AgentVidBench: A Multi-Hop Video Question Answering Benchmark for Evaluating MLLM Agents

Seoyeon An [*1] **Hyeonseo Jang** [*1] **Minsu Kim** [*1] **Chanho Lee** [1] **Younghan Park** [1] **Kangwook Lee** [1]

Abstract

Autonomous MLLM agents are emerging to solve challenging real-world video understanding tasks that require multi-hop multimodal reasoning, yet there is a critical absence of video benchmarks equipped to rigorously evaluate their agentic capabilities. To bridge this gap, we introduce **AgentVidBench**, a multi-hop video question answering benchmark focused on evaluating the spatial, temporal, and causal reasoning capabilities of MLLM agents. Beyond standard question-answer pairs, AgentVidBench provides structured evidence milestones for trajectory evaluation, exposing trace-level failures hidden behind terminal success and providing actionable signals for improving agent behavior. Our trace-level diagnostics categorize these failures into five axes: Task Understanding, Evidence Coverage, Evidence Grounding, Evidence Completeness, and Reasoning Faithfulness. Experiments with 12 MLLMs and four agentic frameworks on AgentVidBench show that while agentic workflows improve standard accuracy, they still frequently succumb to premature fixation and incomplete evidence verification. We further present a simple yet effective agentic strategy that mitigates these failures through flexible, task-adaptive evidence-seeking in a single reasoning loop. Code and datasets are available at `https://github.com/krafton-ai/agentvidbench`.

1. Introduction

There has been a growing emphasis on the development of Multimodal Large Language Models (MLLMs) for video understanding, which harness the sophisticated reasoning capabilities of LLMs to decode and interpret diverse video signals (Alayrac et al., 2022; Zhang et al., 2023). Despite the remarkable potential of MLLMs, relying solely on their current architectural paradigms presents significant limitations in achieving human-level comprehension of complex video scenarios (Li et al., 2024; Fu et al., 2025; Mangalam et al., 2023). The integration of agentic frameworks represents a prominent strategy to mitigate the inherent structural bottlenecks (Yao et al., 2023; Yang et al., 2023; Shen et al., 2023). Unlike traditional single-pass models, agentic frameworks employ iterative reasoning, memory management, and external tool integration to interactively query and analyze video streams, thereby dynamically resolving spatio-temporal ambiguities (Wang et al., 2024; Yang et al., 2024). However, there is a critical absence of complex video benchmarks designed to assess the efficacy of multi-step agentic pipelines in solving compositional video understanding tasks.

To this end, we present **AgentVidBench**, a multi-hop video question answering benchmark specifically focused on evaluating the spatial, temporal, and causal video understanding capabilities of MLLM agents. We emphasize these three criteria as they reflect the necessary shift from passive, uniform video observation to active, interactive video exploration, akin to how humans naturally navigate and comprehend complex visual media. Specifically, solving complex video queries requires agents to explicitly zoom into specific regions for precise spatial reasoning, selectively focus on specific timeframes for temporal grounding, and actively traverse backward and forward across the timeline to infer causal links between disparate events. Crucially, beyond final answers, AgentVidBench includes structured evidence milestones that support reasoning trajectory analysis, enabling fine-grained diagnosis of where agentic reasoning breaks down and providing actionable insights for improving evidence acquisition and decision-making processes.

AgentVidBench comprises a meticulously curated set of 100 complex multiple-choice questions spanning 71 videos sourced from public web repositories. Specifically, the videos capture a broad spectrum of 12 fine-grained genres organized into 4 super-categories, with varying durations of up to 30 minutes. During the authoring process, we prohibited questions resolvable from a single frame or a localized timestamp lookup, ensuring each question is inherently compositional and demands a multi-step trajectory of active tool-use. Every question is cross-validated

[*]Equal contribution [1]KRAFTON. Correspondence to: Kangwook Lee <kangwooklee@krafton.com>.

ICML 2026 Workshop on Failure Modes in Agentic AI, Seoul, South Korea. Copyright 2026 by the author(s).

by multiple human evaluators to confirm a single correct answer. We then systematically generate 25 challenging distractors to form 26 options labeled A–Z, substantially reducing the chance of random guessing. Unlike existing benchmarks that evaluate only the final output, our benchmark uniquely features detailed, step-by-step solution traces for each question to explicitly facilitate the assessment of multi-step agentic workflows.

We evaluate various state-of-the-art MLLMs alongside their integration with agentic workflows using AgentVidBench. Our evaluation includes GPT-5 (OpenAI, 2026), Claude Opus 4.7 (Anthropic, 2026), Gemini 2.5 (Gemini Team, 2025), Qwen3-VL (Qwen Team, 2025), Qwen3.5 (Qwen Team, 2026), and Gemma 4 (Google DeepMind, 2026) for MLLMs, and STAR (Fan et al., 2025a), DVD (Zhang et al., 2025), and AVP (Wang et al., 2025b) for compatible agentic workflows. Experimental results on AgentVidBench demonstrate that conventional single-turn MLLMs struggle to solve multi-hop video QA tasks, whereas deploying them within state-of-the-art agentic workflows enhances their performance. However, our trajectory evaluation reveals that terminal accuracy frequently hides underlying trace-level failures where agent trajectories often lack accurate evidence grounding. To mitigate these trace-level failures, we suggest a simple yet effective agentic workflow that instantiates active evidence-seeking through controllable video inspection, serving as a strong baseline for future research.

Summary of Contributions: (1) We introduce AgentVidBench, a multi-hop video question answering benchmark designed to evaluate the spatial, temporal, and causal reasoning capabilities of MLLM agents; (2) We extensively benchmark state-of-the-art MLLMs both as standalone models and within agentic workflows on AgentVidBench, exposing trace-level failures beyond standard accuracy; (3) We propose a simple yet effective agentic strategy for fine-grained video navigation, and a novel evaluation pipeline to assess the quality of agent trajectories.

2. Related Work

Video Question Answering. Video Question Answering (Video QA) (Zhong et al., 2022) aims to generate accurate natural language responses to queries based on dynamic video content, requiring a comprehensive understanding of spatial, temporal, and causal relationships (Xu et al., 2017; Jang et al., 2017; Shang et al., 2021; Yang et al., 2021; Xiao et al., 2021; 2022). Moving beyond traditional feature-extraction methods, the paradigm has shifted toward transformer-based architectures and Multimodal Large Language Models (MLLMs) (Sun et al., 2021; Khurana & Deshpande, 2021; Zhou et al., 2024; Xiao et al., 2025; Tang et al., 2026). Recently, the field has witnessed a surge in powerful foundation models as both proprietary (e.g., GPT-4o (OpenAI, 2024), Gemini 1.5 (Reid et al., 2024)) and open-source models (e.g., Qwen-3-VL (Qwen Team, 2025), Gemma 4 (Google DeepMind, 2026)) now exhibit unprecedented capabilities in video understanding. However, they typically approach Video QA as a one-step generation process, fundamentally limiting their capacity to resolve complex queries that demand iterative, multi-step reasoning. To overcome this bottleneck, several studies focus on MLLM agents for video understanding by leveraging active temporal navigation (e.g., VideoAgent (Fan et al., 2024), Vgent (Shen et al., 2025)), hierarchical reasoning (e.g., VideoTree (Wang et al., 2025a), SiLVR (Zhang et al., 2026)), and iterative visual discovery (e.g., VideoLucy (Zuo et al., 2025), LVAgent (Chen et al., 2025), DVD (Zhang et al., 2025)). Despite their immense promise, the evaluation of these agentic systems is largely confined to traditional benchmarks comprising simple video queries (Wang et al., 2024; Kugo et al., 2025; Fan et al., 2025b; Dang et al., 2025; Montes & Lozano, 2026), which fail to adequately measure their autonomous planning and iterative reasoning capabilities.

Evaluation Benchmarks for MLLMs. To evaluate the growing capabilities of MLLMs, a variety of Video QA benchmarks have been established. Traditional Video QA datasets primarily focus on basic action recognition and object tracking within short video snippets (Xu et al., 2017; Yu et al., 2019). Subsequent efforts expanded the scope into diverse domains and specialized reasoning tasks, introducing datasets like MovieQA (Tapaswi et al., 2016) and TVQA (Lei et al., 2018) for narrative understanding, TGIF-QA (Jang et al., 2017) for spatio-temporal reasoning in short animations, and NExT-QA (Xiao et al., 2021) for causal and temporal explanations. Building on this foundation to address the necessity for even deeper cognitive evaluation, recent comprehensive benchmarks such as EgoSchema (Mangalam et al., 2023), MVBench (Li et al., 2024), Video-MME (Fu et al., 2025), and Video-MME-v2 (Fu et al., 2026) have been introduced. Although these benchmarking works only provide the performance of MLLMs, we evaluate both MLLMs and advanced MLLM agents on our complex multi-hop video question answering dataset, thereby explicitly quantifying the performance gap between passive perception and autonomous, multi-step reasoning. In addition, the inclusion of structured evidence milestones allows our benchmark to evaluate not only the final answer accuracy but also the intrinsic quality and logical coherence of the MLLM agent trajectories.

3. AgentVidBench

We introduce AgentVidBench, a benchmark designed to evaluate MLLM agents through 100 complex multiple-choice questions over 71 videos. These videos encompass a broad spectrum of 12 fine-grained genres organized into 4

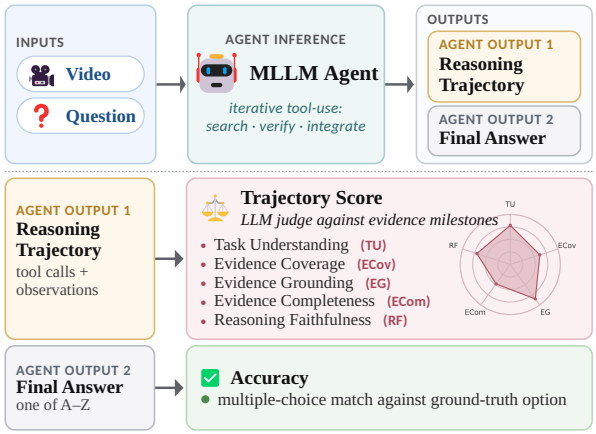


Figure 1. The evaluation pipeline of AgentVidBench. An agent produces a reasoning trajectory and final answer, which are evaluated using a trajectory score and accuracy, respectively. The trajectory score is computed by aggregating LLM-judge scores across five axes: Task Understanding, Evidence Coverage, Evidence Grounding, Evidence Completeness, and Reasoning Faithfulness.

super-categories, with durations ranging from a few seconds to several tens of minutes, thereby capturing a wide variety of dynamic visual contexts. Each question is paired with 26 answer options labeled A–Z, so that random guessing yields an expected accuracy of 3.85%. In the following sections, we describe the theoretical motivation of AgentVidBench (Section 3.1) and how the dataset including videos, questions, answer choices, milestones, and metadata was collected (Section 3.2). We also provide detailed statistics of the dataset (Section 3.3), followed by our comprehensive evaluation pipeline for assessing both standard accuracy and agent trajectory scores (Section 3.4).

3.1. Theoretical Motivation

Standard single-pass MLLMs inherit the fixed computational depth of their underlying transformer architectures. Consequently, these models possess a bounded capacity for sequential computation within a single inference pass.

Corollary 3.1 (Informal)**.** *For any fixed-depth transformer with bounded precision, width, and number of attention heads, there exists a sufficiently large k and a worst-case k-hop reasoning task that the model fundamentally cannot solve in a single forward pass.*

This corollary directly follows from Theorem 5.1 of Yao et al. (2025), and the formal statement and proof are provided in Appendix A. This limitation aligns with a broader body of literature studying the limits of fixed-depth transformers on compositional and multi-hop reasoning (Dziri et al., 2023; Yao et al., 2025; Yehudai et al., 2026). In the context of video understanding, we define a *hop* as a single dependent reasoning step or evidence-update operation, with k denoting the total number of such sequential steps required to resolve a complex query. Motivated by this theoretical bottleneck, we design AgentVidBench to explicitly evaluate models in scenarios where k is sufficiently large, demonstrating the strict necessity of transitioning from single-pass inference to iterative, agentic reasoning.

