera motion* questions, the motion originates from the camera itself and per-object tuple decomposition does not apply. We instead extract two complementary physical cues from the optical-flow field. The spatially averaged flow vector

$$\bar{\mathbf{f}} = \frac{1}{HW} \sum_{x,y} \big(u(x,y),\, v(x,y)\big) \tag{8}$$

indicates the camera's movement direction and speed, while the structural similarity SSIM [43] between temporally spaced frames quantifies overall scene change. Both cues are formatted as textual evidence and prepended to the VLM prompt, enabling the model to reason about camera motion with objective measurements rather than visual impression alone.

3.3 Motion Question Answering via Visually Grounded Claims

Beyond perception enhancement, the structured decomposition of MotionChain naturally addresses the second challenge: the lack of structured constraints on motion understanding. VLMs tend to produce holistic descriptions of the entire video, blending motion-relevant and motion-irrelevant information indiscriminately, which makes it difficult to pinpoint what motion the subject performs and when. MotionChain tackles this problem by converting each tuple s_j into a visually grounded motion claim—a proposition whose truth value can be verified against the visual content within the corresponding temporal interval.

Tuple-to-Claim Conversion. Each tuple s_j naturally decomposes into a set of element-wise claims that can be independently verified:

$$c_j^{(\text{sub})} = \big[\,\text{sub}_j \text{ is present in } V[t_j^{\text{s}} : t_j^{\text{e}}]\,\big], \tag{9}$$

$$c_j^{(\text{mot})} = \big[\,\text{sub}_j \text{ performs mot}_j \text{ during } [t_j^{\text{s}},\, t_j^{\text{e}}]\,\big], \tag{10}$$

$$c_j^{(\text{env})} = \big[\,\text{the environment is env}_j \text{ during } [t_j^{\text{s}},\, t_j^{\text{e}}]\,\big], \tag{11}$$

where $V[t_j^{\text{s}} : t_j^{\text{e}}]$ denotes the frames within the j-th segment's temporal interval. Each claim can be verified by the VLM using only the temporally annotated evidence frames from $\mathcal{F}$ that fall within $[t_j^{\text{s}}, t_j^{\text{e}}]$, rather than the entire video. This factored representation constrains the VLM to attend to the specific time window and motion subject relevant to each claim, effectively suppressing hallucination from motion-irrelevant context.

Chained Verification. When answering a motion question, the VLM evaluates each candidate option against the complete chain of claims $\{c_j^{(\cdot)}\}_{j=1}^{M}$. The VLM

receives the enhanced evidence frames $\mathcal{F}$ annotated with temporal markers, and reasons over the motion chain to determine which option best aligns with the visual evidence across all segments. This mechanism converts the VLM's motion understanding from a free-form holistic response into a structured, temporally traceable reasoning chain: instead of describing "what happens in the video" in aggregate, the model reasons segment by segment, precisely addressing what the subject does and when it occurs. The structured format naturally prevents motion-irrelevant environmental information from dominating the response and delivers concise, precise understandings of how motions evolve over time.

