VIDDEDUCE: ENHANCING LONG-FORM VIDEO UNDERSTANDING VIA DEDUCTIVE VERIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-form video understanding poses challenges in clue localization and logical reasoning. While agents enhance reasoning via adaptive region selection and evidence gathering, current paradigms either introduce query-agnostic redundancy via global text preprocessing, or neglect global context through localized iterative retrieval. These limitations lead agents to form misleading intermediate hypotheses, trapping them in unviable reasoning trajectories. To enable evidence-guided search refinement, we propose VidDeduce, a training-free multi-agent framework comprising a cognitive planner, visual observer, and a verifier based on Natural Language Inference (NLI). The planner integrates clue-relevant captions with prior verification feedback to update accumulated evidence and unresolved gaps, and formulates subsequent search strategies. Translating these strategies into actionable perception, the observer distills relevant captions and adaptively performs visual inspection to acquire observations. The verifier assesses new observations against the accumulated evidence and unresolved gaps via NLI, providing explicit feedback to guide continued investigation of useful clues or revise unproductive search directions, ensuring every exploratory step possesses a traceable logical rationale. Extensive experiments across four video understanding benchmarks demonstrate the effectiveness of VidDeduce, which outperforms strong agentic baselines including DVD and LensWalk on Video-MME, LongVideoBench, and EgoSchema.

1 INTRODUCTION

Long-form Video Understanding requires locating relevant events and integrating evidence across extensive timelines (Wang et al., 2025a; Wu et al., 2024; Zhou et al., 2025). Despite advances in visual perception and reasoning, vision-language models remain constrained by the tension between the immense volume of video content and limited context windows (Chen et al., 2025; Shen et al., 2024; Li et al., 2026b). Common approaches alleviate this constraint through compact visual representations (Li et al., 2024; Liu et al., 2025; Wang et al., 2025b; Song et al., 2024; Shu et al., 2025) or selective retrieval of question-relevant content (Wang et al., 2023; 2024; Luo et al., 2026). However, these strategies risk omitting brief but decisive details, and when observations are determined before reasoning, they offer limited flexibility to accommodate both global evidence aggregation and detailed local inspection. This limitation is particularly consequential for complex queries, where intermediate findings determine what evidence to seek next and at what level of detail.

To accommodate evolving information needs, video agents leverage language models to coordinate iterative evidence acquisition (Yang et al., 2024; Min et al., 2024; Fan et al., 2024; Zhang et al., 2026). Existing approaches typically organize video content into structured memories to support retrieval across temporal scales (Wang et al., 2025c; Zuo et al., 2026), or enable adaptive observation that allows the agent to select tools and dynamically adjust its visual focus based on accumulated context (Yang et al., 2025b; Li et al., 2026a). These mechanisms make information gathering more flexible, but the relevance of observed content does not necessarily establish its validity as a reasoning premise. Precomputed descriptions like captions often omit question-specific details and introduce irrelevant or even misleading cues, whereas localized observations can lack the broader context required to distinguish a target event from visually similar distractions. A core issue is that evidence insufficiency can reflect different underlying states: an observation might *partially support* the current hypothesis, directly *contradict* it, or yield *no informative value*. Although none of these

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

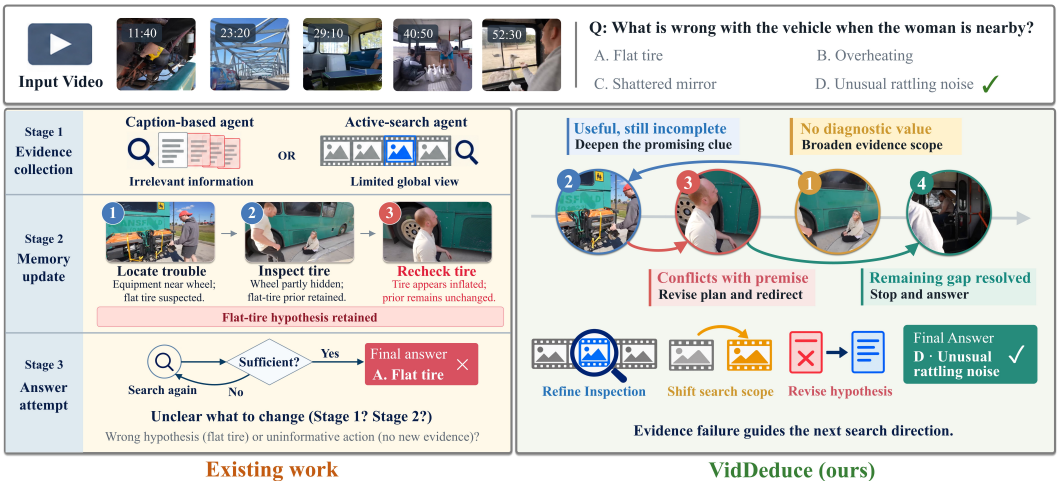


Figure 1: Comparison between conventional video agents and VidDeduce. While conventional agents rely on binary sufficiency checks and retain unsupported premises, VidDeduce uses evidence verification to deepen, broaden, redirect, or terminate the search trajectory.

outcomes resolves the question, they demand entirely different responses to advance toward the final answer. Partial support necessitates focused refinement to bridge the remaining evidence gap. Contradictory observations require revising the current hypothesis and redirecting the search, whereas uninformative observations call for broadening the evidence scope or reconsidering where to search. As illustrated in Fig. 1, existing video agents often collapse these distinct outcomes into a binary sufficiency check, allowing irrelevant observations and unsupported premises to persist across iterations. In contrast, VidDeduce connects evidence verification to trajectory revision, utilizing the relationship between new observations and ongoing evidence needs to steer subsequent exploration.