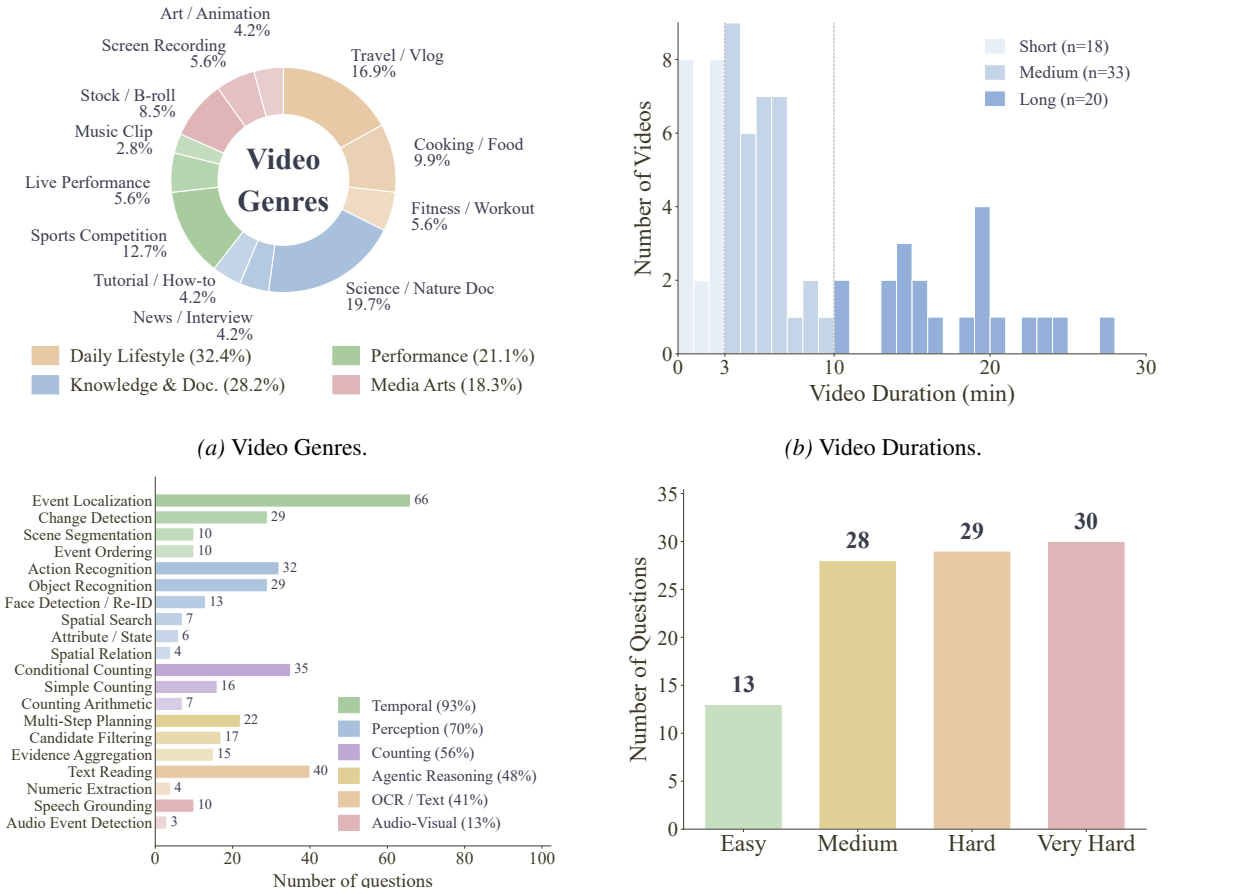
3.2. Benchmark Construction Pipeline

Video Collection. To ensure strict compliance with copyright regulations and guarantee the public availability of our benchmark, we source videos from both publicly accessible online sources and our own recordings. Among the 71 videos in our benchmark, externally sourced videos are drawn from platforms such as YouTube and Wikimedia Commons, as well as openly licensed media repositories, including archives maintained by government and public-sector organizations such as NASA. The full list of video sources and licenses is available in Appendix B.1.

Question-Answer Annotation. To guarantee the quality of our benchmark, all questions and answers are manually designed, written, and validated through a structured authoring workflow with five human annotators. We organize the authoring workflow into a sequential process of question annotation, answer annotation, and distractor generation, for which further details can be found in Appendix B.2.

Milestone Extraction. To address the limitations of standard final-answer evaluation, we introduce milestone extraction that facilitates the rigorous assessment of agent trajectories. We define a milestone as an essential evidence-acquisition or reasoning step required to justify the correct answer. Crucially, milestones are formulated to be tool-agnostic in that they specify *what* evidence is needed rather than *how* an agent should obtain it, ensuring a fair comparison across diverse agentic frameworks with varying toolsets or search paths. The milestone extraction process begins with annotators drafting a detailed ground-truth reasoning note that documents the human video investigation process to reach the answer. This descriptive note is subsequently abstracted into a structured set of evidence milestones. During this abstraction, each milestone is categorized by an evidence type, such as temporal localization, OCR/text reading, visual recognition, counting, exclusion checking, arithmetic, or verification. We provide more details on the milestone extraction process in Appendix B.3. Finally, these structured milestones serve as the reference evidence for the multi-step trajectory evaluation, which is detailed in Section 3.4.

Taxonomy Annotation. Each video and question in our benchmark is annotated with a structured set of taxonomy metadata to support fine-grained analysis. Video-level taxonomy axes include *genre*, *duration*, *production style*, and *audio type*, while question-level taxonomy axes include *required skills* and *difficulty*. We provide more details regarding the taxonomy annotation process and the complete list of taxonomy labels in Appendix B.4.



(a) Video Genres. *(b)* Video Durations.

(c) Required Skills. *(d)* Question Difficulty.

Figure 2. AgentVidBench at a glance. Panels (a) and (b) summarize video-level statistics, including genre composition and duration distribution. Panels (c) and (d) summarize question-level statistics, including required skills and difficulty. In (c), each question is multi-labeled with one or more skills, so capability-group percentages may sum to over 100%; bar colors indicate the corresponding top-level capability group.

3.3. Benchmark Statistics

We present the detailed statistics of AgentVidBench in Figure 2. At the video level, genres are hierarchically organized into four super-categories: *Daily Lifestyle*, *Knowledge & Documentary*, *Performance*, and *Media Arts*. The dataset encompasses short (< 3 minutes), medium (3–10 minutes), and long (> 10 minutes) video clips, providing a diverse temporal distribution to evaluate both local video perception and long-context reasoning.

In addition, questions are multi-tagged with 20 fine-grained sub-skills across six top-level capabilities: *temporal*, *perception*, *counting*, *agentic reasoning*, *OCR/text*, and *audio–visual*. Furthermore, the 100 questions lean heavily toward challenging scenarios, distributed into 13 `easy`, 28 `medium`, 29 `hard`, and 30 `very hard` cases. We note that these difficulty tiers are empirically calibrated using the observed performance of existing MLLMs and MLLM agents alongside initial human judgments.

3.4. Evaluation Pipeline

The AgentVidBench evaluation pipeline consists of two complementary branches: a final-answer branch and a reasoning trajectory branch, as illustrated in Figure 1. The final-answer branch measures standard accuracy by matching the agent's ultimate prediction against the ground-truth option. The reasoning trajectory branch evaluates each trajectory across five axes via an LLM judge, using the evidence milestones as the reference. Overall, the pipeline outputs final-answer accuracy, per-axis trajectory scores, and the normalized aggregated trajectory score (*Traj*).

Motivation for Trajectory Evaluation. Under final-answer accuracy, both a rigorously grounded deduction and a premature guess receive the same credit, even though they reflect substantially different reasoning processes. Even in our benchmark, where the 26-option multiple-choice format reduces the chance of random guessing, we observe cases where agents output the correct answer without covering the necessary evidence milestones. This distinction is espe-

cially important for agentic video QA, where failures often arise from poor long-range temporal grounding, incomplete video sweep, audio–visual misalignment, visual or textual misrecognition, and premature fixation under limited tool or context budgets. We thus evaluate trajectory-level quality beyond final-answer accuracy, drawing on MINERVA (Nagrani et al., 2025) while tailoring its evaluation axes to the agentic failures observed in AgentVidBench.

Failure Annotation and Clustering. To ground our trajectory evaluation in observed agent behavior, we analyze failure patterns from existing video-understanding baselines and agentic frameworks on AgentVidBench. Specifically, we examine trajectories from four methods, namely the single-turn baseline and three agentic frameworks (STAR, DVD, and AVP), paired with 20 MLLM configurations, including backbone and thinking-mode variants, yielding 8,000 trajectories in total. Through qualitative inspection of these trajectories, we first identify recurring failure phenomena and categorize them into a set of 21 fine-grained failure tags. We then group these tags into five higher-level failure clusters that characterize where agentic video investigations break down: *Goal Interpretation*, where the agent solves a different task than requested; *Evidence Localization*, where required evidence intervals are missed or the search stops too early; *Perception*, where visual, textual, audio, counting, or spatial evidence is misread; *Search Discipline*, where the agent commits before completing the required sweep; and *Inferential Faithfulness*, where the final answer does not follow from the collected evidence. The full tags and cluster definitions are provided in Appendix C.

Failure-grounded Trajectory Axes. The five failure clusters motivate the five process-quality axes used by the trajectory judge. Specifically, goal interpretation, evidence localization, perception, search discipline, and inferential faithfulness failures motivate *Task Understanding* (TU), *Evidence Coverage* (ECov), *Evidence Grounding* (EG), *Evidence Completeness* (ECom), and *Reasoning Faithfulness* (RF), respectively. These axes assess whether the agent understands the task constraints, acquires the necessary evidence, grounds its observations in the video, checks the completeness of the relevant temporal scope or candidate set when needed, and derives the answer consistently from the acquired evidence. However, this correspondence is not meant to be exclusive, in that a single failure can affect multiple axes, as search, perception, and reasoning errors often cascade through a trajectory. We thus evaluate the intended cluster–axis alignment empirically rather than assuming a perfectly separable mapping. As shown in Figure 3, each failure cluster has its strongest co-occurrence-controlled marginal effect on its intended axis, while off-diagonal effects remain. This validates the intended axis design while accounting for the coupled nature of agentic video failures.

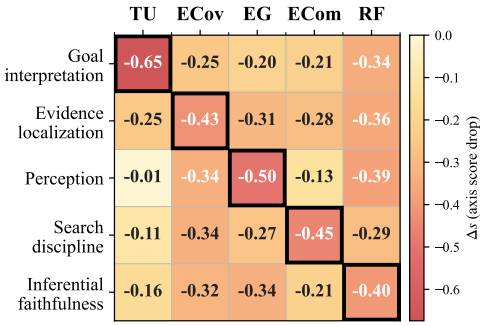


Figure 3. Cluster–axis alignment from co-occurrence-controlled regression. Each cell shows a failure cluster's isolated marginal effect on an axis score, after controlling for co-occurring tags. More negative values indicate larger score drops. Diagonal dominance supports the intended cluster–axis alignment.

Milestone-based Trajectory Scoring. To enable a fair comparison across agentic frameworks with different tool interfaces and search strategies, the trajectory branch compares each trajectory against tool-agnostic evidence milestones rather than a specific human solution path, as described in Appendix B.3. For each question, the judge receives the question, answer options, evidence milestones, ECom applicability metadata, and the agent trajectory, then assigns scores for TU, ECov, EG, ECom, and RF. The judge evaluates whether the trajectory contains, grounds, and uses the evidence needed to justify the answer, rather than rewarding the matching of a particular timestamp sequence or tool-use pattern. We finally report *Traj* as the normalized average of the applicable per-axis scores. This milestone-based protocol facilitates process-level diagnostics for agent failure modes while enabling fair, path-independent comparisons across diverse agentic frameworks. Additional details on the trajectory evaluation are provided in Appendix C.

4. Experiments

4.1. Settings

Models. To comprehensively evaluate diverse model families, we evaluate a broad suite of state-of-the-art MLLMs on our benchmark. The evaluated models are grouped into proprietary and open-source MLLMs. For proprietary MLLMs, we include GPT-5 (OpenAI, 2026), Claude Opus 4.7 (Anthropic, 2026), Gemini 2.5 Flash (Gemini Team, 2025), and Gemini 2.5 Pro (Gemini Team, 2025). For open-source MLLMs, we include Qwen3-VL-(4B, 8B) (Qwen Team, 2025), Qwen3.5-(9B, 27B) (Qwen Team, 2026), and Gemma 4-(E2B, E4B, 26B-A4B, 31B) (Google DeepMind, 2026). In the single-turn MLLM evaluation setting, models process the video and generate an answer in a single pass, without iterative interaction or tool use. Following prior work (Fu et al., 2026), we use 50 frames for single-turn evaluation of GPT-5 and Claude Opus 4.7 due to API cost and

context-window constraints, and adopt 1 fps sampling in all other settings. In the MLLM agent evaluation setting (Fan et al., 2025a; Zhang et al., 2025; Wang et al., 2025b), the models serve as the core reasoning modules within various agentic video understanding frameworks, coordinating multi-step reasoning processes and determining when and how each framework should invoke external tools.