Algorithm 1 The MotionChain Pipeline

Require: Video V, question Q, VLM M, LLM L
Ensure: Answer $\hat{a}$
// MotionChain Construction (Sec. 3.1)
$\mathcal{MC}^{(0)} \leftarrow M(V, Q)$ {question-aware coarse grounding}
fb $\leftarrow L(\mathcal{MC}^{(0)})$ {LLM evaluates coverage, granularity, accuracy}
$\mathcal{MC} \leftarrow M(V, Q, \text{fb})$ {VLM refines with feedback}
$E(t) \leftarrow \text{FarnebackFlow}(V)$ {motion energy profile, Eq. 2}
$\mathcal{B}_{\text{flow}} \leftarrow \{t \mid |\Delta E(t)| > \mu_\Delta + k\sigma_\Delta\}$ {Eq. 3}
$\mathcal{B} \leftarrow \{t_j^{\text{s}}, t_j^{\text{e}}\}_{j=1}^{M} \cup \mathcal{B}_{\text{flow}}$ {merge boundaries, Eq. 4}
// Motion Perception Enhancement (Sec. 3.2)
$\mathcal{F}_{\text{ev}} \leftarrow \text{NonUniformSample}(\mathcal{B}, \tau, K)$ {Eq. 5}
$\mathcal{F} \leftarrow \mathcal{F}_{\text{uniform}} \cup \mathcal{F}_{\text{ev}}$ {Eq. 6}
for each $f_t \in \mathcal{F}_{\text{ev}}$ **do**
  $\mathbf{R} \leftarrow \text{SpatialGround}(f_t, \text{sub}_j)$ {text-driven segmentation, Eq. 7}
  $f_t \leftarrow \text{Highlight}(f_t, \mathbf{R})$
end for
Annotate each $f_t \in \mathcal{F}$ with $[\mathtt{t} = \tau\mathtt{s}]$
// Structured Motion Expression (Sec. 3.3)
$\hat{a} \leftarrow M(\mathcal{F}, Q)$ {answer with grounded claims}
return $\hat{a}$

4 Experiments

4.1 Experimental Setup

Benchmarks. We evaluate on two fine-grained video motion benchmarks: FAVOR-Bench [38], which contains 8184 open-ended questions across six categories covering action detail and camera motion, and MotionBench [21], whose public dev set includes 4018 multiple-choice questions across six categories such as action order and repetition count.

Implementation Details. Since MotionBench videos have an average duration of only 8.4 s and 41.4% are shorter than 5 s, simple motion questions on short videos can often be answered without supplementary evidence frames. To filter out such easy cases, we employ three VLMs—Qwen3-VL-8B [1], InternVL3.5-8B [40], and Molmo2-8B [8]—to answer each question with 32 uniformly sampled frames. Samples where all three models agree are returned directly as simple cases; disputed samples are routed to the MotionChain pipeline for further

enhancement. On FAVOR-Bench, all samples are processed through the full MotionChain pipeline to validate its effectiveness on longer videos. During MotionChain construction, the LLM for the ReAct [45] coarse-to-fine loop is Qwen-Plus; all VLMs are served via vLLM [24]. Both VLM and LLM inference use $temperature = 0.7$, $top\_p = 0.8$, and $top\_k = 20$. Optical flow is computed via Farneback [19] at 4 Hz with energy changepoint threshold $k = 1.5$; SAM3 [3] provides text-driven spatial grounding.

4.2 Main Results

Results on FAVOR-Bench. Tab. 1 shows that MotionChain with Qwen3-VL-8B achieves 58.03% overall accuracy, outperforming all compared methods. Notably, the adaptive sampler averages only 23.8 frames per video, just 3.32 frames beyond the 20.58-frame 1 fps baseline, yet all six categories improve over the baseline, with the largest gains on Single Action Detail (+2.83%), Non-Subject Motion (+3.13%), and Camera Motion (+2.79%)—categories that demand precise temporal localization and benefit most from structured decomposition.

Table 1: Comparison on FAVOR-Bench [38]. Overall and per-category accuracy (%). ∗: leaderboard [38]; †: original paper. Best results are in **bold**.

Method	# Frames	Overall	AS	HAC	SAD	MAD	CM	NSM
Proprietary VLMs*								
GPT-4o [23]	1 fps	42.09	40.65	45.10	42.84	45.48	36.00	48.44
Gemini-1.5-Pro [36]	1 fps	49.87	49.22	53.73	48.80	54.85	41.58	56.25
Open-Source VLMs*								
Qwen2.5VL-7B [2]	1 fps	40.76	39.48	43.28	43.14	43.65	33.49	39.60
Qwen2.5VL-72B [2]	1 fps	48.14	50.28	46.98	48.13	51.78	40.28	51.56
InternVL2.5-8B [6]	8	34.59	31.97	38.68	38.09	37.76	26.14	35.94
InternVL2.5-78B [6]	8	38.54	38.38	40.62	39.05	43.65	29.40	39.06
Qwen3-VL-8B [1]	1 fps	56.68	57.57	64.11	52.83	59.67	46.05	64.06
***Enhancement Methods**†*								
MotionSight (Qwen2.5VL-7B) [18]	1 fps	45.10	44.00	49.50	44.90	48.80	38.10	39.10
MotionSight (InternVL3-78B) [18]	16	53.80	56.20	58.90	52.80	58.40	37.10	57.80
MotionChain								
+ Qwen3-VL-8B	23.8	**58.03**	**58.02**	**64.44**	**55.66**	**60.83**	**48.84**	**67.19**