To this end, we propose VidDeduce, a training-free multi-agent framework that regulates search trajectories through Natural Language Inference (NLI) feedback. VidDeduce evaluates new observations against accumulated evidence and unresolved gaps, using the verification feedback to update its reasoning state and guide subsequent search. The resulting NLI signals and reasoning provide an explicit basis for refining useful clues, revising contradicted assumptions, or reconsidering the search direction when observations are contradictory or uninformative. This feedback regulates evidence acquisition across clue-driven context refresh, gap-focused caption retrieval, and targeted VLM probing, while also adapting temporal search scopes and observation scales. To dynamically adapt to these evolving evidence needs, VidDeduce combines coarse textual context with dual-scale visual exploration, maintaining access to broader video information while enabling focused inspection of local details. By translating cognitive verification into actionable trajectory adjustments, VidDeduce supports deeper exploration of promising paths, decisively pivots away from contradictory or uninformative clues, and accurately terminates search once sufficient evidence is deduced. Our main contributions are summarized as follows:

- We identify a recurring failure mode in iterative video reasoning: adopting intermediate clues as reasoning premises without adequate verification propagates unsupported assumptions into subsequent searches, leading to fixation on irrelevant details or unproductive search trajectories.
- We propose VidDeduce, a training-free multi-agent framework that integrates NLI-guided trajectory regulation with dual-scale visual exploration. Verifier feedback jointly adapts search queries, temporal scopes, and observation scales, supporting focused refinement of promising clues and redirection from unproductive trajectories.
- Experiments on video understanding benchmarks demonstrate the effectiveness of VidDeduce. Among the compared methods, VidDeduce achieves the highest accuracy on Video-MME (Long), LongVideoBench (Long), and EgoSchema. Ablations and trajectory analyses further show that NLI-guided evidence acquisition improves accuracy over confidence-based feedback and supports targeted search adjustments in response to conflicting or uninformative observations.

2 Related Work

Long-form Video Understanding. Long-form video understanding balances broad temporal coverage with fine-grained perception under limited context budgets (Ren et al., 2024). Beyond token compression and selective sampling, existing methods organize video content into representations for retrieval and reasoning. LLoVi (Zhang et al., 2024) aggregates clip captions, while Lifelong-Memory (Wang et al., 2023) maintains textual video memories. VideoTree (Wang et al., 2025c) and VideoLucy (Zuo et al., 2026) use hierarchical representations for coarse-to-fine exploration, whereas MR. Video (Pang & Wang, 2026) adopts MapReduce-based information aggregation. Complementary work focuses on coordinating reasoning and evidence acquisition. DoraemonGPT (Yang et al., 2024) combines symbolic memory with tree-search planning, VideoAgent (Wang et al., 2024) retrieves frames to address missing information, and MoReVQA (Min et al., 2024) employs modular reasoning stages. VCA (Yang et al., 2025b) guides segment exploration with intrinsic rewards, while Deep Video Discovery (Zhang et al., 2026) adaptively selects browsing, retrieval, and inspection tools. Vgent (Shen et al., 2026) integrates graph-based retrieval with structured reasoning, and LensWalk (Li et al., 2026a) dynamically adjusts observation scope and sampling density. Building on these advances, VidDeduce treats the search process as a revisable trajectory by coordinating clue-driven context refresh, gap-focused semantic retrieval, and adaptive visual inspection, allowing search targets, temporal coverage, and observation granularity to evolve with the acquired evidence.

Agent Reflection and Test-time Scaling. When tackling complex long-sequence reasoning, endowing agents with self-reflection and test-time scaling capabilities is a key approach to enhancing system robustness. ReAct (Yao et al., 2022) interleaves reasoning with actions, Tree of Thoughts (Yao et al., 2023) explores and evaluates alternative reasoning paths, and Reflexion (Shinn et al., 2023) uses verbal feedback to improve later attempts. Prior work connects question answering with NLI by converting question-answer pairs into declarative hypotheses and assessing their entailment from supporting passages (Demszky et al., 2018). Stepwise deductive verification further improves reasoning reliability by checking whether intermediate conclusions are justified by their supporting premises (Ling et al., 2023). These studies highlight the importance of examining the basis of intermediate conclusions rather than evaluating final answers alone. Building upon these insights, VidDeduce integrates an explicit NLI-based arbitration mechanism to assess new observations against accumulated evidence and unresolved needs. This deductive verification provides signals and reasoning that guide subsequent adjustments to search queries, temporal scopes, and observation scales, enabling deeper exploration of promising clues and redirection from unsupported trajectories.

3 Methodology

Overview. Given a long video V and a user query Q, our goal is to seek an answer A by gathering and verifying question-relevant evidence. To overcome the limitations of existing video agents, which are prone to being trapped in unproductive trajectories during long-sequence exploration and treating invalid local contexts as priors, we propose VidDeduce, a multi-agent collaborative framework driven by deductive verification. As illustrated in Fig. 2(c), the framework decouples cognitive planning from dual-scale visual execution. Through the plan-observe-verify loop, verification feedback coordinates query updates for context refresh, caption retrieval, and VLM probing, while also regulating temporal scopes and observation scales. These adjustments allow evidence acquisition to be refined or redirected as reasoning evolves. VidDeduce consists of three main components: a cognitive planner that tracks evidence and formulates search strategies, a visual observer that integrates caption-derived context with dual-scale visual inspection, and an NLI-based verifier responsible for logical alignment and trajectory regulation.

3.1 Cognitive Tracking and Search Planning

Video Database Construction. To maintain a holistic view for temporal localization while avoiding the computational overhead of captioning query-irrelevant content, we first construct a coarse-grained global video database indexed by dense semantic embeddings prior to inference. Specifically, we partition the video into N consecutive clips $\{v_i\}_{i=1}^{N}$. For each clip v_i, a vision-language model generates a caption c_i, which an embedding model then encodes into a dense semantic vector

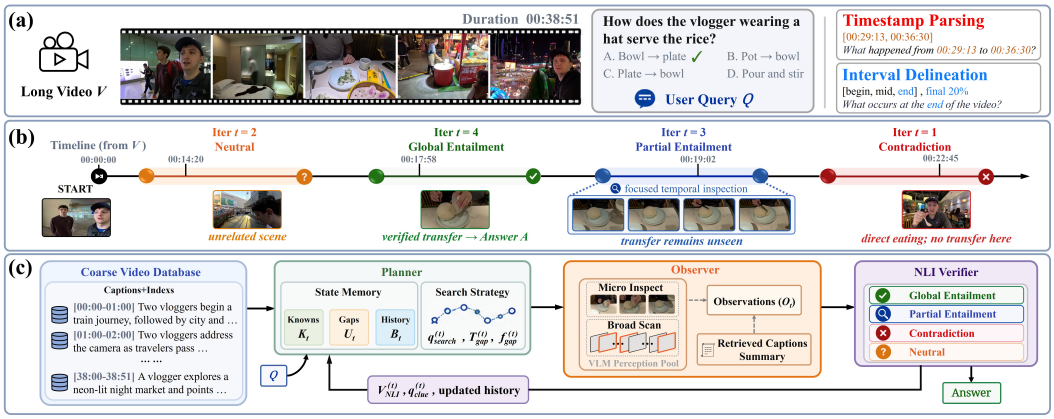


Figure 2: Overview of our proposed VidDeduce. (a) Video and question inputs. (b) A verification-guided search trajectory. Each iteration focuses on selected intervals. (c) Framework architecture.

e_i. We integrate the clips, their corresponding captions, and dense semantic vectors into a global video database, formulated as $\mathcal{D}_{\text{global}} = \{(v_i, c_i, e_i)\}_{i=1}^{N}$.