Agentic Frameworks. To evaluate our benchmark across diverse agentic approaches, we consider three frameworks that organize video evidence and guide visual perception in different ways. Throughout the evaluation, we replace only the main agent while keeping the original tools and execution pipeline of each method unchanged, thereby isolating how agent-level planning shapes video inspection under the same setting. STAR (Fan et al., 2025a) represents structured scheduling, alternating temporal and spatial tool use to narrow question-relevant regions. DVD (Zhang et al., 2025) follows search-centric planning over multi-granular evidence, combining global browsing, clip retrieval, and frame inspection. AVP (Wang et al., 2025b) adopts active-perception planning, deciding what, where, and how to observe with reflection-based refinement. Together with our method (described next), this suite tests whether the benchmark remains effective across structured scheduling, search-centric retrieval, active observation, and task-adaptive planning.

Our Method. We introduce a task-adaptive reference agent (Ours) that retains the controllable inspection interface of AVP (Wang et al., 2025b) but restructures its orchestration around the failure clusters identified in Section 3.4. Our central observation is that many agentic video QA failures arise not from isolated reasoning errors, but from *evidence verification being delegated across an agent boundary*. In compartmentalized orchestrations such as AVP's Planner-Observer-Reflector pipeline, each component may reason correctly within its own scope, yet evidence handed across boundaries is rarely re-verified by the receiving component. A fluent but ungrounded observation is then accepted as confident evidence, and an under-specified sweep as sufficient. Figure 5 illustrates both failures on a counting question (Q100). Guided by this observation, we build on AVP with three design decisions targeting specific failure clusters:

(i) Unified reasoning loop for *Inferential Faithfulness*. We collapse AVP's separate planning, observation, and reflection stages into a single ReAct-style loop (Yao et al., 2023) where evidence acquisition, sufficiency judgment, and final answer extraction share a single reasoning context. This eliminates cross-component handoffs: the agent that decides the final answer is the same agent that gathered and verified the underlying evidence, so ungrounded observations and answer–evidence mismatches no longer propagate through an unchecked summary boundary.

(ii) Audio as a first-class evidence channel for *Evidence Localization* and *Perception*. Audio cues are easily diluted during extended visual inspection, costly to localize in raw video, and altogether inaccessible to MLLMs without native audio support. We therefore add a tool that retrieves pre-computed speech transcripts for either the full video or a specified time interval. This provides a dedicated audio evidence channel that the agent can query cheaply on demand alongside visual inspection.

(iii) Question-conditioned scanning policy for *Search Discipline* and *Evidence Localization*. Exhaustive questions such as counting and ordering demand fundamentally different temporal coverage strategies from localized ones such as moment-finding, but a single uniform sweep is rarely optimal for either. We therefore equip the agent with question-conditioned scanning guidelines—segment-by-segment coverage for exhaustive questions and focused windows for localized ones—together with explicit coverage feedback on the inspected temporal scope, which lets the agent decide when its sweep is complete. This addresses both premature commitment and missed required segments.

Ours serves as a simple but effective baseline showing that explicitly designing against the observed failure clusters can substantially improve trajectory quality, even without introducing fundamentally new capabilities.

Metrics. We evaluate the performance of both MLLMs and MLLM agents primarily using standard accuracy for multiple-choice questions, which directly measures the correctness of the final outcome. Since a correct final answer does not guarantee a sound reasoning process, we also evaluate trajectory scores for agentic workflows. This additional metric ensures that the final deduction is strictly grounded in a valid evidence-gathering process rather than mere chance.

4.2. Performance of Vanilla MLLMs and MLLM Agents

We compare the performance of vanilla MLLMs and MLLM agents on AgentVidBench in terms of accuracy and trajectory score in Table 1. First, the results of different open-source MLLM agents under the same agentic framework show that performance varies depending on the main agent, even with a fixed set of tools. Considering that the main agent does not directly observe the video and relies entirely on external tools for visual interpretation, these performance differences suggest that planning ability can substantially influence video understanding even under the same perceptual capacity. However, as in the case of STAR (Fan et al., 2025a), when the recognition capability of the external tools is insufficient, the agentic system can even underperform the single-turn performance of the vanilla MLLM. We further compare different frameworks and observe that performance tends to improve as the main agent is given greater flexibility in deciding what to observe and where to focus. For

Table 1. Performance comparison of proprietary and open-source MLLMs across different agentic workflows on the AgentVidBench dataset. We report both Accuracy (Acc.) and Trajectory (Traj.) scores for each configuration. Colored arrows next to Ours indicate absolute changes relative to the corresponding Single-Turn setting for the same model and metric.

Models	Single-Turn		STAR		DVD		AVP		Ours	
	Acc.	Traj.	Acc.	Traj.	Acc.	Traj.	Acc.	Traj.	Acc.	Traj.
Proprietary MLLMs										
GPT-5	0.29	0.55	0.14	0.33	0.34	0.60	0.41	0.65	**0.49** ↑0.20	**0.76** ↑0.21
Claude Opus 4.7	0.30	0.44	0.19	0.37	0.43	0.63	0.38	0.64	**0.54** ↑0.24	**0.79** ↑0.35
Gemini 2.5 Flash	**0.46**	0.66	0.10	0.29	0.27	0.47	**0.46**	0.65	0.45 ↓0.01	**0.71** ↑0.05
Gemini 2.5 Pro	0.51	0.71	0.13	0.35	0.30	0.47	0.56	0.71	**0.58** ↑0.07	**0.77** ↑0.05
Open-source MLLMs										
Qwen3-VL-4B	0.18	0.30	0.07	0.12	0.23	0.31	**0.49**	**0.67**	0.38 ↑0.20	0.65 ↑0.36
Qwen3-VL-8B	0.18	0.30	0.08	0.13	0.16	0.26	**0.41**	0.64	0.40 ↑0.22	**0.70** ↑0.41
Qwen3.5-9B	0.22	0.50	0.06	0.16	0.33	0.51	**0.47**	0.69	0.41 ↑0.19	**0.72** ↑0.22
Qwen3.5-27B	0.25	0.57	0.09	0.27	0.38	0.53	0.46	0.71	**0.50** ↑0.25	**0.77** ↑0.20
Gemma-4-E2B	0.07	0.25	0.08	0.19	0.30	0.28	**0.53**	**0.71**	0.38 ↑0.31	0.68 ↑0.43
Gemma-4-E4B	0.14	0.39	0.07	0.22	0.24	0.44	**0.46**	**0.67**	0.40 ↑0.26	**0.67** ↑0.28
Gemma-4-26B-A4B	0.22	0.43	0.10	0.24	0.37	0.56	0.52	0.72	**0.53** ↑0.31	**0.76** ↑0.33
Gemma-4-31B	0.37	0.61	0.10	0.29	0.40	0.58	**0.49**	0.71	0.48 ↑0.11	**0.75** ↑0.14

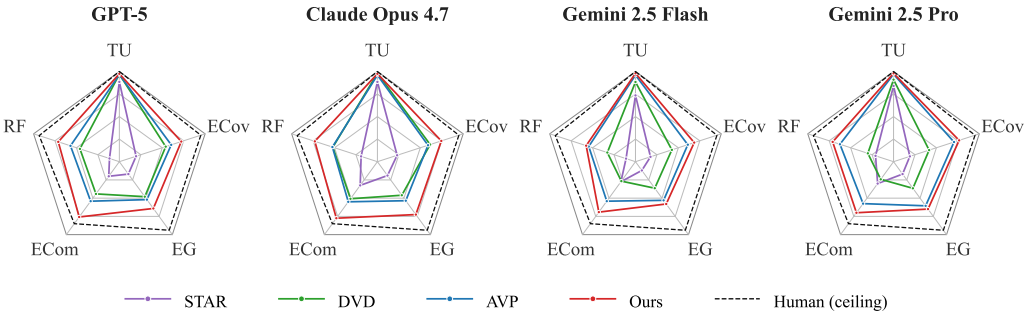


Figure 4. Trajectory score profiles of agentic frameworks. Each radar plot shows the scores across five trajectory evaluation dimensions: Task Understanding (TU), Evidence Coverage (ECov), Evidence Grounding (EG), Evidence Completeness (ECom), and Reasoning Faithfulness (RF). The dashed outline indicates the human reference score, where larger regions represent stronger trajectory quality.

example, although both AVP (Wang et al., 2025b) and our method employ Gemini 2.5 Pro as a video observation tool, our method achieves better performance by allowing the agent to actively select the observation target and location, rather than following the fixed plan–observe–reflect pipeline adopted by AVP. Overall, these results suggest that strong performance on AgentVidBench requires not only capable visual tools but also effective planning and adaptive observation strategies, making it a suitable benchmark for evaluating agentic frameworks in complex video understanding. Additional ablation study and efficiency analysis results are provided in Appendix E.

4.3. Analysis of Trajectory Score and Evaluation

Trajectory Score Analysis. We also report detailed trajectory scores across five dimensions in Figure 4. We first observe that TU remains consistently high across methods, indicating that MLLMs generally understand the task requirements, while the primary performance gaps emerge in evidence acquisition and the subsequent reasoning process. For example, STAR obtains low ECov and EG scores, indicating insufficient visual evidence acquisition, primarily due to the limited recognition capabilities of its external tools. This deficiency consequently lowers scores on the remaining axes, which depend on the acquired evidence. In contrast, DVD improves ECov and EG through multi-granular evidence search; however, its ECom and RF scores, which measure the reasoning process of deriving answers from evidence, remain relatively low. AVP and our method mitigate this limitation by coupling query-conditioned targeted observation with evidence reflection, enabling more faithful reasoning from the acquired evidence to the answer.

Anatomy of a Qualitative Example. Figure 5 is less instructive for what each system gets wrong than for *why* the architecture invites that failure. On **ECov**, the single-turn baseline hits a perception ceiling—uniform sampling at fixed resolution dilutes small objects in the pooled

Q100 · Orange Traffic Cones Between Two Landmarks

In this driving video, how many orange traffic cones are visible on the road between the moment the scooter passes the "Shekinah" building and the moment it passes the police car on the causeway? Count all visible orange traffic cones, including those that are part of or attached to other objects (e.g., signs).

A. 0. E. 4. H. 7. K. 10. N. 13. Q. 16. T. 19. W. 22. ... (26 options A–Z)

	ST	AVP	OURS
TU	✓	✓	✓
ECOV	~	~	✓
EG	~	✗	✓
ECOM	~	~	✓
RF	~	✗	✓
FINAL	✗	✗	✓

ECOV Evidence Coverage — *Did the trajectory acquire each reference milestone?*

ST ~ *"...A fourth cone is visible further down the road on the right side, past a parked white truck.* ***No other orange traffic cones are visible on the road within the specified segment of the video.****"*
— bounds M1, M2 found; M3 only partially covered (4 of 7 cones, sign-attached + utility-truck cluster missed).

OURS ✓ *"04:36: a single cone on the left...* ***07:13.5: a 'CAUTION MEN WORKING' sign... with an orange traffic cone placed on top of it.*** *07:16–07:17.5: five further cones around the utility truck."*
— every reference milestone enumerated with a stamp.

EG Evidence Grounding — *Do claimed observations match what is actually in the frames?*

AVP ✗ *"A line of* ***nineteen*** *orange traffic cones is visible along a curve on the causeway."*
— the pivotal observation is fabricated; no such 19-cone line exists in the video.

OURS ✓ *"07:13.5: a 'CAUTION MEN WORKING' sign...* ***with an orange traffic cone placed on top of it****. 07:16.5:* ***two more orange traffic cones*** *... next to the utility truck."*
— each claim grounds to an actual frame at the cited timestamp.

ECOM Evidence Completeness — *Was the temporal scope checked sufficiently for an exhaustive question?*

AVP ~ *Plan:* `load_mode=uniform, fps=1.0` ***over the full video****. Reflect:* `sufficient=true` *after* ***1 round, 0 replans****.*
— a sweep happened, but a single coarse pass was judged enough; granularity caused the miscount.

OURS ✓ *Round 1:* `analyze_video(fps=1, focus="locate Shekinah & police car")`*. Round 2:* ***`analyze_video(start=196, end=618, fps=2, focus="count cones")`****.*
— locate-then-focus: a second focused pass at higher fps over just the question's interval.

RF Reasoning Faithfulness — *Does the final answer follow from the evidence the agent actually acquired?*

AVP ✗ *"The total number of cones is the sum of these two groups:* ***2 + 19 = 21.****" → V*
— arithmetic is clean, but the chain rests on the unsupported "19": faithful to a fabricated premise, not to evidence.

OURS ✓ *"...a total of 7 orange traffic cones:* ***one at 4:36, one on a sign at 7:13, and a cluster of five cones around a work zone between 7:16 and 7:18.****" → H*
— each addend traces to an enumerated grounded observation; `1 + 1 + 5 = 7` maps to H.