Results on MotionBench. Tab. 2 presents the full comparison. MotionChain with Qwen3-VL-8B achieves the highest overall accuracy of 63.59%, surpassing even the 106B proprietary GLM-4.5V with only 8B parameters. The adaptive sampler averages 26.24 frames for Qwen3 and 26.62 for InternVL3.5, close to the 32-frame baseline, yet still delivers consistent gains. Both backbone VLMs benefit from the enhancement pipeline. The most pronounced improvements appear on categories demanding precise temporal reasoning: Action Order rises by over 2% for both models, and InternVL's Camera Motion increases from 48.57% to 54.29%. The optical-flow-derived textual cues are particularly effective when the base VLM has weaker visual perception.

Table 2: Comparison on MotionBench. Overall and per-category accuracy (%). ∗: leaderboard [21]; †: original paper; ‡: reproduced by us. Adaptive # Frames denotes the average per sample.

Method	# Frames	Overall	MR	LM	CM	MO	AO	RC
Proprietary VLMs*								
GPT-5 [34]	2 fps	58.81	61.37	66.67	58.61	73.79	**49.38**	26.74
Gemini-2.5-Flash [9]	2 fps	56.12	59.25	59.97	59.90	72.55	44.40	29.14
GLM-4.5V (106B) [22]	2 fps	61.42	65.08	65.16	58.72	77.38	48.34	40.37
GLM-4V-Plus [22]	2 fps	62.80	64.10	67.00	**67.40**	73.50	46.70	42.80
Open-Source VLMs*								
Seed1.5-VL [20]	2 fps	58.81	61.37	66.67	58.61	73.79	**49.38**	26.74
Qwen2.5-VL-72B [2]	64	56.14	59.48	58.96	47.95	72.89	45.53	28.73
InternVL2.5-78B [6]	16	60.90	65.20	64.60	55.60	75.70	46.90	33.70
Kimi-VL (16B) [37]	2 fps	54.60	58.70	55.78	45.64	73.48	42.08	22.52
Qwen3-VL-8B‡ [1]	32	62.87	67.52	66.12	60.78	**79.57**	45.47	37.00
InternVL3.5-8B‡ [40]	32	56.25	62.11	60.44	48.57	71.16	38.92	33.00
Molmo2-8B‡ [8]	32	62.47	66.17	65.38	62.08	77.68	43.74	**43.25**
Enhancement Methods								
MotionSight† (Qwen2.5VL-7B) [18]	1 fps	55.60	59.70	58.10	48.30	73.60	40.10	33.50
MotionSight† (InternVL3-78B) [18]	16	63.00	68.50	65.40	58.70	78.60	47.60	37.00
Open-o3-Video‡ (7B) [31]	1 fps	56.50	61.37	58.24	58.44	72.03	40.08	28.75
MotionChain								
+ Qwen3-VL-8B	26.24	**63.59**	**69.28**	**67.22**	59.74	78.55	47.59	36.25
+ InternVL3.5-8B	26.62	57.39	62.38	58.97	54.29	73.62	41.43	32.50

4.3 Ablation Study

We ablate two stages of the enhancement pipeline, applying each uniformly to every sample without the consensus routing used in the main results: (1) MC Frame, which applies MotionChain-guided adaptive frame extraction alone; (2) MC Frame + Visual Enhancement, which further adds temporal annotation and SAM3 [3]-based spatial region enhancement.