Temporal Search Guidance. We derive temporal guidance from both the user question and the evolving evidence requirements. As illustrated in Fig. 2(a), timestamp parsing extracts explicit numerical references from Q, while interval delineation maps references to the beginning, middle, or end of the video to $[0, 0.2L]$, $[0.4L, 0.6L]$, or $[0.8L, L]$, respectively, where L denotes the video duration. Both are represented on the global video timeline as $\mathcal{T}_Q = \{[t^j_{\text{start}}, t^j_{\text{end}}]\}_{j=1}^{K}$, where K denotes the number of reference intervals, and t^j_{start} and t^j_{end} denote the start and end timestamps of the j-th interval, respectively. These intervals provide a common temporal reference for caption retrieval and subsequent visual inspection. To accommodate different levels of temporal specificity, we distinguish interval-restricted from interval-prioritized retrieval through a binary flag f_Q. The former retains only clips overlapping $\mathcal{T}_Q$, whereas the latter prioritizes overlapping clips and supplements them with the highest-ranked matches elsewhere when fewer than k candidates are available. Without a temporal reference, retrieval considers the full video. Before initiating visual exploration, the system attempts an initial answer using the chronological captions, filtered to the specified intervals when interval-restricted selection applies. Questions judged answerable with high confidence are resolved directly; otherwise, the system proceeds to iterative evidence acquisition.

During exploration, question-derived temporal references are complemented by evidence-conditioned intervals. At iteration t, the planner infers a temporal scope $\mathcal{T}^{(t)}_{\text{gap}}$ from the current evidence gaps and verifier feedback, together with a selection flag $f^{(t)}_{\text{gap}}$ specifying interval-restricted or interval-prioritized retrieval. This scope takes precedence when specified; otherwise, retrieval falls back to $\mathcal{T}_Q$. Unlike the question-derived reference, the inferred scope can be refined, expanded, or relocated as evidence needs evolve, enabling the non-chronological exploration illustrated in Fig. 2(b). Combining global temporal references with evolving search scopes, and interval restriction with prioritization, balances temporal specificity with exploratory flexibility: retrieval concentrates on relevant regions while retaining opportunities to recover evidence outside the current focus.

Search Strategy Generation. The planner combines retrieved caption context with prior search and verification records to determine what evidence to seek next and where to search. The caption context $\mathcal{C}^{(t)}_{\text{global}}$ is initially retrieved using the original question Q and refreshed after each iteration using the verifier-derived clue $q^{(t)}_{\text{clue}}$:

$$\mathcal{C}^{(0)}_{\text{global}} = \text{Retrieve}(\mathcal{D}_{\text{global}}, Q, \mathcal{T}_Q, f_Q)\,, \quad \mathcal{C}^{(t+1)}_{\text{global}} = \text{Retrieve}\Big(\mathcal{D}_{\text{global}}, q^{(t)}_{\text{clue}}, \mathcal{T}_Q, f_Q\Big)\,. \tag{1}$$

The planner maintains a structured history buffer $\mathcal{B}_t$ of prior search and verification records to trace decision rationales and identify unsuccessful search patterns. At iteration t, it integrates this history with the current caption context $\mathcal{C}^{(t)}_{\text{global}}$ to update two complementary cognitive states: accumulated

evidence $\mathcal{K}_t$ and unresolved gaps $\mathcal{U}_t$. During this update, the planner retains useful clues, revises unsupported assumptions, and reconsiders uninformative search directions. These states jointly guide the generation of a targeted visual search query $q_{\text{search}}^{(t)}$, temporal scope $\mathcal{T}_{\text{gap}}^{(t)}$, and selection flag $f_{\text{gap}}^{(t)}$ defined above. The joint state update and strategy generation are formulated as:

$$\mathcal{K}_t, \mathcal{U}_t, q_{\text{search}}^{(t)}, \mathcal{T}_{\text{gap}}^{(t)}, f_{\text{gap}}^{(t)} = \text{Planner}\Big(Q, \mathcal{B}_t, \mathcal{C}_{\text{global}}^{(t)}\Big). \tag{2}$$

By regenerating these directives based on the updated cognitive states, the system maintains dynamic control over its exploration trajectory. Rather than rigidly adhering to an initial plan, the planner can leverage newly confirmed clues to seamlessly refine its temporal focus or shift the search direction to address emerging evidence gaps.

3.2 Temporal Localization and Multi-scale Visual Execution

Caption Summarization. Given the planner's search query $q_{\text{search}}^{(t)}$ and temporal guidance, the observer retrieves relevant captions $\mathcal{C}_{\text{match}}^{(t)}$ from the global video database:

$$\mathcal{C}_{\text{match}}^{(t)} = \text{Retrieve}\Big(\mathcal{D}_{\text{global}}, q_{\text{search}}^{(t)}, \mathcal{T}_{\text{gap}}^{(t)}, f_{\text{gap}}^{(t)}\Big), \tag{3}$$

where retrieval uses the question-derived reference $\mathcal{T}_Q$ when $\mathcal{T}_{\text{gap}}^{(t)}$ is unspecified.

The shared captions provide reusable video-level descriptions but may include content unrelated to the current question. The observer therefore distills the retrieved captions into a summary s_t conditioned on the search query and unresolved gaps $\mathcal{U}_t$. This summary extracts question-specific clues from reusable descriptions without recaptioning the entire video for each question. It is subsequently combined with visual findings to form the observation.