Figure 5. Qualitative example of trajectory evaluation on AgentVidBench. The figure illustrates how the LLM judge evaluates agent trajectories across four axes (ECov, EG, ECom, RF), excluding Task Understanding (TU) which is consistently correct across methods. Compared to Single-Turn and AVP, our method improves evidence coverage, grounding, completeness, and reasoning faithfulness.

representation—which our agent breaks through by localizing the relevant interval first and then revisiting it at higher fps. On **EG**, AVP's Observer fabricates a fictitious cluster of cones that the Reflector accepts at face value: the Observer-to-Reflector interface is a one-way textual summary, so a fluent observation reads as confident evidence regardless of grounding, whereas our agent collapses this seam by extracting the answer within the same loop that gathers evidence. On **ECom**, the same trajectory halts after a single coarse pass because AVP gates halting on Reflector confidence rather than on coverage of the question's temporal scope, while our agent instead consumes a discrete coverage label and commits to a segment-by-segment sweep for counting questions. These failures cascade into **RF**: the arithmetic is locally correct but rests on an ungrounded premise, illustrating how faithfulness collapses silently when evidence verification is delegated across an agent boundary: each component reasons correctly within its own scope, yet no component verifies the evidence.

5. Conclusion

In this work, we introduce AgentVidBench, a multi-hop video question answering benchmark designed to evaluate the spatial, temporal, and causal reasoning abilities of MLLM agents. By combining challenging compositional questions with step-by-step evidence milestones, our benchmark enables evaluation beyond final-answer accuracy and supports fine-grained analysis of agent reasoning trajectories. Through extensive experiments, we show that existing single-turn MLLMs still struggle with multi-hop video understanding, while agentic workflows can consistently improve both answer accuracy and evidence-grounded reasoning quality. Given the growing interest in MLLM agents and multimodal video tasks, we believe AgentVidBench can serve as a valuable benchmark for future research on agentic video understanding, with its trajectory-level diagnostics guiding the development of agentic systems that not only reach correct answers but also faithfully acquire, ground, and reason over evidence.

References

Alayrac, J., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., Ring, R., Rutherford, E., Cabi, S., Han, T., Gong, Z., Samangooei, S., Monteiro, M., Menick, J. L., Borgeaud, S., Brock, A., Nematzadeh, A., Sharifzadeh, S., Binkowski, M., Barreira, R., Vinyals, O., Zisserman, A., and Simonyan, K. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022.

Anthropic. Introducing claude opus 4.7. https://www.anthropic.com/news/claude-opus-4-7, April 2026.

Chen, B., Yue, Z., Chen, S., Wang, Z., Liu, Y., Li, P., and Wang, Y. Lvagent: Long video understanding by multi-round dynamical collaboration of MLLM agents. *CoRR*, abs/2503.10200, 2025.

Dang, J., Song, H., Xiao, J., Wang, B., Peng, H., Li, H., Yang, X., Wang, M., and Chua, T. MUPA: towards multi-path agentic reasoning for grounded video question answering. *CoRR*, abs/2506.18071, 2025.

Dziri, N., Lu, X., Sclar, M., Li, X. L., Jiang, L., Lin, B. Y., Welleck, S., West, P., Bhagavatula, C., Bras, R. L., Hwang, J. D., Sanyal, S., Ren, X., Ettinger, A., Harchaoui, Z., and Choi, Y. Faith and fate: Limits of transformers on compositionality. In *NeurIPS*, 2023.

Fan, S., Cui, J., Guo, M.-H., and Yang, S. Tool-augmented spatiotemporal reasoning for streamlining video question answering task. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025a. URL https://openreview.net/forum?id=OFz4VDn0SO.

Fan, S., Guo, M., and Yang, S. Agentic keyframe search for video question answering. *CoRR*, abs/2503.16032, 2025b.

Fan, Y., Ma, X., Wu, R., Du, Y., Li, J., Gao, Z., and Li, Q. Videoagent: A memory-augmented multimodal agent for video understanding. In *ECCV (22)*, Lecture Notes in Computer Science, pp. 75–92. Springer, 2024.

Fu, C., Dai, Y., Luo, Y., Li, L., Ren, S., Zhang, R., Wang, Z., Zhou, C., Shen, Y., Zhang, M., Chen, P., Li, Y., Lin, S., Zhao, S., Li, K., Xu, T., Zheng, X., Chen, E., Shan, C., He, R., and Sun, X. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *CVPR*, pp. 24108–24118. Computer Vision Foundation / IEEE, 2025.

Fu, C., Yuan, H., Dong, Y., Zhang, Y.-F., Shen, Y., Hu, X., Li, X., Su, J., Long, C., Xie, X., et al. Video-mme-v2: Towards the next stage in benchmarks for comprehensive video understanding. *arXiv preprint arXiv:2604.05015*, 2026.

Gemini Team. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *CoRR*, abs/2507.06261, 2025.

Google DeepMind. Gemma 4: Byte for byte, the most capable open models. https://blog.google/innovation-and-ai/technology/developers-tools/gemma-4/, April 2026.

Jang, Y., Song, Y., Yu, Y., Kim, Y., and Kim, G. TGIF-QA: toward spatio-temporal reasoning in visual question answering. In *CVPR*, pp. 1359–1367. IEEE Computer Society, 2017.

Khurana, K. and Deshpande, U. Video question-answering techniques, benchmark datasets and evaluation metrics leveraging video captioning: A comprehensive survey. *IEEE Access*, 9:43799–43823, 2021.

Kugo, N., Li, X., Li, Z., Gupta, A., Khatua, A., Jain, N., Patel, C., Kyuragi, Y., Ishii, Y., Tanabiki, M., Kozuka, K., and Adeli, E. Videomultiagents: A multi-agent framework for video question answering. *CoRR*, abs/2504.20091, 2025.

Lei, J., Yu, L., Bansal, M., and Berg, T. L. TVQA: localized, compositional video question answering. In *EMNLP*, pp. 1369–1379. Association for Computational Linguistics, 2018.

Li, K., Wang, Y., He, Y., Li, Y., Wang, Y., Liu, Y., Wang, Z., Xu, J., Chen, G., Lou, P., Wang, L., and Qiao, Y. Mvbench: A comprehensive multi-modal video understanding benchmark. In *CVPR*, pp. 22195–22206. IEEE, 2024.

Mangalam, K., Akshulakov, R., and Malik, J. Egoschema: A diagnostic benchmark for very long-form video language understanding. In *NeurIPS*, 2023.

Montes, T. and Lozano, F. Viqagent: Zero-shot video question answering via agent with open-vocabulary grounding validation. In *VISAPP (2)*, pp. 79–90. SCITEPRESS, 2026.

Nagrani, A., Menon, S., Iscen, A., Buch, S., Mehran, R., Jha, N., Hauth, A., Zhu, Y., Vondrick, C., Sirotenko, M., Schmid, C., and Weyand, T. MINERVA: evaluating complex video reasoning. *CoRR*, abs/2505.00681, 2025.

OpenAI. Gpt-4o system card. *CoRR*, abs/2410.21276, 2024.

OpenAI. Openai GPT-5 system card. *CoRR*, abs/2601.03267, 2026.

Qwen Team. Qwen3-vl technical report. *CoRR*, abs/2511.21631, 2025.

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

Reid, M., Savinov, N., Teplyashin, D., Lepikhin, D., Lillicrap, T. P., Alayrac, J., Soricut, R., Lazaridou, A., Firat, O., Schrittwieser, J., Antonoglou, I., Anil, R., Borgeaud, S., Dai, A. M., Millican, K., Dyer, E., Glaese, M., Sottiaux, T., Lee, B., Viola, F., Reynolds, M., Xu, Y., Molloy, J., Chen, J., Isard, M., Barham, P., Hennigan, T., McIlroy, R., Johnson, M., Schalkwyk, J., Collins, E., Rutherford, E., Moreira, E., Ayoub, K., Goel, M., Meyer, C., Thornton, G., Yang, Z., Michalewski, H., Abbas, Z., Schucher, N., Anand, A., Ives, R., Keeling, J., Lenc, K., Haykal, S., Shakeri, S., Shyam, P., Chowdhery, A., Ring, R., Spencer, S., Sezener, E., and et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *CoRR*, abs/2403.05530, 2024.

Shang, X., Li, Y., Xiao, J., Ji, W., and Chua, T. Video visual relation detection via iterative inference. In *ACM Multimedia*, pp. 3654–3663. ACM, 2021.

Shen, X., Zhang, W., Chen, J., and Elhoseiny, M. Vgent: Graph-based retrieval-reasoning-augmented generation for long video understanding. *CoRR*, abs/2510.14032, 2025.

Shen, Y., Song, K., Tan, X., Li, D., Lu, W., and Zhuang, Y. Hugginggpt: Solving AI tasks with chatgpt and its friends in hugging face. In *NeurIPS*, 2023.

Sun, G., Liang, L., Li, T., Yu, B., Wu, M., and Zhang, B. Video question answering: a survey of models and datasets. *Mob. Networks Appl.*, 26(5):1904–1937, 2021.

Tang, Y., Bi, J., Xu, S., Song, L., Liang, S., Wang, T., Zhang, D., An, J., Lin, J., Zhu, R., Vosoughi, A., Huang, C., Zhang, Z., Liu, P., Feng, M., Zheng, F., Zhang, J., Luo, P., Luo, J., and Xu, C. Video understanding with large language models: A survey. *IEEE Trans. Circuits Syst. Video Technol.*, 36(2):1355–1376, 2026.

Tapaswi, M., Zhu, Y., Stiefelhagen, R., Torralba, A., Urtasun, R., and Fidler, S. Movieqa: Understanding stories in movies through question-answering. In *CVPR*, pp. 4631–4640. IEEE Computer Society, 2016.

Wang, X., Zhang, Y., Zohar, O., and Yeung-Levy, S. Videoagent: Long-form video understanding with large language model as agent. In *ECCV (80)*, Lecture Notes in Computer Science, pp. 58–76. Springer, 2024.

Wang, Z., Yu, S., Stengel-Eskin, E., Yoon, J., Cheng, F., Bertasius, G., and Bansal, M. Videotree: Adaptive tree-based video representation for LLM reasoning on long videos. In *CVPR*, pp. 3272–3283. Computer Vision Foundation / IEEE, 2025a.

Wang, Z., Zhou, H., Wang, S., Li, J., Xiong, C., Savarese, S., Bansal, M., Ryoo, M. S., and Niebles, J. C. Active video perception: Iterative evidence seeking for agentic long video understanding, 2025b. URL https://arxiv.org/abs/2512.05774.

Xiao, J., Shang, X., Yao, A., and Chua, T. Next-qa: Next phase of question-answering to explaining temporal actions. In *CVPR*, pp. 9777–9786. Computer Vision Foundation / IEEE, 2021.

Xiao, J., Yao, A., Liu, Z., Li, Y., Ji, W., and Chua, T. Video as conditional graph hierarchy for multi-granular question answering. In *AAAI*, pp. 2804–2812. AAAI Press, 2022.

Xiao, J., Huang, N., Qin, H., Li, D., Li, Y., Zhu, F., Tao, Z., Yu, J., Lin, L., Chua, T., and Yao, A. Videoqa in the era of llms: An empirical study. *Int. J. Comput. Vis.*, 133(7): 3970–3993, 2025.

Xu, D., Zhao, Z., Xiao, J., Wu, F., Zhang, H., He, X., and Zhuang, Y. Video question answering via gradually refined attention over appearance and motion. In *ACM Multimedia*, pp. 1645–1653. ACM, 2017.

Yang, X., Feng, F., Ji, W., Wang, M., and Chua, T. Deconfounded video moment retrieval with causal intervention. In *SIGIR*, pp. 1–10. ACM, 2021.

Yang, Z., Li, L., Wang, J., Lin, K., Azarnasab, E., Ahmed, F., Liu, Z., Liu, C., Zeng, M., and Wang, L. MM-REACT: prompting chatgpt for multimodal reasoning and action. *CoRR*, abs/2303.11381, 2023.

Yang, Z., Chen, G., Li, X., Wang, W., and Yang, Y. Doraemongpt: Toward understanding dynamic scenes with large language models (exemplified as A video agent). In *ICML*, Proceedings of Machine Learning Research, pp. 55976–55997. PMLR / OpenReview.net, 2024.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K. R., and Cao, Y. React: Synergizing reasoning and acting in language models. In *ICLR*. OpenReview.net, 2023.

Yao, Y., Du, Y., Zhu, D., Hahn, M., and Koller, A. Language models can learn implicit multi-hop reasoning, but only if they have lots of training data. In *EMNLP*, pp. 9684–9702. Association for Computational Linguistics, 2025.

Yehudai, G., Amsel, N., and Bruna, J. Compositional reasoning with transformers, rnns, and chain of thought, 2026. URL https://arxiv.org/abs/2503.01544.

Yu, Z., Xu, D., Yu, J., Yu, T., Zhao, Z., Zhuang, Y., and Tao, D. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *AAAI*, pp. 9127–9134. AAAI Press, 2019.

Zhang, C., Lin, Y., Wang, Z., Bansal, M., and Bertasius, G. Silvr: A simple language-based video reasoning framework. *Trans. Mach. Learn. Res.*, 2026, 2026.