Tab. 3 presents the same analysis on FAVOR-Bench, where the trend differs notably. Both variants consistently improve across *all* object-motion categories without any consensus filtering, and MC Frame alone raises overall accuracy by 1.35%. Notably, Camera Motion is handled by a dedicated tool based on the optical-flow cues introduced in Eq. (8), independent of the MC Frame and Visual Enhancement stages; the separated CM column therefore reflects this independent contribution, which brings a 2.97% gain. Among object-motion categories, the largest gains appear on Non-Subject Motion and Single Action Detail, which benefit most from fine-grained temporal decomposition. Comparing MC Frame with MC Frame + Visual Enhancement, the latter further improves SAD (+0.24%), MAD (+0.41%), and NSM (+1.56%), since SAM3-based spatial region enhancement helps the VLM focus on the correct motion subject rather than the full scene, which is especially useful when the question involves fine-grained detail or non-primary subjects. This contrast with MotionBench highlights a core finding: structured decomposition is universally beneficial for open-ended

description tasks, whereas multiple-choice questions require selective routing to preserve already-reliable answers.

Table 3: Ablation of MotionChain components on FAVOR-Bench (Qwen3-VL-8B). MC Frame and MC Frame + Visual Enhancement are applied to all 8184 samples. Best results per column are **bold**.

Method	Overall	Object Motion					Camera Motion
		AS	HAC	SAD	MAD	NSM	CM
Qwen3-VL-8B [1]	56.68	57.57	64.11	52.83	59.67	64.06	46.05
+ MC Frame	**58.03**	**58.02**	**64.44**	55.66	60.83	67.19	–
+ MC Frame + Visual Enhancement	57.93	57.53	63.98	**55.90**	**61.24**	**68.75**	–
+ Optical-Flow Camera Cues	–	–	–	–	–	–	**49.02**

4.4 Analysis

The three-VLM consensus vote partitions MotionBench by difficulty: 52.8% of samples reach unanimous agreement with 78.6% accuracy and are returned directly, while only the remaining 47.2% disputed cases enter the full MotionChain pipeline. Tab. 4 verifies that the gain is not due to model ensembling or a larger frame budget: strict majority voting, Qwen3 fallback, and larger uniform-frame baselines all remain below MotionChain.

Table 4: Routing fairness and frame-budget controls on MotionBench. 3-VLM uses Qwen3-VL-8B, InternVL3.5-8B, and Molmo2-8B with 32 frames each; Err denotes samples exceeding the 65,536-token input limit.

Method	# Frames	Valid/Err	Overall	MR	LM	CM	MO	AO	RC
Qwen3 uniform	16@896	4018/0	60.85	65.16	65.93	55.84	76.52	47.01	33.75
Qwen3 uniform	64@896	3514/504	61.30	65.58	67.29	52.60	76.96	46.88	37.96
Qwen3 uniform	128@448	3127/891	60.35	64.99	65.13	49.65	74.72	47.17	**39.56**
3-VLM majority	3×32	4018/0	58.51	64.41	58.97	56.10	75.22	39.11	34.75
3-VLM maj. + Qwen3 fallback	3×32	4018/0	60.35	65.97	60.99	58.70	76.52	42.77	35.25
MotionChain + Qwen3-VL-8B	26.24	4018/0	**63.59**	**69.28**	**67.22**	**59.74**	**78.55**	**47.59**	36.25

Tab. 5 reports end-to-end latency. Although the full pipeline costs more than direct 32-frame answering, the router avoids this cost on easy samples and applies the full pipeline only to disputed cases.

Table 5: End-to-end wall-clock latency on MotionBench. Each MotionChain row reports cumulative latency up to that stage.