Adaptive Visual Probing. Building on the retrieved context, the observer determines which candidate clips require visual inspection and formulates a targeted visual query $q_{\text{visual}}^{(t)}$. The planner specifies the search objective and temporal guidance, while the observer chooses inspection intervals from the timestamps associated with $\mathcal{C}_{\text{match}}^{(t)}$, rather than independently defining temporal boundaries. This division combines embedding-based semantic matching with language-model interpretation of caption-derived context to concentrate VLM computation on relevant regions. The observer schedules one tool per iteration from the visual toolset $\tau \in \{\tau_{\text{coarse}}, \tau_{\text{fine}}\}$ according to the coverage and detail required by the unresolved evidence gaps. For broad exploration, τ_{coarse} jointly examines multiple segments distributed across the video within an aggregate temporal budget, supporting comparison across candidate regions. For focused inspection, τ_{fine} examines a single continuous interval for detailed analysis of coherent actions, object attributes, and on-screen text. These tools allow the observation scale to follow evolving evidence requirements rather than a fixed coarse-to-fine sequence. The selected tool produces temporally referenced visual findings, which are combined with the caption summary s_t to form the observation O_t:

$$O_t = \text{Concat}\Big(s_t, \text{VLM}_\tau\Big(V_{\text{selected}}^{(t)}, q_{\text{visual}}^{(t)}\Big)\Big), \tag{4}$$

where $V_{\text{selected}}^{(t)}$ denotes the video segments selected for inspection at iteration t, VLM_τ performs visual analysis using tool τ. The combined observation is subsequently assessed by the verifier against the unresolved evidence requirements.

3.3 NLI-Based Verification and Trajectory Regulation

Evidence Verification. Existing reflection mechanisms in video agents typically rely on binary sufficiency judgments or relevance scores. Such signals may identify insufficient evidence but offer limited guidance on its underlying cause or the appropriate search adjustment, leaving agents susceptible to uninformative sampling and repetitive visual actions. To distinguish why an observation remains insufficient, the NLI-based verifier conducts semantic and logical alignment of the newly acquired observation O_t with the accumulated evidence $\mathcal{K}_t$ and unresolved gaps $\mathcal{U}_t$. It returns a verification signal $V_{\text{NLI}}^{(t)}$, supporting reasoning r_t, and a search clue $q_{\text{clue}}^{(t)}$, which together guide trajectory regulation and ensure that each subsequent action has practical significance for the ongoing search:

$$V_{\text{NLI}}^{(t)}, r_t, q_{\text{clue}}^{(t)} = \text{Verifier}(Q, \mathcal{K}_t, \mathcal{U}_t, O_t). \tag{5}$$

Table 1: Definitions of the Natural Language Inference (NLI) signals.

NLI Signal	Definition
Global Entailment	The observations resolve the remaining evidence gaps and, together with accumulated evidence, provide sufficient support for answering the original query.
Partial Entailment	The observations provide relevant evidence or clues that partially resolve the current evidence gaps but remain insufficient for answering the query.
Contradiction	The observations conflict with accumulated evidence or current hypotheses, indicating that the corresponding assumption or search direction requires revision.
Neutral	The observations provide no substantive clues for resolving the remaining gaps, suggesting that the current search direction should be reconsidered or redirected.

Here, r_t explains the evidential relation represented by $V_{\mathrm{NLI}}^{(t)}$, while $q_{\mathrm{clue}}^{(t)}$ guides the update of the caption context for the next iteration according to Eq. 1. According to how O_t contributes to the current reasoning state, $V_{\mathrm{NLI}}^{(t)}$ is categorized into four NLI signals, as summarized in Table 1.

State Feedback and Trajectory Regulation. The verification outcome, its reasoning, and the associated clue are returned to the planner together with the current observation. The corresponding search and verification record is appended chronologically to the history buffer:

$$\mathcal{B}_{t+1} = \mathcal{B}_t \oplus \left(\mathcal{K}_t, \mathcal{U}_t, q_{\mathrm{search}}^{(t)}, O_t, V_{\mathrm{NLI}}^{(t)}, r_t\right), \tag{6}$$

where $\oplus$ denotes appending the current iteration record to the history buffer in chronological order.

Global entailment terminates exploration and initiates answer generation. Otherwise, the planner uses $\mathcal{B}_{t+1}$ to update the accumulated evidence $\mathcal{K}_{t+1}$ and unresolved gaps $\mathcal{U}_{t+1}$. Partial entailment supports deeper examination of the current direction, contradiction motivates revision of the underlying assumption, and neutral feedback indicates the observation contributes no relevant evidence, prompting reconsideration of the current search direction. Rather than prescribing fixed actions, these signals provide an evidential basis for adjusting the query, temporal scope, and observation scale. The updated state informs the next search strategy, completing the plan-observe-verify loop. If global entailment is not reached within the iteration budget $T_{\max}$, visual exploration terminates and the final answer is derived from accumulated evidence and available coarse video context.

4 Experiments

4.1 Experimental Results

Datasets. We evaluate VidDeduce on four complementary video understanding benchmarks. **LVBench** (Wang et al., 2025a) evaluates temporal grounding and the retrieval of sparsely distributed evidence from hour-scale videos across diverse domains. We use the full benchmark, comprising 1,549 multiple-choice questions over 103 videos. **LongVideoBench** (Wu et al., 2024) focuses on referring reasoning over detailed multimodal information in videos of varying durations. We evaluate on its long-duration validation split, which contains 564 questions over 188 videos lasting 15-60 minutes. **Video-MME** (Fu et al., 2025) covers six major visual domains and evaluates video understanding across a broad range of temporal durations. We use its long split, comprising 900 questions over 300 videos lasting 30-60 minutes. **EgoSchema** (Mangalam et al., 2023) evaluates long-horizon reasoning over three-minute egocentric videos. We report results on the standard validation subset of 500 questions. We do not use auxiliary subtitles or ASR transcripts for any benchmark.

Implementation Details. We use standard multiple-choice accuracy as the evaluation metric across all benchmarks. GPT-5.3 (OpenAI, 2026) serves as the reasoning backbone for the planner, observer, and verifier, while Qwen2.5-VL-72B (Bai et al., 2025) performs visual perception. We use Qwen3-Embedding-0.6B (Zhang et al., 2025b) for dense caption retrieval, which returns the top-k candidates with $k = 16$. The maximum number of plan-observe-verify iterations is set to $T_{\max} = 5$. Benchmark-specific captioning and visual inspection settings are detailed in the Appendix.