Zhang, H., Li, X., and Bing, L. Video-llama: An instruction-tuned audio-visual language model for video understanding. In *EMNLP (Demos)*, pp. 543–553. Association for Computational Linguistics, 2023.

Zhang, X., Jia, Z., Guo, Z., Li, J., Li, B., Li, H., and Lu, Y. Deep video discovery: Agentic search with tool use for long-form video understanding. *CoRR*, abs/2505.18079, 2025.

Zhong, Y., Ji, W., Xiao, J., Li, Y., Deng, W., and Chua, T. Video question answering: Datasets, algorithms and challenges. In *EMNLP*, pp. 6439–6455. Association for Computational Linguistics, 2022.

Zhou, P., Wang, L., Liu, Z., Hao, Y., Hui, P., Tarkoma, S., and Kangasharju, J. A survey on generative AI and LLM for video generation, understanding, and streaming. *CoRR*, abs/2404.16038, 2024.

Zuo, J., Deng, Y., Kong, L., Yang, J., Jin, R., Zhang, Y., Sang, N., Pan, L., Liu, Z., and Gao, C. Videolucy: Deep memory backtracking for long video understanding. *CoRR*, abs/2510.12422, 2025.

A. Appendix – Theoretical Details

A.1. Formal Fixed-Depth Consequence

The informal corollary in the main text states that a fixed-depth single-pass transformer cannot solve arbitrarily deep worst-case multi-hop reasoning tasks in a single forward pass. We formalize this statement by directly applying Theorem 5.1 of Yao et al. (2025), using their k-hop relation-composition setting and assumptions.

Corollary A.1 (Formal version of Corollary 3.1)**.** *Consider any fixed causal transformer architecture with L layers, H attention heads, hidden dimension d, and p-bit precision. Under the query-independent-attention assumption in Theorem A.2, there exists a sufficiently large hop count k, a sufficiently large entity universe E, and a relation set $\mathcal{R} \subseteq E^E$ such that this transformer cannot solve the corresponding worst-case k-hop relation-composition task in a single forward pass.*

This corollary follows immediately from the following known lower bound.

A.2. Known k-Hop Depth Lower Bound

The theorem below is Theorem 5.1 of Yao et al. (2025), stated here for completeness. It formalizes a worst-case lower bound for implicit k-hop relation composition in a single transformer forward pass.

Theorem A.2 (Worst-case k-hop lower bound)**.** *Let E be an entity universe and let $k \leq |E| - 2$. Then there exists a relation set $\mathcal{R} \subseteq E^E$ such that the following holds.*

Consider a causal transformer operating with p-bit precision, hidden dimension d, H attention heads, and L layers. Suppose the model solves the k-hop relation-composition task over E and $\mathcal{R}$, where the target answer is

$$r_k \circ r_{k-1} \circ \cdots \circ r_1(e).$$

Assume that the attention patterns do not depend on the query e. These attention patterns may still depend on the relation sequence $r_1, \ldots, r_k$. Then

$$L \geq \frac{k}{8pdH}.$$

Reference. This is Theorem 5.1 of Yao et al. (2025), restated in our notation. □

A.3. Proof of Corollary A.1

Proof. Fix the given transformer architecture, and let L, p, d, H denote its depth, precision, hidden dimension, and number of attention heads. Choose an integer k such that

$$k > 8pdHL.$$

Then choose an entity universe E large enough that

$$k \leq |E| - 2.$$

By Theorem A.2, there exists a relation set $\mathcal{R} \subseteq E^E$ such that solving the corresponding k-hop relation-composition task requires

$$L \geq \frac{k}{8pdH}.$$

However, by our choice of k,

$$\frac{k}{8pdH} > L.$$

Thus the fixed L-layer transformer does not have sufficient depth to solve this worst-case k-hop relation-composition task in a single forward pass under the stated assumption. This proves the corollary. □

A.4. Interpretation and Implication for Multi-Hop Video QA

Corollary A.1 is the formal consequence we need for our benchmark motivation: for any fixed transformer architecture, there exists a worst-case k-hop relation-composition task that cannot be solved in a single forward pass once k is sufficiently large. The supporting lower bound can be written as

$$LpdH = \Omega(k),$$

where L is the number of transformer layers, while p, d, H capture per-layer capacity through numerical precision, hidden dimension, and number of attention heads. Thus, the result should be read as a width–depth–precision tradeoff: increasing per-layer capacity can partially compensate for depth, but for a fixed architecture under single-pass inference—without externalized intermediate states, recurrence, or other adaptive test-time computation—the required depth grows with the number of composed relations.

The formal relation-composition task is not itself a theorem about video QA. Rather, we use it as a theoretical abstraction of a broader computational bottleneck: fixed-depth single-pass transformers have bounded internal sequential computation, and sufficiently deep dependent reasoning can exceed this budget. This perspective motivates our benchmark design.

In video QA, the analogous difficulty arises when answering a question requires a sequence of dependent evidence updates. A model may need to acquire evidence from one part of the video, use that evidence to guide the search for another part, verify a relation between objects or events, and then update its answer based on the new evidence. A single-pass VLM must compress this entire process into one fixed-depth forward pass. Following this motivation, we construct AgentVidBench around this regime.

B. Appendix – AgentVidBench Construction Details

This appendix expands on Section 3.2 with the concrete configuration of every stage used to build AgentVidBench. The complete benchmark is hosted at `https://huggingface.co/datasets/agentvidbench/agentvidbench`.

B.1. Video Collection

AgentVidBench contains 71 videos collected from publicly accessible or author-owned sources. Publicly sourced videos are primarily drawn from YouTube and Wikimedia Commons, with additional clips from NASA, DVIDS, Vimeo, Current Biology, and the US National Park Service. We also include four self-recorded videos. Table 2 summarizes the distribution of source platforms.

For each video, we manually checked the license or reuse terms before inclusion. All videos were verified to be covered by Creative Commons licenses, public-domain status, or author ownership permitting release with the benchmark package. Table 3 summarizes the resulting license distribution.

Table 2. Distribution of video sources in AgentVidBench.

Source platform	# Videos
YouTube	44
Wikimedia Commons	10
NASA	8
Self Recorded	4
DVIDS (US DoD)	2
Current Biology	1
Vimeo	1
US National Park Service	1
Total	**71**

B.2. Question-Answer Annotation

To guarantee the quality of our benchmark, all questions and answers are manually designed, written, and validated through a structured authoring workflow with five human annotators. We organize the sequential authoring workflow by functional roles as follows:

Table 3. Distribution of video licenses in AgentVidBench.

License	# Videos
YouTube CC BY (reuse allowed)	43
Public Domain	14
CC BY-NC 4.0	4
CC BY-SA 4.0	4
CC BY 4.0	3
CC BY-SA 3.0	2
CC BY 3.0	1
Total	**71**

- **Question Annotation**: Human annotators manually formulate questions for each video through comprehensive and iterative reviews of the video content. In the authoring process, annotators seek to cover a diverse set of skills spanning perception, counting, temporal and spatial reasoning, audio understanding, and multi-step agentic reasoning. Each question is written to necessitate active video exploration, such as revisiting specific moments or aggregating evidence across distant timestamps, while strictly maintaining a single verifiable correct answer.
- **Answer Annotation**: To ensure rigorous quality control, the answer to each question is determined by the absolute consensus of independent human annotators. Specifically, each question is initially answered by three annotators, and they compare their responses with each other. If their responses align perfectly, the consensus answer is finalized as the ground truth. However, if they identify potential issues such as answer discrepancies or ambiguous phrasing, the questions for these flagged instances are refined until agreement on a singular correct answer is achieved. As an additional validation step, we analyze preliminary model responses to uncover latent edge cases not anticipated during human review.
- **Distractor Generation**: Following the annotation of the correct answer, we generate 25 distractors for each question based on the answer type. For example, numeric type answers are paired with nearby integer values that span a plausible range, while non-numeric type answers are paired with similar words to the correct answer or relevant candidates visible in the video. To prevent models from exploiting structural shortcuts, the candidate set may also include assertions of unanswerability (e.g., "the requested name is not mentioned") or threshold conditions (e.g., "26+"), strategically serving as either the actual ground truth or deceptive distractors. We note that the generated distractors avoid semantic ambiguity, partial validity, and trivial identifiability.

B.3. Milestone Extraction Details

Final-answer accuracy alone cannot determine whether an agent actually acquired the video evidence needed to justify its answer. To evaluate the agent's investigation process, annotators manually write a detailed ground-truth reasoning note for each question, describing how a human would investigate the video to reach the answer. We then abstract these notes into structured evidence milestones, where each milestone specifies a tool-agnostic evidence or reasoning step needed to support the answer.

Milestones are intentionally more abstract than raw solution trajectories. Rather than prescribing a particular tool sequence, timestamp, frame index, or navigation path, they define the underlying evidence requirements of a question. This allows agents with different interfaces or search strategies to be judged against the same evidence reference.

B.3.1. Milestone Schema

Each milestone consists of a stable identifier, an evidence type, and a tool-agnostic natural-language description. The identifier is used to align milestone-level coverage judgments, the type indicates the kind of evidence or reasoning step required, and the description states what evidence should be acquired.

B.3.2. Tool-agnostic Annotation Principles

Milestones are written to specify *what* evidence is needed, not *how* the evidence should be obtained. This is necessary because agentic video frameworks vary in their tool interfaces and interaction granularity: some use low-level controls such as seeking and frame-rate adjustment, others use high-level video-analysis calls, and zero-shot baselines may receive the video in a single pass. Comparing raw tool traces would therefore privilege one interface or search strategy over another.

To avoid encoding path-specific cues, milestone descriptions remove or abstract away narrow timestamps, exact time ranges, tool names, occurrence timestamp lists, and solution-specific decomposition details from the original human reasoning note. For example, a human explanation may mention that a clue appears at a particular timestamp, but the milestone should instead describe the evidence to be found at that moment. Similarly, for counting questions, milestones should not enumerate every timestamped occurrence; they should describe the counting target and any relevant exclusion condition. Table 4 gives representative examples of path-specific cues and their tool-agnostic milestone counterparts.

Table 4. Examples of path-specific cues and their tool-agnostic milestone counterparts.

Path-specific cue	Path-specific reference	Tool-agnostic milestone
Tool name	Use the OCR tool to read the model code.	Read the model code on the bezel as `CF-19`.
Specific timestamp	At 02:13, inspect the laptop screen.	Locate the close-up moment where the laptop screen and bezel text are clearly visible.
Fixed time range	The relevant evidence is in the 00:20–00:50 segment.	Locate the driving segment where the relevant event is visible.
Timestamp list	Check the moments at 2:49, 2:50, 3:36, and 3:51.	Identify all candidate moments showing the target action.
Exact arithmetic chain	Sum the counts as `4+2+1+7+10+2+5+3+3`.	Sum the extracted counts to obtain the final total.

B.3.3. Milestone Statistics

Rather than prescribing full solution trajectories, the milestone representation provides a compact evidence checklist. Each question is represented by 3–7 milestones, with most questions containing four or five evidence units. Table 5 summarizes the number of milestones per question.

Table 6 shows that temporal localization is the most frequent milestone type, followed by visual recognition, exclusion checking, counting, context identification, and OCR/text reading. This distribution reflects the benchmark's emphasis on locating relevant evidence over time and combining it with perceptual, textual, counting, or exclusion-based reasoning steps.

Table 5. Number of evidence milestones per question in AgentVidBench.

# Milestones	# Questions	%
3	14	14.0
4	32	32.0
5	40	40.0
6	12	12.0
7	2	2.0

Table 6. Distribution of evidence milestone types. Percentages are computed over 456 milestones.

Milestone type	Occurrences	%
Temporal localization	113	24.8
Visual recognition	69	15.1
Exclusion check	59	12.9
Counting	54	11.8
Context identification	43	9.4
OCR/text reading	41	9.0
Verification	29	6.4
Audio-visual evidence	28	6.1
Comparison	11	2.4
Arithmetic	9	2.0

B.3.4. REFERENCE VALIDATION

We use milestone-only references as the default input to the trajectory judge. To validate this choice, we compare judge scores obtained with milestone-only references against scores obtained with the original human answer explanations. As shown in Table 7, the two references produce strongly correlated process-quality scores across the five axes, suggesting that milestones preserve the core evidence information needed for trajectory evaluation.

At the same time, milestone references substantially reduce path-specific constraints. Human explanations often contain timestamps, time ranges, timestamp lists, or solution-specific arithmetic decompositions. Directly using them as judge references can reward trajectories that follow the same human solution path and penalize different but valid search strategies. Table 8 shows that milestone references preserve the judging signal while removing most path-specific cues, making them a more suitable default for evaluating agents that may reach the same evidence through different valid paths.

Table 7. Correlation between human-reference and milestone-reference judge scores.