Configuration	Med.	P95
Direct Qwen3, 32f	3.95s	8.39s
+ MC construction	19.82s	35.40s
+ MC-frame selection	24.62s	42.60s
+ SAM3 ROI enhancement	25.62s	43.43s
+ MC-Verify reasoning	26.97s	45.99s

Fig. 4 illustrates two representative improvement cases, corresponding to the perception and understanding stages of MotionChain, respectively.

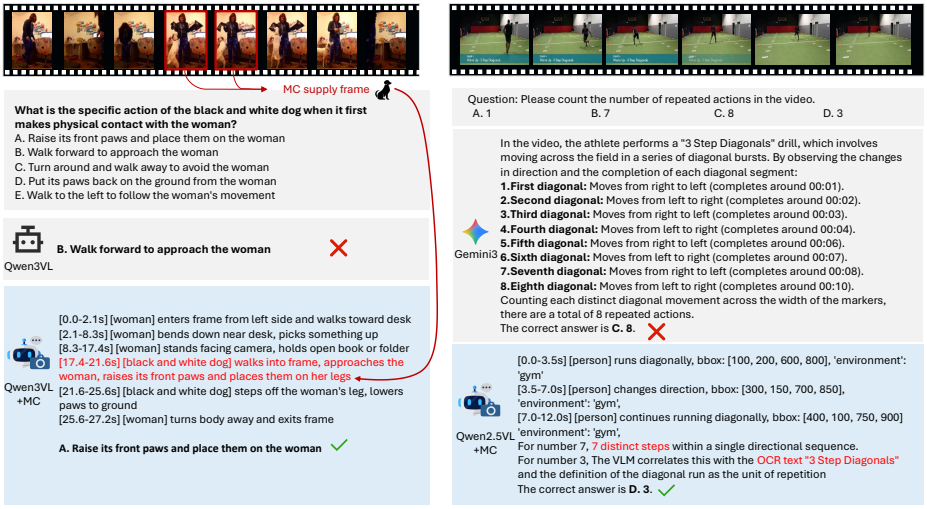

Fig. 4: Qualitative example of MotionChain. Left: uniform 1 fps sampling misses a brief paw-raising moment ($t = 17$–21 s); MotionChain-adaptive sampling supplements evidence frames (red borders) that capture it, yielding the correct answer. **Right:** without structured decomposition, Gemini-3 counts eight individual traversals and answers "8"; MotionChain groups them into three complete drill repetitions via temporally ordered tuples and OCR evidence, arriving at "3".

In the perception case, a 27.2 s video shows a dog entering the frame, approaching a woman, and briefly raising its front paws onto her legs. The question asks what action the dog performs upon first physical contact. Under uniform 1 fps sampling, the paw-raising moment spans less than two seconds and falls between grid points, so the VLM never observes it and mistakenly answers "approaching". MotionChain segments the video into six motion phases and detects an optical-flow changepoint near $t = 17.4$ s; the boundary-aware sampler supplements two evidence frames around this transition, enabling the VLM to correctly identify the paw-raising action.

In the understanding case, an athlete performs a "3 Step Diagonals" warm-up drill, running diagonally back and forth across the field. The question asks the model to count the number of repeated actions. Without structured decomposition, Gemini-3 enumerates eight individual one-way diagonal traversals and selects the wrong answer "8". MotionChain resolves this ambiguity by constructing three temporally ordered tuples, each corresponding to one complete drill repetition. The VLM further leverages the OCR text "3 Step Diagonals" as semantic evidence, correctly identifying the complete drill pattern as the counting unit and arriving at the answer "3". This case highlights how structured motion understanding prevents the VLM from conflating different levels of motion granularity.

5 Conclusion

We present MotionChain, a training-free framework for motion-centric video QA that decomposes continuous video motions into temporally ordered ⟨time range, subject, motion, environment⟩ tuples. On the perception side, VLM–LLM collaborative segmentation with optical-flow calibration locates motion-critical timestamps, and a boundary-aware sampler retrieves evidence frames missed by fixed-rate sampling. On the understanding side, each tuple is converted into a visually grounded motion claim, constraining the VLM to reason segment by segment with traceable visual evidence. Experiments on MotionBench and FAVOR-Bench show that MotionChain with 8B VLMs rivals or surpasses more than 100B scale models and training-based methods on motion-centric QA, verifying the effectiveness and generalization of MotionChain within this scope.