Main Results. We compare VidDeduce with proprietary VLMs (Team et al., 2024; Gemini Team et al., 2023; OpenAI, 2024; 2025a), open-source VLMs (Zhang et al., 2025a; Li et al., 2026b; Chen

Table 2: Performance comparison with commercial VLMs, open-source VLMs, and agent-based systems on four video understanding benchmarks. The best two results are highlighted in **bold**.

Method	LVBench (Overall)	Video-MME (Long)	LongVideoBench (Val-Long)	EgoSchema (Val)
Commercial VLMs				
Gemini-1.5-Pro	33.1	67.4	58.6	–
Gemini-2.0-Flash	48.3	63.0	45.7	71.2
GPT-4o	48.9	65.3	60.9	70.4
GPT-4.1	51.9	63.1	64.6	72.2
o3	57.1	64.7	60.6	63.2
GPT-5	59.8	**68.4**	61.8	73.8
Open-Source VLMs				
VideoLLaMA3-7B	45.3	54.9	–	–
VideoChat-Flash-7B	48.2	55.6	–	–
InternVL2.5-72B	43.6	62.6	–	–
Qwen2.5-VL-72B	47.7	63.1	54.2	75.4
AdaReTaKe-72B	53.3	65.0	–	–
Agent-based Systems				
VideoAgent	29.3	–	–	63.2
VideoTree	28.8	54.2	–	–
VCA	41.3	56.3	–	73.6
Ego-R1	–	64.9	–	68.2
MR. Video	60.8	61.8	61.6	73.0
Logic-in-Frames	–	55.2	54.2	–
DVD	**74.2**	67.3	**68.6**	76.6
VideoLucy	58.8	66.8	–	–
LensWalk	59.4	66.7	67.3	**77.2**
VidDeduce (Ours)	**65.2**[1]	**71.3**	**69.9**	**78.6**

et al., 2024; Wang et al., 2025b), and agent-based video understanding systems (Tian et al., 2025). As shown in Table 2, VidDeduce achieves an average accuracy of 71.3% across the four benchmarks. With Qwen2.5-VL-72B serving as a shared visual backbone, VidDeduce improves the average accuracy across the four benchmarks by 11.2% over the standalone VLM and by 3.6% over LensWalk (Li et al., 2026a). These gains confirm the effectiveness of verification-guided evidence acquisition in our framework. On Video-MME, VidDeduce surpasses DVD (Zhang et al., 2026) and MR. Video (Pang & Wang, 2026) by 4.0% and 9.5%, respectively. The benefits extend to LVBench and LongVideoBench, on which VidDeduce improves over GPT-5 (OpenAI, 2025b) by 5.4% and 8.1%, respectively. NLI feedback provides a reliable basis for subsequent search by indicating which clues should be retained and when observations contradict or fail to advance the current reasoning. Queries and temporal focus can therefore be revised as needed, while the observation scale shifts between broad localization and focused inspection as evidence requirements evolve. VidDeduce further exceeds o3 (OpenAI, 2025c) by 15.4% on EgoSchema, indicating that verification-guided exploration also supports temporally dependent reasoning in shorter egocentric videos.

4.2 Ablation and Analysis

Component Ablations. VidDeduce integrates dual-scale visual execution, NLI-based verification, and temporal search guidance. To isolate their contributions, we evaluate four ablated variants on Video-MME (Long), as shown in Table 3. Restricting the observer to fine or coarse inspection decreases accuracy by 2.11% and 1.55%, respectively. Fine-only inspection limits each observation to a single continuous interval, weakening comparison across temporally separated candidates, whereas coarse-only inspection provides broad coverage but is less effective at resolving detailed actions and visual attributes. Replacing explicit NLI verification with confidence results in a 1.44% decline, demonstrating the value of structured evidence feedback for trajectory regulation. Removing temporal guidance causes the degradation of 3.33%, indicating that restricting or prioritizing candidate intervals is important for reducing irrelevant context and concentrating visual inspection. The complete framework achieves the best performance, confirming proposed components provide complementary benefits.

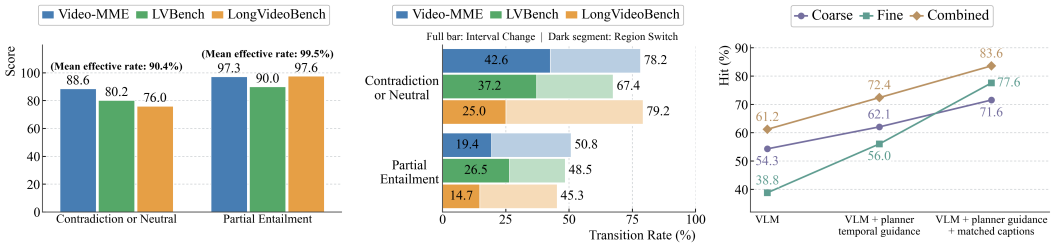
[1] We analyze the performance gap with DVD in Appendix A.3.

Table 3: Ablation of visual execution and trajectory regulation components on Video-MME (Long).

Visual Execution		Trajectory Regulation		Acc. (%)
Coarse	Fine	NLI Verif.	Temp. Guid.	
	✓	✓	✓	69.22 (↓ 2.11)
✓		✓	✓	69.78 (↓ 1.55)
✓	✓		✓	69.89 (↓ 1.44)
✓	✓	✓		68.00 (↓ 3.33)
✓	✓	✓	✓	**71.33**

Table 4: Backbone generalization on a subset of Video-MME (Long).

VLM	LLM	Acc. (%)
Qwen2.5-VL-72B	–	60.4
	Qwen3.6-35B-A3B	67.4 (↑ 7.0)
	GPT-4o	60.7 (↑ 0.3)
	GPT-5.3	**71.5** (↑ 11.1)
InternVL2.5-8B	–	53.3
	Qwen3.6-35B-A3B	60.4 (↑ 7.1)
	GPT-4o	56.3 (↑ 3.0)
	GPT-5.3	**65.9** (↑ 12.6)



(a) Query Adaptation **(b) VLM Temporal Adjustment** **(c) Temporal Grounding**

Figure 3: NLI-guided evidence acquisition. **(a)** Query adaptation scores and effective rates. **(b)** Changes between consecutive VLM observation intervals. Interval Change denotes shifts in observation intervals; Region Switch denotes shifts to entirely separate regions (dark segments). **(c)** LVBench Hit with visual observations, planner guidance, and caption cues.