Process-quality axis	Pearson r
Task understanding	0.656
Ground-truth evidence coverage	0.756
Evidence grounding	0.814
Evidence completeness	0.746
Reasoning faithfulness	0.837
Mean	0.762

Table 8. Path-specific cues detected in human explanations and milestone references.

Reference input	Affected Qs	Tool-agnostic rate
Human answer explanation	85/100	15.0%
Evidence milestones	1/100	99.0%

B.4. Taxonomy Labels

We annotate each video and question with taxonomy metadata to support fine-grained analysis of benchmark composition and model performance. Taxonomy labels with deterministic criteria, such as *duration*, are assigned based on predefined rules, whereas *production style* and *audio type* are determined via manual review. For *genre* and *required skills*, we utilize a human-in-the-loop approach where MLLM-generated initial labels are rigorously verified by human annotators. Finally, *difficulty* is initially estimated by humans and then empirically calibrated using the observed performance of the evaluated MLLMs and agentic frameworks to mitigate subjective bias.

Table 9 summarizes the taxonomy axes, whether they are assigned at the video or question level, and their label cardinality. Video-level metadata describes the content genre, duration, production style, and audio characteristics of each source video, while question-level metadata describes the skills required to answer the question and its difficulty.

Video genre and required skills are organized hierarchically. Table 10 lists the four video-genre super-categories and their corresponding fine-grained genres. Production style, audio type, and required skills are multi-label axes, since a video or question may involve multiple styles, modalities, or skills. Their label sets are provided in Tables 11 and 12.

C. Appendix – Evaluation Pipeline

As illustrated in Figure 1, AgentVidBench evaluates each system output along two complementary branches. The first branch evaluates the final answer by matching the selected option against the ground-truth answer. This provides the standard multiple-choice accuracy metric. The second branch evaluates the reasoning trajectory produced before the final answer, using an LLM judge to assess whether the system acquired, grounded, and integrated the video evidence needed to justify its prediction.

The motivation for the second branch is that final-answer accuracy alone does not reveal how the answer was obtained. Two systems may select the same option while following very different processes: one may localize the relevant moments, verify

Table 9. Overview of taxonomy axes in AGENTVIDBENCH.

Level	Axis	# Labels	Description
Video	Genre	12	Content genre, grouped into 4 super-categories.
	Duration	3	Short, medium, or long duration bin.
	Production style	14	Camera, editing, and presentation style.
	Audio type	5	Dominant audio characteristics.
Question	Required skills	20	Multi-label skills, grouped into 6 capabilities.
	Difficulty	4	Question difficulty label.

Table 10. Video genre taxonomy.

Super-category	Description
Daily Lifestyle	Everyday activity-oriented videos. *Genres:* Travel/Lifestyle Vlog, Cooking/Food, Fitness/Workout.
Knowledge & Documentary	Informational, educational, or documentary-style videos. *Genres:* Science/Nature Documentary, News/Interview, Tutorial/How-to.
Performance	Event- or performance-centered videos. *Genres:* Sports Competition, Live Performance, Music Clip.
Media Arts	Screen-based, artistic, or contextual visual content. *Genres:* Screen Recording, Art/Animation, Stock/B-roll.

Table 11. Production-style and audio-type labels. Labels are multi-label at the video level.

Taxonomy	Labels
Production style	scene-cut edit, static camera, handheld camera, single continuous shot, drone/aerial, tracking camera, animation, screen capture, subtitle present, split screen, timelapse, text-overlay heavy, multicam switch, slow motion.
Audio type	speech, music, ambient sound, sound effects, silent.

Table 12. Required-skill taxonomy. Labels are multi-label at the question level.

Capability	Description	Fine-grained skills
Temporal	Reasoning about time, sequence, or scene structure of the video.	event localization, event ordering, scene segmentation, change detection.
Perception	Frame-level visual recognition of entities, attributes, or activities.	object recognition, action recognition, face detection / re-identification, attribute/state recognition, spatial relation, spatial search.
Counting	Quantifying occurrences across the video, possibly under conditions or with arithmetic over multiple counts.	simple counting, conditional counting, counting arithmetic.
Agentic reasoning	Higher-order skills: planning multi-step investigations, filtering candidate answers, or aggregating evidence across observations.	multi-step planning, candidate filtering, evidence aggregation.
OCR/Text	Reading textual or numeric content rendered visually in the frame.	OCR/text reading, numeric value extraction.
Audio-visual	Tasks that require integrating audio cues with visual content.	speech grounding, audio event detection.

distractors, and derive the answer from grounded observations, while another may rely on incomplete evidence or commit prematurely. We therefore evaluate trajectory-level process quality in addition to answer correctness.

For a given question, the trajectory judge receives the question text, answer options, a pre-classified sweep-applicability flag, pre-computed evidence milestones, and the rendered agent trajectory. The gold answer and externally parsed predicted

answer letter are not provided to the judge, preventing answer label leakage. Final-answer correctness is computed separately by exact multiple-choice matching. The original human answer explanation is not provided; instead, the judge uses milestone-only references, as described in Appendix B.3. This design evaluates whether the trajectory recovers the underlying evidence needed for the answer, rather than whether it follows the same timestamp sequence or tool-use path as the human-written solution.

Judge Input Construction. Evaluated systems produce different kinds of intermediate outputs. Agentic frameworks may generate structured tool-use trajectories with calls, observations, and intermediate reasoning, whereas non-agentic baselines may produce only a direct textual response. Before judging, we render each prediction into a common text format containing the predicted final answer and all available intermediate observations, retrieved evidence, tool calls, and reasoning traces. This normalization allows the same judge prompt to be applied across heterogeneous systems without assuming a shared tool interface.

Failure Annotation and Clustering. To ground the trajectory axes in observed agent behavior, we analyze failure patterns from existing video-understanding baselines and agentic frameworks on AgentVidBench. We examine trajectories from 4 methods—the Single-Turn baseline and the 3 agentic frameworks STAR, DVD, and AVP—paired with 20 MLLM backbones, yielding 8,000 trajectories in total. Through qualitative inspection, we identify recurring failure phenomena and encode them as a closed vocabulary of 21 fine-grained failure tags. We then group these tags into five higher-level failure clusters that characterize where agentic video investigations break down. Table 13 lists the full tag vocabulary, cluster definitions, and the corresponding process-quality axis.

Process-quality Axes. We define the five process-quality axes by analyzing recurring failure modes observed in MLLM and MLLM-agent trajectories. These failures cluster around five stages of the investigation process: understanding the task, acquiring the required evidence, grounding observations in the video, checking whether the evidence search is complete, and faithfully deriving the answer from the acquired evidence. The trajectory judge therefore assigns scores along five axes: Task Understanding (TU), Ground-truth Evidence Coverage (ECov), Evidence Grounding (EG), Evidence Completeness (ECom), and Reasoning Faithfulness (RF). Each applicable axis is scored on a 0–2 scale, where 0 indicates a severe failure, 1 indicates partial success, and 2 indicates strong performance.

- **Task Understanding (TU)**: Measures whether the agent correctly understands the question, including the target entity, constraints, exclusion rules, temporal scope, and answer format. This axis captures failures such as solving for the wrong object, ignoring a qualifier, or misinterpreting "first", "last", "only", or similar constraints.
- **Ground-truth Evidence Coverage (ECov)**: Measures whether the trajectory acquires the evidence specified by the reference milestones. This axis focuses on whether the necessary evidence is present in the trajectory, rather than whether the final option happens to be correct.
- **Evidence Grounding (EG)**: Measures whether the observations claimed by the agent match the reference evidence. This includes OCR values, object identities, counts, visual attributes, spatial relations, audio cues, and other intermediate observations. An agent may search in the right place but still receive a low EG score if it misreads or misrecognizes the evidence.
- **Evidence Completeness (ECom)**: Measures whether the agent sufficiently checks the relevant temporal scope or candidate set when the question requires evidence-completeness verification. This axis is especially important for counting, ordering, first/last, absence, or exhaustive-search questions. If a question is locally answerable from a single anchored observation, ECom is marked as not applicable and excluded from the applicable-axis mean.
- **Reasoning Faithfulness (RF)**: Measures whether the final answer follows consistently from the evidence acquired in the trajectory. This axis captures errors such as unsupported leaps, arithmetic mistakes, contradictions between observations and conclusion, or incorrect option mapping.

Applicability of Evidence Completeness (ECom). Evidence Completeness (ECom) is applied only when a question requires evidence-completeness verification, such as checking the full temporal scope or multiple candidate instances before committing to an answer. Such questions include counting tasks, total or aggregate questions, first/last or Nth-instance questions, ordering/ranking questions, and cases where absence or exclusion must be verified across the video. For locally

Table 13. Recurring failure clusters observed in agentic VQA trajectories on AgentVidBench, with the complete 21-tag closed vocabulary used by the trajectory judge. Each tag is assigned to exactly one of the five trajectory-evaluation axes (TU/ECov/EG/ECom/RF), introduced in Section 3.4.

Cluster (axis)	Failure tags	What goes wrong
Goal interpretation (TU)	`misunderstood_condition`	Agent solves a different question than asked.

Evidence localization (ECov)	`wrong_target_entity`
	`missed_required_segment`
	`wrong_temporal_segment`
	`premature_answer`
	`failure_to_expand_search`
	`failure_to_adjust_granularity`
		Required evidence intervals are not visited; the agent either misses the segment outright or stops the search too early.

Perception (EG)	`visual_misrecognition`
	`ocr_error`
	`counting_error`
	`duplicate_counting`
	`audio_transcript_error`
	`spatial_relation_error`
		The agent reports visual, textual, audio, counting, or spatial evidence that does not match the video.

Search discipline (ECom)	`insufficient_full_video_scan` `premature_fixation`	The agent commits before examining the full timeline for sweep-required questions.

Inferential faithfulness (RF)	`evidence_answer_conflict`
	`unsupported_observation`
	`wrong_arithmetic`
	`wrong_option_mapping`
	`overconfident_uncertainty`
	`excluded_item_counted`
		The final answer does not follow from the evidence collected by the agent.

anchored questions, where a perfect agent could answer from a single well-localized observation, ECom is treated as not applicable. To ensure consistency across frameworks, we pre-determine ECom applicability for each question, include it as dataset metadata, and inject it into the judge input, rather than asking the judge to re-decide applicability independently for each trajectory.

Trajectory Score Aggregation. We aggregate the applicable process-axis scores into a normalized trajectory score:

$$\mathrm{Traj} = \frac{1}{2} \cdot \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} s_a,$$

where $s_a \in \{0, 1, 2\}$ is the score for axis a, and $\mathcal{A} \subseteq \{\mathrm{TU}, \mathrm{ECov}, \mathrm{EG}, \mathrm{ECom}, \mathrm{RF}\}$ is the set of applicable axes. When ECom is not applicable, it is excluded from $\mathcal{A}$. We denote this normalized score as *Traj.* throughout the experimental tables.

Auxiliary Diagnostics. In addition to the process-axis scores used to compute *Traj.*, the judge outputs auxiliary diagnostics for further analysis. First, it assigns a coverage label to each reference milestone: `covered`, `partial`, `incorrect`, or `missing`. These labels are not part of the main trajectory score; they are used to inspect whether each milestone is fully supported, partially addressed, contradicted by an incorrect observation, or absent in the rendered trajectory. This helps distinguish failures caused by missing evidence from those caused by incomplete grounding or incorrect observations. For aggregate reporting, we convert these labels into a soft milestone coverage rate by assigning scores of 1.0, 0.5, 0.0, and 0.0 to `covered`, `partial`, `incorrect`, and `missing`, respectively.

Second, the judge assigns zero or more closed-vocabulary failure tags that summarize common error modes, such as misunderstood conditions, missed evidence, visual misrecognition, OCR errors, counting errors, insufficient evidence-

completeness verification, wrong arithmetic, or evidence–answer conflict. Together, milestone coverage labels and failure tags make it possible to analyze not only how well a system performs, but also why its trajectory succeeds or fails.

Co-occurrence-controlled validation of the tag→axis mapping. Because a single trajectory typically fires several failure tags simultaneously (e.g., a trajectory that misreads on-screen text often also misses the required segment and produces an unsupported observation), naively averaging axis scores over all records that contain a tag conflates the tag's own marginal effect with that of its co-firing neighbours. We therefore validate the cluster→axis mapping in Table 13 by partial-regression analysis on the 8,000-record sweep used in Section 3.4. For each axis $a \in \{\mathrm{TU}, \mathrm{ECov}, \mathrm{EG}, \mathrm{ECom}, \mathrm{RF}\}$ we fit ordinary least squares

$$s_a \;=\; \alpha \;+\; \sum_{T \in \mathcal{T}} \beta_{T,a}\, \mathbf{1}[T \text{ fired}] \;+\; \gamma_a\, \mathbf{1}[\text{sweep}],$$

where $\mathcal{T}$ is the 21-tag vocabulary and the sweep indicator is omitted for $a = \mathrm{ECom}$ (which is undefined on locally-anchored questions and is fit only on sweep-required records). The estimated $\beta_{T,a}$ is the isolated marginal effect of tag T on axis a holding all other tags fixed, removing the co-occurrence inflation present in raw record-level means.