References

1. Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., et al.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025)
2. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., et al.: Qwen2.5-vl technical report. arXiv preprint arXiv:2502.13923 (2025)
3. Carion, N., Gustafson, L., Hu, Y.T., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K.V., Khedr, H., Huang, A., et al.: Sam 3: Segment anything with concepts. arXiv preprint arXiv:2511.16719 (2025)
4. Chen, L.H., Lu, S., Zeng, A., Zhang, H., Wang, B., Zhang, R., Zhang, L.: Motionllm: Understanding human behaviors from human motions and videos. IEEE Transactions on Pattern Analysis and Machine Intelligence (2025)
5. Chen, Y., Xue, F., Li, D., Hu, Q., Zhu, L., Li, X., Fang, Y., Tang, H., Yang, S., Liu, Z., et al.: Longvila: Scaling long-context visual language models for long videos. arXiv preprint arXiv:2408.10188 (2024)
6. Chen, Z., Wang, W., Cao, Y., Liu, Y., Gao, Z., Cui, E., Zhu, J., Ye, S., Tian, H., Liu, Z., et al.: Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. arXiv preprint arXiv:2412.05271 (2024)
7. Cheng, Z., Leng, S., Zhang, H., Xin, Y., Li, X., Chen, G., Zhu, Y., Zhang, W., Luo, Z., Zhao, D., et al.: Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. arXiv preprint arXiv:2406.07476 (2024)
8. Clark, C., Zhang, J., Ma, Z., Park, J.S., Salehi, M., Tripathi, R., Lee, S., Ren, Z., Kim, C.D., Yang, Y., et al.: Molmo2: Open weights and data for vision-language models with video understanding and grounding. arXiv preprint arXiv:2601.10611 (2026)
9. Comanici, G., Bieber, E., Schaekermann, M., Pasupat, I., Sachdeva, N., Dhillon, I., Blistein, M., Ram, O., Zhang, D., Rosen, E., 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)
10. Dan, J., Jin, T., Chi, H., Dong, S., Xie, H., Cao, K., Yang, X.: Trust-aware conditional adversarial domain adaptation with feature norm alignment. Neural Networks **168**, 518–530 (2023)