Generalization Across Model Backbones. We assess the adaptability of our framework across various language and visual backbones on a 270-question subset constructed by randomly sampling three videos from each of the 30 Video-MME categories. As shown in Table 4, with Qwen2.5-VL-72B (Bai et al., 2025) as the visual executor, Qwen3.6-35B-A3B (Team, 2026) achieves 67.4%, while GPT-5.3 (OpenAI, 2026) improves accuracy from the standalone result of 60.4% to 71.5%. Using the smaller InternVL2.5-8B (Chen et al., 2024), Qwen3.6-35B-A3B obtains 60.4%, whereas GPT-4o (OpenAI, 2024) and GPT-5.3 raise accuracy from 53.3% to 56.3% and 65.9%, respectively. These results demonstrate that VidDeduce generalizes across backbone capacities, with stronger reasoning models enhancing evidence tracking, search planning, and verification to compensate for weaker visual perception through iterative evidence refinement.

Search Adaptation and Grounding. We evaluate the effectiveness of NLI-guided query revision and temporal exploration. Qwen3-8B (Yang et al., 2025a) rates query adaptation on a five-level scale mapped to a 0-100 score, with ratings of at least level 3 considered effective. As shown in Fig. 3(a), most queries adapt effectively to feedback across all three benchmarks, with partial entailment yielding stronger and more consistent query adaptation. In Fig. 3(b), we compare the temporal intervals inspected by the VLM across consecutive observations. Contradictory or neutral feedback more often prompts interval changes, including shifts to entirely separate regions, whereas partial entailment is more likely to guide further exploration of the current region. We also find that subsequent verification yields partial or global entailment in 59.8% of transitions following contradiction or neutral feedback and 90.9% following partial entailment. These results support the effectiveness of NLI feedback in recovering supporting evidence after conflicting or inconclusive observations while preserving supported clues for further refinement.

We further evaluate temporal grounding using LVBench's annotated evidence intervals (Wang et al., 2025a), defining Hit as the percentage of observations whose temporal intervals overlap these annotations. As shown in Fig. 3(c), 54.3% of coarse VLM observations overlap the annotated evidence, compared with 38.8% of fine VLM inspections, which cover only 2.9% of video duration on average. Combining their observed intervals raises this proportion to 61.2%, and including the associated planner windows increases it to 72.4%. Adding caption intervals retrieved within the corresponding

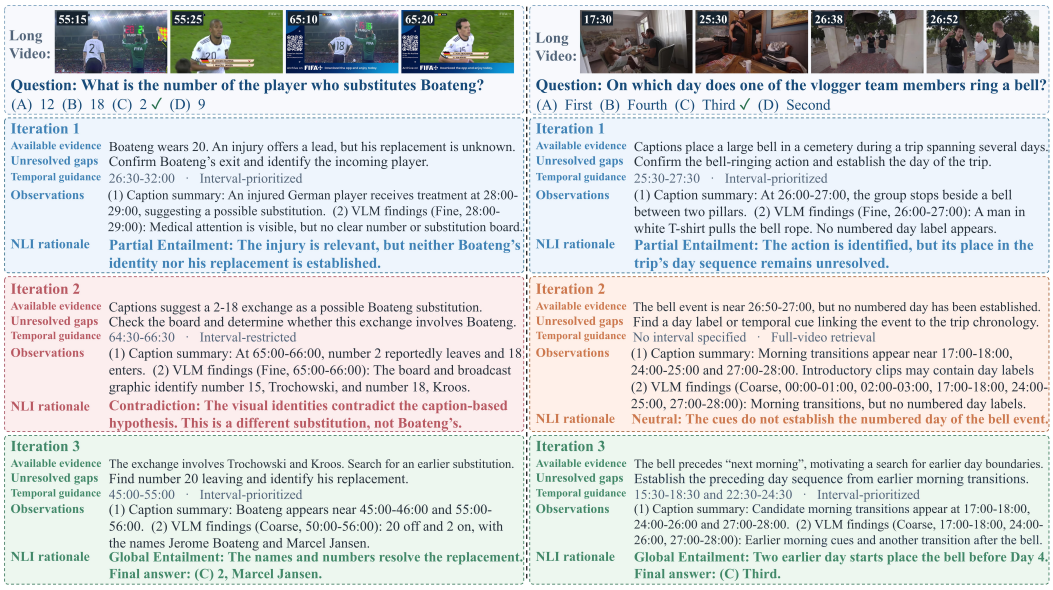


Figure 4: Verification-guided search examples. Left: contradictory evidence prompts revision of a substitution hypothesis. Right: neutral feedback prompts further examination of the day sequence.

observation steps further increases it to 83.6%. These results highlight the complementary value of dual-scale observations and temporal cues in locating evidence.

Search Efficiency. VidDeduce averages 5.82 LLM calls per question on Video-MME (Long), while LensWalk (Li et al., 2026a) reports 6.8 reasoning steps, totaling approximately 12.6 LLM calls including subject-memory updates. On LVBench, VidDeduce averages 5.43 LLM calls versus DVD's 7.3 reasoning calls (Zhang et al., 2026). During active search, VidDeduce processes 162.1, 143.0, and 39.6 frames per question on Video-MME (Long), LVBench, and EgoSchema, respectively, compared with 387.1, 290.3, and 89.7 for LensWalk. On LongVideoBench (Long), VidDeduce uses 77.2 frames versus 32 for Logic-in-Frames (Guo et al., 2026), while achieving 69.9% accuracy compared with 54.2%. These comparisons show fewer reasoning calls and active frames than the selected agent configurations, alongside a trade-off between visual budget and accuracy under sparse sampling.

Qualitative Analysis of Search Trajectories. Fig. 4 illustrates how NLI feedback revises search trajectories. In the substitution example, contradictory visual evidence exposes a mismatch between the caption-based hypothesis and the observed players, prompting the planner to discard the mistaken association and search an earlier interval for Boateng's replacement. In the bell example, partial entailment establishes the event while leaving its day unresolved. The subsequent neutral judgment prompts a shift from searching for explicit day labels to reconstructing the day sequence through earlier morning transitions. Both cases show how verification turns unresolved evidence into concrete search actions.