We then compute, for every (cluster, axis) pair, the frequency-weighted mean $\bar{\beta}_{c,a} = \sum_{T \in c} n_T \beta_{T,a} \,/\, \sum_{T \in c} n_T$, where n_T is the number of records on which T fired. For all five clusters, $\bar{\beta}_{c,a}$ attains its minimum (most negative score drop) at the axis to which the cluster is assigned, and conversely each axis attains its column minimum at its assigned cluster, so the cluster→axis correspondence is empirically valid (Fig. 3).

Judge Configuration. The final pipeline uses a milestone-based reference and a single bundled judge call for Task Understanding (TU), Evidence Coverage (ECov), Evidence Grounding (EG), Evidence Completeness (ECom), and Reasoning Faithfulness (RF). We select the judge backbone based on validation experiments measuring whether judge scores separate correct and incorrect trajectories and produce stable process-quality judgments across axes.

C.0.1. Model Settings and Prompt Templates

To support reproducibility, we provide the model configurations and prompt templates used for milestone extraction and trajectory judging. The final pipeline uses GPT-5 for extracting milestones and Claude Opus 4.7 for judging trajectories. All trajectory judge calls use milestone-only references.

Table 14. Model settings used in milestone extraction and trajectory judging.

Component	Model	Settings
Milestone extraction	GPT-5	Reasoning effort: medium; maximum completion tokens: 16,384.
Trajectory judge	Claude Opus 4.7	1M context enabled; maximum output tokens: 4,096.

Milestone Extraction Prompt. The milestone extraction prompt converts a human-written ground-truth reasoning note into a structured, tool-agnostic list of evidence milestones.

Milestone extraction system prompt

System prompt.
You are abstracting a video QA question's human-written solution path into tool-agnostic evidence milestones.

Different agents use different tool interfaces, such as low-level video controls, high-level video-analysis calls, or zero-shot video input. Comparing raw tool sequences would be unfair. Extract instead the underlying evidence the agent must acquire to reach the answer: what to find, not how to find it.

Inputs.
You will receive the question text, answer options, gold answer, gold answer text, and the human-written answer explanation.

Output.
Return a JSON object with a `milestones` array. Each milestone contains:

- `id`: `M1`, `M2`, ... starting from 1.
- `type`: one of `context_identification`, `temporal_localization`, `ocr_text_reading`, `audio_visual_evidence`, `visual_recognition`, `counting`, `comparison`, `exclusion_check`,

`arithmetic`, or `verification`.

- `description`: a one-line English description of the evidence to acquire.

Core principles.
Milestones must be tool-agnostic. Do not mention tool names such as OCR tools or video-analysis calls. Do not include meta-references such as "in the ground-truth solution." Do not bake specific timestamps, narrow time ranges, timestamp lists, or solution-path-specific arithmetic decompositions into the description. Use descriptions such as "Locate the moment when the target object is visible" rather than "Inspect the video at 02:13."

Do not include a final option-mapping milestone. Stop at the value derivation or evidence requirement; mapping to an option letter is not treated as evidence.

For counting questions, do not create one milestone per occurrence. Group the target as a single counting milestone and use a separate exclusion-check milestone when the question requires excluding distractors or duplicates.

For OCR or identification questions, include the value to be read when that value is necessary for the answer. For arithmetic or comparison questions, include the required operation at the evidence level without prescribing a particular human decomposition unless the decomposition itself is required by the question.

Return JSON only, with no markdown fence and no preamble.

Trajectory Judge Prompt. The trajectory judge receives a milestone-only reference in the default setting. The original human answer explanation is not included in the judge input, so that the judge evaluates whether the trajectory recovers the underlying evidence rather than whether it follows the same human solution path.

Trajectory judge system prompt

System prompt.
You are a strict evaluator of video-investigation trajectories on AgentVidBench.

This call evaluates the five process-quality axes, P1–P5, together with milestone coverage labels and failure tags. Score each applicable axis on a 0–2 scale, where 0 is worst and 2 is best. P4 may be `null` when the question does not require a full-scope evidence sweep.

Follow the tool-agnostic principle: do not penalize differences in tool names, tool interfaces, or search paths. The agent may have arrived at the same evidence via a different path.

Use `GT_MILESTONES` as the sole reference for evidence. Do not use anything else as a reference.

Process axes.

- **P1 Task Understanding**: did the agent correctly understand WHAT the question asks — target entity, constraints, exclusions, temporal scope, and option-letter range?
  - **2**: agent's framing matches the question exactly; all constraints (exclusions, qualifiers, "first/last", option range A–Z) acknowledged either explicitly or via consistent behavior.
  - **1**: core task understood but a constraint or qualifier missed (e.g., misses an exclusion, treats "first" as "any").
  - **0**: fundamental misunderstanding — wrong target entity, wrong question type, or ignores a critical constraint that determines the answer.
- **P2 Ground-truth Evidence Coverage**: how many of the `GT_MILESTONES` did the agent's trajectory ACQUIRE? Use the milestone_coverage classifications as the basis. All milestones treated uniformly (no required/convenient distinction).
  - **2**: all milestones covered (or covered via tool-agnostic alternate paths).
  - **1**: some milestones missing or partially covered, but enough acquired that the answer was reachable.
  - **0**: most milestones missing or incorrect; the agent's answer is essentially unsupported by acquired evidence.
- **P3 Evidence Grounding**: are the values the agent CLAIMS (OCR readings, counts, observed objects, audio quotes, arithmetic intermediate values) consistent with what the milestones say?
  - **2**: every claimed value matches the milestone evidence; no fabrication, no misreading.
  - **1**: most values correct; at least one mismatch (wrong OCR digit, off-by-N count, mis-identified object) but the mismatch is non-fatal.
  - **0**: multiple claimed values wrong, OR the value that determines the final answer is wrong/fabricated.
- **P4 Exhaustive Evidence Sweep** (may be `null`): did the agent verify that the relevant temporal scope was adequately inspected before committing? The user prompt declares `SWEEP_REQUIRED` per qid (`yes`/`no`/`unknown`). If `no` (anchored question, single-segment determination), set this axis to `null` and skip. If `yes`:
  - **2**: agent explicitly extends inspection past the first match AND confirms no additional / contradicting / qualifying evidence in remaining temporal scope, citing concrete inspected scope.

  - **1**: extends past first match but partial — scans later portion without explicit FULL-scope confirmation, OR mentions later events as a possibility without inspecting them.
  - **0**: commits at first match without any check. Verbal confidence ("clearly", "I'm sure") does NOT count as a sweep.
- **P5 Reasoning Faithfulness**: does each reasoning step follow from the evidence the agent claimed? Is the arithmetic correct? Are there unsupported leaps from evidence to the final answer?
  - **2**: each step traceable to acquired evidence; arithmetic correct; no unsupported leaps; final answer follows from the chain.
  - **1**: minor leap or arithmetic slip but core flow defensible; final answer still consistent with the trajectory's evidence.
  - **0**: reasoning contradicts own evidence (claims X then concludes ¬X), OR commits a fatal arithmetic / logic error, OR final answer is disconnected from the evidence acquired.

Milestone coverage.
For each reference milestone (fixed IDs M1..Mn given in the user prompt), assign one of four labels: `covered`, `partial`, `incorrect`, or `missing`. The output must list the same milestones in the same order. These labels feed an auxiliary mc_rate aggregate (`covered`=1, `partial`=0.5, `incorrect`=0, `missing`=0) reported alongside the P-axis scores.

Failure tags.
Select zero to eight tags from the closed vocabulary:

```
misunderstood_condition      wrong_target_entity
visual_misrecognition        ocr_error
audio_transcript_error       unsupported_observation
counting_error               duplicate_counting
excluded_item_counted        wrong_arithmetic
wrong_option_mapping         wrong_temporal_segment
missed_required_segment      insufficient_full_video_scan
spatial_relation_error       premature_fixation
failure_to_expand_search     failure_to_adjust_granularity
evidence_answer_conflict     premature_answer
overconfident_uncertainty
```

The predicted answer letter is computed externally; this judge does NOT score answer correctness.

Output. Return strict JSON only.

Judge User Template. For each question–prediction pair, we instantiate the following user prompt.

Trajectory judge user template

```
QUESTION_ID: {qid}
FRAMEWORK: {framework}
SWEEP_REQUIRED: {sweep_required}

QUESTION:
{question_text}

OPTIONS:
{options}

GT_MILESTONES (curated, fixed IDs M1..Mn --- classify coverage status for EACH
    , in order):
{milestones_block}

PREDICTED_TRAJECTORY:
{prediction}

Evaluate this prediction against the rubric. Output the JSON object only.
Reminder: milestone_coverage MUST contain exactly {n_milestones}
entries with IDs {milestone_ids} in this order.
```

Judge Output Schema. The judge returns a strict JSON object with process scores, short rationales, auxiliary milestone coverage labels, failure tags, and a concise summary.

Trajectory judge output schema

```
{
  "process_scores": {
    "task_understanding": 0,
    "gt_evidence_coverage": 0,
    "evidence_grounding": 0,
    "evidence_completeness": null,
    "reasoning_faithfulness": 0
  },
  "axis_rationales": {
    "P1": "...",
    "P2": "...",
    "P3": "...",
    "P4": "...",
    "P5": "..."
  },
  "milestone_coverage": [
    {
      "milestone_id": "M1",
      "status": "covered",
      "evidence_in_prediction": "..."
    }
  ],
  "failure_tags": ["..."],
  "summary": "..."
}
```

D. Appendix – Auxiliary Trajectory Diagnostics

This appendix provides additional trajectory-level diagnostics beyond the aggregate trajectory score reported in the main text. These diagnostics are derived from the same milestone-based trajectory judge described in Appendix C, but they are used only for analysis and are not included in the computation of *Traj*.

D.1. Milestone Coverage Rate

In addition to the five process-quality axes, the trajectory judge assigns a coverage label to each reference milestone: `covered`, `partial`, `incorrect`, or `missing`. We report a soft milestone coverage rate that assigns full credit to `covered` milestones, half credit to `partial` milestones, and zero credit to `incorrect` or `missing` milestones:

$$\text{MC-Rate} = \frac{n_{\text{covered}} + 0.5\, n_{\text{partial}}}{N_{\text{milestones}}}.$$

Here, $N_{\text{milestones}}$ denotes the total number of reference milestones. Unlike the main trajectory score, this metric does not average over the five process-quality axes. It provides a direct diagnostic view of how much of the reference evidence each system recovers at the milestone level.

D.2. Detailed Process-quality Profiles

Figures 6, 7, and 8 provide the full trajectory process-quality profiles for proprietary MLLMs, Qwen-family open-source MLLMs, and Gemma-family open-source MLLMs, respectively. These figures complement the aggregate trajectory scores reported in the main text by showing how each method differs across the five process-quality axes: Task Understanding (TU), Evidence Coverage (ECov), Evidence Grounding (EG), Evidence Completeness (ECom), and Reasoning Faithfulness (RF).

Table 15. Soft milestone coverage rate on the AgentVidBench dataset, reported per backbone × method. The metric assigns full credit to `covered` milestones, half credit to `partial` milestones, and zero credit to `incorrect` or `missing` milestones. Blue arrows indicate absolute changes relative to the corresponding Single-Turn setting.

Models	**Single-Turn**	**STAR**	**DVD**	**AVP**	**Ours**
Proprietary MLLMs					
GPT-5	0.58	0.29	0.58	0.66	**0.75** ↑0.17
Claude Opus 4.7	0.42	0.31	0.63	0.64	**0.78** ↑0.36
Gemini 2.5 Flash	0.68	0.25	0.44	0.66	**0.71** ↑0.03
Gemini 2.5 Pro	0.73	0.28	0.45	0.74	**0.79** ↑0.06
Open-source MLLMs					
Qwen family					
Qwen3-VL-4B	0.34	0.16	0.07	**0.70**	0.60 ↑0.26
+*Think*	0.28	0.04	0.24	**0.70**	0.63 ↑0.35
Qwen3-VL-8B	0.42	0.16	0.10	0.64	**0.67** ↑0.25
+*Think*	0.27	0.05	0.24	0.67	**0.67** ↑0.40
Qwen3.5-9B	0.36	0.13	0.50	0.67	**0.70** ↑0.35
+*Think*	0.56	0.12	0.51	0.70	**0.73** ↑0.17
Qwen3.5-27B	0.56	0.21	0.59	0.70	**0.80** ↑0.23
+*Think*	0.62	0.24	0.54	0.71	**0.78** ↑0.16
Gemma family					
Gemma-4-E2B	0.21	0.11	0.32	**0.71**	0.68 ↑0.47
+*Think*	0.21	0.10	0.22	**0.71**	0.68 ↑0.47
Gemma-4-E4B	0.34	0.15	0.42	0.70	**0.74** ↑0.40
+*Think*	0.38	0.14	0.40	**0.68**	0.66 ↑0.28
Gemma-4-26B-A4B	0.44	0.21	0.53	0.73	**0.76** ↑0.32
+*Think*	0.44	0.16	0.57	0.72	**0.77** ↑0.33
Gemma-4-31B	0.61	0.20	0.57	0.71	**0.76** ↑0.15
+*Think*	0.62	0.23	0.60	0.70	**0.75** ↑0.13

Figure 6. Trajectory score profiles of proprietary MLLMs under Single-Turn and agentic frameworks on AgentVidBench. Each radar plot shows the average scores across five trajectory evaluation dimensions: Task Understanding (TU), Evidence Coverage (ECov), Evidence Grounding (EG), Evidence Completeness (ECom), and Reasoning Faithfulness (RF). Columns correspond to Single-Turn, STAR, DVD, AVP, and Ours. The dashed outline indicates the human reference score, where larger filled regions represent stronger trajectory quality.