11. Dan, J., Jin, T., Chi, H., Shen, Y., Yu, J., Zhou, J.: Homda: High-order moment-based domain alignment for unsupervised domain adaptation. Knowledge-Based Systems **261**, 110205 (2023)
12. Dan, J., Liu, M., Xie, C., Yu, J., Xie, H., Li, R., Dong, S.: Similar norm more transferable: Rethinking feature norms discrepancy in adversarial domain adaptation. Knowledge-Based Systems **296**, 111908 (2024)
13. Dan, J., Liu, W., Liu, M., Xie, C., Dong, S., Ma, G., Tan, Y., Xing, J.: Hogda: Boosting semi-supervised graph domain adaptation via high-order structure-guided adaptive feature alignment. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 11109–11118 (2024)
14. Dan, J., Liu, W., Xie, X., Yu, H., Dong, S., Tan, Y.: Tfgda: Exploring topology and feature alignment in semi-supervised graph domain adaptation through robust clustering. Advances in Neural Information Processing Systems **37**, 50230–50255 (2024)
15. Dan, J., Liu, Y., Deng, J., Xie, H., Li, S., Sun, B., Luo, S.: Topofr: A closer look at topology alignment on face recognition. Advances in Neural Information Processing Systems **37**, 37213–37240 (2024)
16. Dan, J., Liu, Y., Sun, B., Deng, J., Luo, S.: Transface++: Rethinking the face recognition paradigm with a focus on accuracy, efficiency, and security. IEEE Transactions on Pattern Analysis and Machine Intelligence (2025)
17. Dan, J., Liu, Y., Xie, H., Deng, J., Xie, H., Xie, X., Sun, B.: Transface: Calibrating transformer training for face recognition from a data-centric perspective. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 20642–20653 (2023)
18. Du, Y., Fan, T., Nan, K., Xie, R., Zhou, P., Li, X., Yang, J., Yang, Z., Tai, Y.: Motionsight: Boosting fine-grained motion understanding in multimodal llms. arXiv preprint arXiv:2506.01674 (2025)
19. Farnebäck, G.: Two-frame motion estimation based on polynomial expansion. In: Scandinavian conference on Image analysis. pp. 363–370. Springer (2003)
20. Guo, D., Wu, F., Zhu, F., Leng, F., Shi, G., Chen, H., Fan, H., Wang, J., Jiang, J., Wang, J., et al.: Seed1. 5-vl technical report. arXiv preprint arXiv:2505.07062 (2025)
21. Hong, W., Cheng, Y., Yang, Z., Wang, W., Wang, L., Gu, X., Huang, S., Dong, Y., Tang, J.: Motionbench: Benchmarking and improving fine-grained video motion understanding for vision language models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 8450–8460 (2025)
22. Hong, W., Yu, W., Gu, X., Wang, G., Gan, G., Tang, H., Cheng, J., Qi, J., Ji, J., Pan, L., et al.: Glm-4.5 v and glm-4.1 v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. arXiv preprint arXiv:2507.01006 (2025)
23. Hurst, A., Lerer, A., Goucher, A.P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A., Welihinda, A., Hayes, A., Radford, A., et al.: Gpt-4o system card. arXiv preprint arXiv:2410.21276 (2024)
24. Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C.H., Gonzalez, J., Zhang, H., Stoica, I.: Efficient memory management for large language model serving with pagedattention. In: Proceedings of the 29th symposium on operating systems principles. pp. 611–626 (2023)
25. Li, H., Chen, J., Wei, Z., Huang, S., Hui, T., Gao, J., Wei, X., Liu, S.: Llava-st: A multimodal large language model for fine-grained spatial-temporal understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8592–8603 (2025)