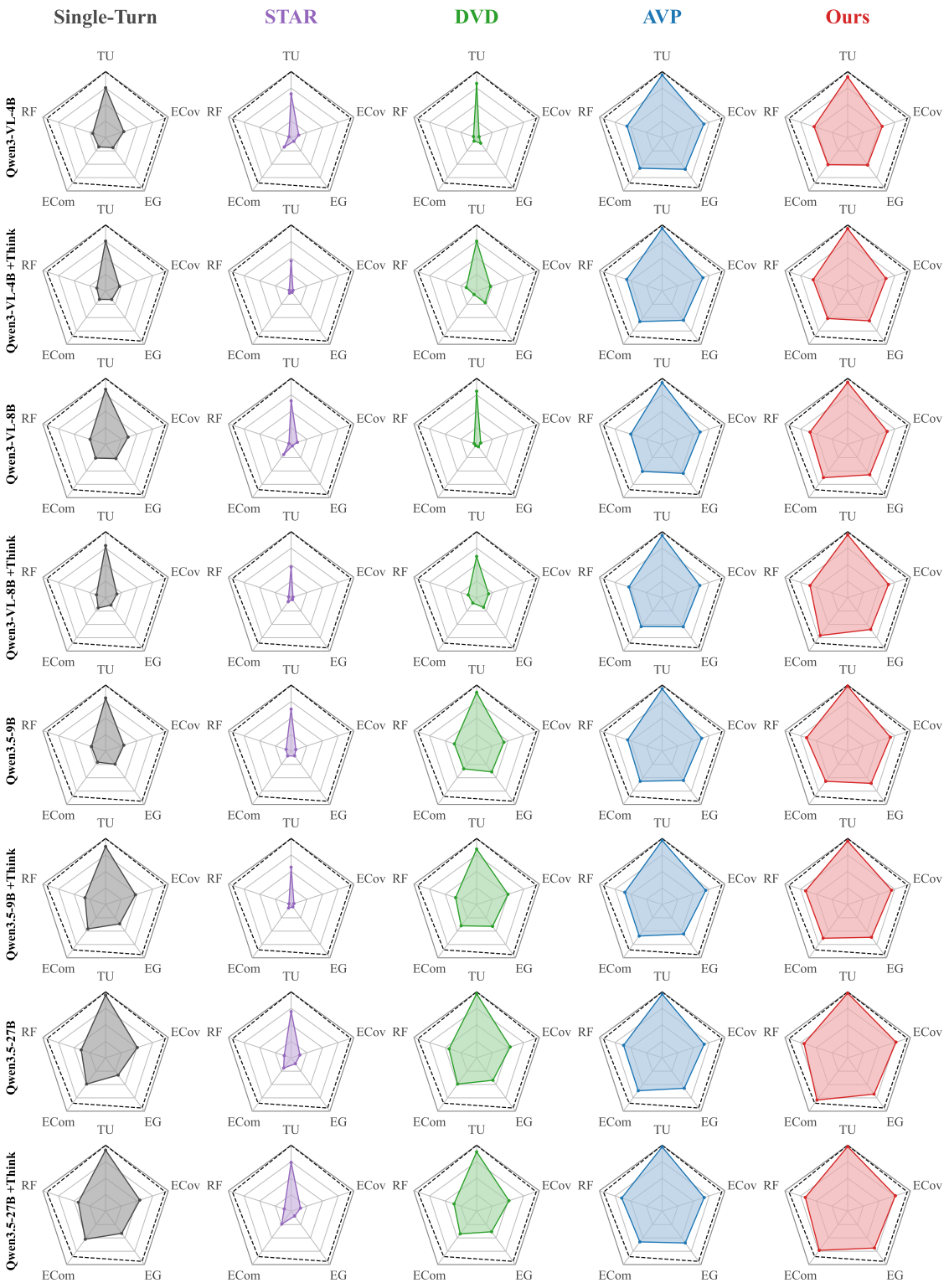


Figure 7. Trajectory score profiles of Qwen-family open-source MLLMs under Single-Turn and agentic frameworks on AgentVidBench. Each row corresponds to a Qwen backbone or its thinking variant, and each column corresponds to Single-Turn, STAR, DVD, AVP, or Ours. Each radar plot shows the average scores across five trajectory evaluation dimensions: Task Understanding (TU), Evidence Coverage (ECov), Evidence Grounding (EG), Evidence Completeness (ECom), and Reasoning Faithfulness (RF). The dashed outline indicates the human reference score, where larger filled regions represent stronger trajectory quality.

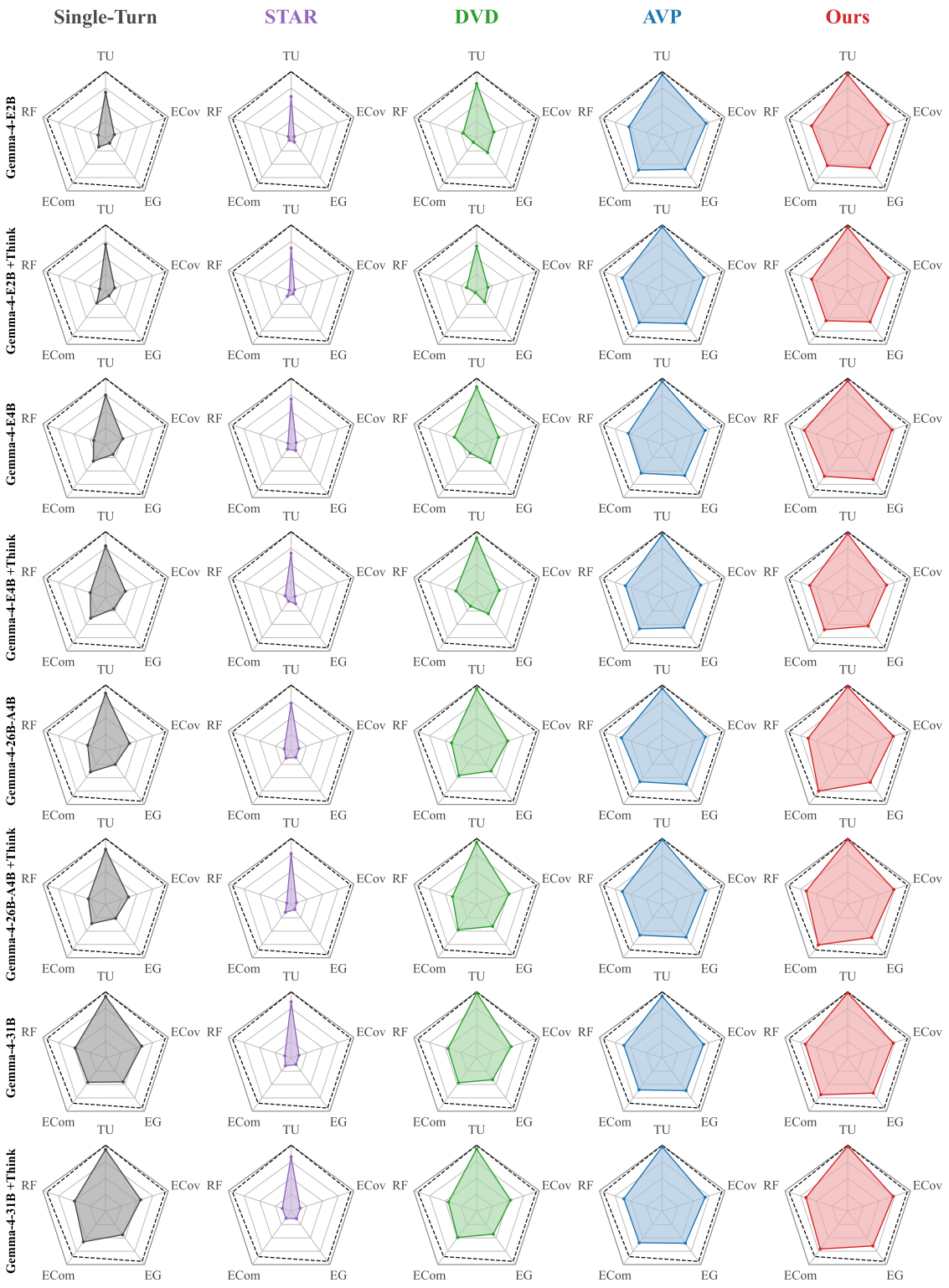


Figure 8. Trajectory score profiles of Gemma-family open-source MLLMs under Single-Turn and agentic frameworks on AgentVidBench. Each row corresponds to a Gemma backbone or its thinking variant, and each column corresponds to Single-Turn, STAR, DVD, AVP, or Ours. Each radar plot shows the average scores across five trajectory evaluation dimensions: Task Understanding (TU), Evidence Coverage (ECov), Evidence Grounding (EG), Evidence Completeness (ECom), and Reasoning Faithfulness (RF). The dashed outline indicates the human reference score, where larger filled regions represent stronger trajectory quality.

E. Appendix – Additional Performance Analysis

E.1. Ablation Study on Adaptive Video Inspection

To better understand which types of video control matter for agentic video understanding within AgentVidBench, we ablate the main tool-control axes available to the agent when deciding how to inspect the video. These axes correspond to complementary ways of inspecting video evidence: *fps* controls temporal granularity (FPS), *resolution* controls spatial fidelity (Res.), and *time interval* controls which portion of the video is observed (Int.). All ablations use Gemini 2.5 Pro on the same evaluation set, and we report final-answer accuracy and trajectory score.

The results in Table 16 show that removing adaptive control generally reduces accuracy, while the impact of each control axis varies across agentic frameworks. Specifically, while the resolution and time interval tools play a dominant role in the AVP framework, all three tools demonstrate comparable importance in our framework. These results indicate that AgentVidBench is effective for evaluating the gains obtained from adaptive inspection compared to passive or uniformly allocated observation.

Table 16. Ablation study on adaptive video-inspection controls using Gemini 2.5 Pro.

	Adaptive Controls			**AVP**		**Ours**	
Settings	FPS	Res.	Int.	Acc.	Traj.	Acc.	Traj.
W/o FPS	1	✓	✓	0.50	0.69	0.50	0.73
W/o Res.	✓	low	✓	0.49	0.68	0.56	0.73
W/o Res.	✓	med	✓	0.43	0.66	0.55	0.77
W/o Int.	✓	✓	full	0.46	0.68	0.50	0.72
Full	✓	✓	✓	0.56	0.71	0.58	0.77

E.2. Efficiency Analysis

We report performance per token in Table 17, defining token efficiency as accuracy divided by the total number of tokens measured in millions. Overall, single-turn baseline shows the lowest efficiency, achieving a score of 3.36%. This is because it processes the entire video regardless of which segments are relevant for solving the problem. In contrast, agentic methods can use tools to selectively inspect the relevant parts of the video, leading to larger performance gains per token. AVP achieves the highest token efficiency of 5.28%, whereas our method achieves the best accuracy of 58% while remaining more efficient than the single-turn baseline. This result indicates that tool-based reasoning improves token utilization, demonstrating its effectiveness from an efficiency perspective as well.

Table 17. Accuracy vs. token efficiency. Eff. denotes token efficiency, computed as accuracy (Acc.) divided by million tokens (MTok.).

Methods	**Acc.**	**MTok.**	**Eff. (%)**
Single-Turn	0.51	15.16	3.36
AVP	0.56	10.60	5.28
Ours	0.58	15.44	3.76

F. Appendix – Limitations

Although AgentVidBench enables rigorous evaluation through carefully curated questions paired with fine-grained trajectories, its current coverage remains limited. This limitation primarily stems from two factors: our effort to collect openly licensed videos and the labor-intensive nature of designing questions that require multi-step reasoning and constructing validated trajectories. To address this limitation, future work will explore automated data collection methods for scaling AgentVidBench without compromising annotation quality.