26. Li, K., He, Y., Wang, Y., Li, Y., Wang, W., Luo, P., Wang, Y., Wang, L., Qiao, Y.: Videochat: Chat-centric video understanding. Science China Information Sciences **68**(10), 200102 (2025)
27. Lin, B., Ye, Y., Zhu, B., Cui, J., Ning, M., Jin, P., Yuan, L.: Video-llava: Learning united visual representation by alignment before projection. In: Proceedings of the 2024 conference on empirical methods in natural language processing. pp. 5971–5984 (2024)
28. Liu, R., Sun, S., Tang, H., Gao, W., Li, G.: Flow4agent: Long-form video understanding via motion prior from optical flow. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 23817–23827 (2025)
29. Liu, Y., Lin, K.Q., Chen, C.W., Shou, M.Z.: Videomind: A chain-of-lora agent for long video reasoning. arXiv preprint arXiv:2503.13444 (2025)
30. Ma, Z., Gou, C., Shi, H., Sun, B., Li, S., Rezatofighi, H., Cai, J.: Drvideo: Document retrieval based long video understanding. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 18936–18946 (2025)
31. Meng, J., Li, X., Wang, H., Tan, Y., Zhang, T., Kong, L., Tong, Y., Wang, A., Teng, Z., Wang, Y., et al.: Open-o3 video: Grounded video reasoning with explicit spatio-temporal evidence. arXiv preprint arXiv:2510.20579 (2025)
32. Qi, H., Ye, S., Mathis, A., Mathis, M.W.: Llavaction: evaluating and training multi-modal large language models for action understanding. arXiv preprint arXiv:2503.18712 (2025)
33. Qin, M., Liu, X., Liang, Z., Shu, Y., Yuan, H., Zhou, J., Xiao, S., Zhao, B., Liu, Z.: Video-xl-2: Towards very long-video understanding through task-aware kv sparsification. arXiv preprint arXiv:2506.19225 (2025)
34. Singh, A., Fry, A., Perelman, A., Tart, A., Ganesh, A., El-Kishky, A., McLaughlin, A., Low, A., Ostrow, A., Ananthram, A., et al.: Openai gpt-5 system card. arXiv preprint arXiv:2601.03267 (2025)
35. Tang, X., Qiu, J., Xie, L., Tian, Y., Jiao, J., Ye, Q.: Adaptive keyframe sampling for long video understanding. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 29118–29128 (2025)
36. Team, G., Georgiev, P., Lei, V.I., Burnell, R., Bai, L., Gulati, A., Tanzer, G., Vincent, D., Pan, Z., Wang, S., et al.: Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. arXiv preprint arXiv:2403.05530 (2024)
37. Team, K., Du, A., Yin, B., Xing, B., Qu, B., Wang, B., Chen, C., Zhang, C., Du, C., Wei, C., et al.: Kimi-vl technical report. arXiv preprint arXiv:2504.07491 (2025)
38. Tu, C., Zhang, L., Chen, P., Ye, P., Zeng, X., Cheng, W., Yu, G., Chen, T.: Favorbench: A comprehensive benchmark for fine-grained video motion understanding. arXiv preprint arXiv:2503.14935 (2025)
39. Wang, H., Lai, C., Sun, Y., Ge, W.: Weakly supervised gaussian contrastive grounding with large multimodal models for video question answering. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 5289–5298 (2024)
40. Wang, W., Gao, Z., Gu, L., Pu, H., Cui, L., Wei, X., Liu, Z., Jing, L., Ye, S., Shao, J., et al.: Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. arXiv preprint arXiv:2508.18265 (2025)
41. Wang, X., Hua, J., Lin, W., Zhang, Y., Zhang, F., Wu, J., Zhang, D., Nie, L.: Haic: Improving human action understanding and generation with better captions for multi-modal large language models. In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 10158–10181 (2025)

42. Wang, Y., Wang, Y., Wu, P., Liang, J., Zhao, D., Liu, Y., Zheng, Z.: Efficient temporal extrapolation of multimodal large language models with temporal grounding bridge. In: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing. pp. 9972–9987 (2024)
43. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
44. Xu, T., Dan, J.: Ehm: Exploring dynamic alignment and hierarchical clustering in unsupervised domain adaptation via high-order moment-guided contrastive learning. Neural Networks **185**, 107188 (2025)
45. Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K.R., Cao, Y.: React: Synergizing reasoning and acting in language models. In: The eleventh international conference on learning representations (2022)
46. Yu, S., Cho, J., Yadav, P., Bansal, M.: Self-chained image-language model for video localization and question answering. Advances in Neural Information Processing Systems **36**, 76749–76771 (2023)
47. Zhang, H., Gu, X., Li, J., Ma, C., Bai, S., Zhang, C., Zhang, B., Zhou, Z., He, D., Tang, Y.: Thinking with videos: Multimodal tool-augmented reinforcement learning for long video reasoning. arXiv preprint arXiv:2508.04416 (2025)
48. Zhang, X., Jia, Z., Guo, Z., Li, J., Li, B., Li, H., Lu, Y.: Deep video discovery: Agentic search with tool use for long-form video understanding. arXiv preprint arXiv:2505.18079 (2025)
49. Zitkovich, B., Yu, T., Xu, S., Xu, P., Xiao, T., Xia, F., Wu, J., Wohlhart, P., Welker, S., Wahid, A., et al.: Rt-2: Vision-language-action models transfer web knowledge to robotic control. In: Conference on Robot Learning. pp. 2165–2183. PMLR (2023)
50. Zuo, J., Deng, Y., Kong, L., Yang, J., Jin, R., Zhang, Y., Sang, N., Pan, L., Liu, Z., Gao, C.: Videolucy: Deep memory backtracking for long video understanding. arXiv preprint arXiv:2510.12422 (2025)