5 Conclusion

In this work, we proposed VidDeduce, a training-free multi-agent framework for long-form video understanding through verification-guided evidence acquisition. By distinguishing partial support, contradiction, and uninformative observations, NLI feedback enables the planner to refine useful clues, revise unsupported hypotheses, and adjust subsequent searches. Retrieved captions and dual-scale visual inspection translate these decisions into targeted observations, connecting broader temporal information with question-specific evidence. Experiments across four benchmarks demonstrate the effectiveness of VidDeduce, while ablations and search analyses support the contribution of verification to evidence acquisition. These findings suggest that effective video reasoning requires not only locating relevant content, but also assessing how new observations support or challenge the current reasoning and using that assessment to determine what to examine next.

AI USE STATEMENT

In this work, we used generative AI tools for refining the language, improving readability, ensuring clarity of the manuscript, and code debugging. We have not used generative AI tools for ideation, research methodology, data analysis, or experimental design, and mathematical theorem proving is not applicable to this work. Additionally, we used generative AI tools for basic sentence rephrasing and grammar checking. We have meticulously reviewed all AI-assisted work by manually cross-checking the generated text against our original drafts to ensure scientific accuracy, preserve our original intent, and prevent plagiarism. We take full responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work adheres strictly to the ICLR Code of Ethics. In this study, no human subjects or animal experimentation were involved, and no personally identifiable information (PII) was utilized. All datasets used in our experiments, including LVBench, LongVideoBench, Video-MME, and EgoSchema, were sourced in full compliance with relevant usage guidelines and licensing agreements, ensuring no violation of privacy. Furthermore, we have taken deliberate care throughout our research process to evaluate and mitigate potential biases or discriminatory outcomes within our proposed methodologies. We remain committed to maintaining transparency and scientific integrity.

REPRODUCIBILITY STATEMENT

To support reproducibility, we document the implementation details of VidDeduce in Section 4.1 and Appendix A.1, including model configurations, inference infrastructure, benchmark splits, and video sampling settings. Additionally, the video understanding benchmarks used in our experiments, including LVBench, LongVideoBench, Video-MME, and EgoSchema, are publicly available, with the evaluation splits specified in Section 4.1. The source code will be made publicly available upon acceptance to facilitate reproduction of our results and further research.

REFERENCES

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

Yukang Chen, Fuzhao Xue, Dacheng Li, Qinghao Hu, Ligeng Zhu, Xiuyu Li, Yunhao Fang, Haotian Tang, Shang Yang, Zhijian Liu, et al. LongVILA: Scaling long-context visual language models for long videos. In *International Conference on Learning Representations*, volume 2025, pp. 18227–18246, 2025.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.

Dorottya Demszky, Kelvin Guu, and Percy Liang. Transforming question answering datasets into natural language inference datasets. *arXiv preprint arXiv:1809.02922*, 2018.

Yue Fan, Xiaojian Ma, Rujie Wu, Yuntao Du, Jiaqi Li, Zhi Gao, and Qing Li. VideoAgent: A memory-augmented multimodal agent for video understanding. In *European Conference on Computer Vision*, pp. 75–92. Springer, 2024.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-MME: The first-ever comprehensive evaluation benchmark of multi-modal LLMs in video analysis. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24108–24118. IEEE, 2025.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Weiyu Guo, Ziyang Chen, Shaoguang Wang, Jianxiang He, Yijie Xu, Jinhui Ye, Ying Sun, and Hui Xiong. Logic-in-Frames: Dynamic keyframe search via visual semantic-logical verification for long video understanding. *Advances in Neural Information Processing Systems*, 38:124389–124422, 2026.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547, 2019.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pp. 611–626, 2023.

Keliang Li, Yansong Li, Hongze Shen, Mengdi Liu, Hong Chang, and Shiguang Shan. LensWalk: Agentic video understanding by planning how you see in videos. *arXiv preprint arXiv:2603.24558*, 2026a.

Xinhao Li, Yi Wang, Jiashuo Yu, Xiangyu Zeng, Yuhan Zhu, Haian Huang, Jianfei Gao, Kunchang Li, Yinan He, Chenting Wang, et al. VideoChat-Flash: Hierarchical compression for long-context video modeling. In *International Conference on Learning Representations*, volume 2026, pp. 109089–109117, 2026b.

Yanwei Li, Chengyao Wang, and Jiaya Jia. LLaMA-VID: An image is worth 2 tokens in large language models. In *European Conference on Computer Vision*, pp. 323–340. Springer, 2024.

Zhan Ling, Yunhao Fang, Xuanlin Li, Zhiao Huang, Mingu Lee, Roland Memisevic, and Hao Su. Deductive verification of chain-of-thought reasoning. *Advances in Neural Information Processing Systems*, 36:36407–36433, 2023.

Xiangrui Liu, Yan Shu, Zheng Liu, Ao Li, Yang Tian, and Bo Zhao. Video-XL-Pro: Reconstructive token compression for extremely long video understanding. *arXiv preprint arXiv:2503.18478*, 2025.

Yongdong Luo, Xiawu Zheng, Guilin Li, Shukang Yin, Haojia Lin, Chaoyou Fu, Jinfa Huang, Jiayi Ji, Fei Chao, Jiebo Luo, et al. Video-RAG: Visually-aligned retrieval-augmented long video comprehension. *Advances in Neural Information Processing Systems*, 38:168008–168033, 2026.

Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. EgoSchema: A diagnostic benchmark for very long-form video language understanding. *Advances in Neural Information Processing Systems*, 36:46212–46244, 2023.

Juhong Min, Shyamal Buch, Arsha Nagrani, Minsu Cho, and Cordelia Schmid. MoReVQA: Exploring modular reasoning models for video question answering. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13235–13245. IEEE, 2024.

OpenAI. GPT-4o system card, 2024. URL https://arxiv.org/abs/2410.21276.

OpenAI. Introducing GPT-4.1 in the API. https://openai.com/index/gpt-4-1/, 2025a. Accessed: 2025-04-14.

OpenAI. GPT-5 system card. https://cdn.openai.com/gpt-5-system-card.pdf, 2025b.

OpenAI. Introducing OpenAI o3 and o4-mini. https://openai.com/index/introducing-o3-and-o4-mini/, 2025c. Accessed: 2025-05-15.

OpenAI. GPT-5.3 Instant system card. https://openai.com/index/gpt-5-3-instant-system-card/, 2026.

Ziqi Pang and Yu-Xiong Wang. MR. Video: MapReduce as an effective principle for long video understanding. *Advances in Neural Information Processing Systems*, 38:6640–6672, 2026.

Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. TimeChat: A time-sensitive multimodal large language model for long video understanding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14313–14323. IEEE, 2024.

Xiaoqian Shen, Yunyang Xiong, Changsheng Zhao, Lemeng Wu, Jun Chen, Chenchen Zhu, Zechun Liu, Fanyi Xiao, Balakrishnan Varadarajan, Florian Bordes, et al. LongVU: Spatiotemporal adaptive compression for long video-language understanding. *arXiv preprint arXiv:2410.17434*, 2024.

Xiaoqian Shen, Wenxuan Zhang, Jun Chen, and Mohamed Elhoseiny. Vgent: Graph-based retrieval-reasoning-augmented generation for long video understanding. *Advances in Neural Information Processing Systems*, 38:100805–100830, 2026.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36:8634–8652, 2023.

Yan Shu, Zheng Liu, Peitian Zhang, Minghao Qin, Junjie Zhou, Zhengyang Liang, Tiejun Huang, and Bo Zhao. Video-XL: Extra-long vision language model for hour-scale video understanding. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26160–26169. IEEE, 2025.

Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, et al. MovieChat: From dense token to sparse memory for long video understanding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18221–18232. IEEE, 2024.

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

Qwen Team. Qwen3.6-35b-a3b: Agentic coding power, now open to all, 2026.

Shulin Tian, Ruiqi Wang, Hongming Guo, Penghao Wu, Yuhao Dong, Xiuying Wang, Jingkang Yang, Hao Zhang, Hongyuan Zhu, and Ziwei Liu. Ego-R1: Chain-of-tool-thought for ultra-long egocentric video reasoning. *arXiv preprint arXiv:2506.13654*, 2025.

Weihan Wang, Zehai He, Wenyi Hong, Yean Cheng, Xiaohan Zhang, Ji Qi, Ming Ding, Xiaotao Gu, Shiyu Huang, Bin Xu, et al. LVBench: An extreme long video understanding benchmark. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22958–22967. IEEE, 2025a.

Xiao Wang, Qingyi Si, Shiyu Zhu, Jianlong Wu, Li Cao, and Liqiang Nie. AdaReTaKe: Adaptive redundancy reduction to perceive longer for video-language understanding. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 5417–5432, 2025b.

Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. VideoAgent: Long-form video understanding with large language model as agent. In *European Conference on Computer Vision*, pp. 58–76. Springer, 2024.

Ying Wang, Yanlai Yang, and Mengye Ren. LifelongMemory: Leveraging LLMs for answering queries in long-form egocentric videos. *arXiv preprint arXiv:2312.05269*, 2023.

Ziyang Wang, Shoubin Yu, Elias Stengel-Eskin, Jaehong Yoon, Feng Cheng, Gedas Bertasius, and Mohit Bansal. VideoTree: Adaptive tree-based video representation for LLM reasoning on long videos. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3272–3282. IEEE, 2025c.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. LongVideoBench: A benchmark for long-context interleaved video-language understanding. *Advances in Neural Information Processing Systems*, 37:28828–28857, 2024.

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

Zeyuan Yang, Delin Chen, Xueyang Yu, Maohao Shen, and Chuang Gan. VCA: Video curious agent for long video understanding. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20168–20179. IEEE, 2025b.

Zongxin Yang, Guikun Chen, Xiaodi Li, Wenguan Wang, and Yi Yang. DoraemonGPT: Toward understanding dynamic scenes with large language models (exemplified as a video agent). *arXiv preprint arXiv:2401.08392*, 2024.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of Thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems*, 36:11809–11822, 2023.

Boqiang Zhang, Kehan Li, Zesen Cheng, Zhiqiang Hu, Yuqian Yuan, Guanzheng Chen, Sicong Leng, Yuming Jiang, Hang Zhang, Xin Li, et al. VideoLLaMA 3: Frontier multimodal foundation models for image and video understanding. *arXiv preprint arXiv:2501.13106*, 2025a.

Ce Zhang, Taixi Lu, Md Mohaiminul Islam, Ziyang Wang, Shoubin Yu, Mohit Bansal, and Gedas Bertasius. A simple LLM framework for long-range video question-answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 21715–21737, 2024.

Xiaoyi Zhang, Zhaoyang Jia, Zongyu Guo, Jiahao Li, Bin Li, Houqiang Li, and Yan Lu. Deep Video Discovery: Agentic search with tool use for long-form video understanding. *Advances in Neural Information Processing Systems*, 38:89863–89895, 2026.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, et al. Qwen3 Embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025b.

Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Zhengyang Liang, Shitao Xiao, Minghao Qin, Xi Yang, Yongping Xiong, Bo Zhang, et al. MLVU: Benchmarking multi-task long video understanding. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13691–13701. IEEE, 2025.

Jialong Zuo, Yongtai Deng, Lingdong Kong, Jingkang Yang, Rui Jin, Yiwei Zhang, Nong Sang, Liang Pan, Ziwei Liu, and Changxin Gao. VideoLucy: Deep memory backtracking for long video understanding. *Advances in Neural Information Processing Systems*, 38:25660–25691, 2026.

A Appendix

A.1 Implementation Details

Hardware and Inference Infrastructure. Local inference runs on four NVIDIA A800 GPUs, with Qwen2.5-VL-72B (Bai et al., 2025) served through vLLM (Kwon et al., 2023) for visual perception. Planning, observation scheduling, and verification share a GPT-5.3 reasoning backbone accessed through the OpenAI API. For caption retrieval, Qwen3-Embedding-0.6B (Zhang et al., 2025b) encodes clip descriptions into dense vectors indexed by FAISS (Johnson et al., 2019); each caption retains its associated temporal metadata. Captions and their embeddings are computed once per video and reused across questions.

Distinct Settings for Different Benchmarks. We evaluate on the LVBench (Wang et al., 2025a), the long split of Video-MME (Fu et al., 2025), the long-duration validation split of LongVideoBench (Wu et al., 2024), and the validation set of EgoSchema (Mangalam et al., 2023). We adapt caption duration, visual inspection budgets, and frame sampling rates to the temporal characteristics of each benchmark, as summarized in Table 5. Caption generation uses 60-second clips for LVBench and Video-MME, 30-second clips for LongVideoBench, and 10-second clips for EgoSchema. Coarse inspection distributes its temporal budget across selected segments, whereas fine inspection examines a continuous interval. For EgoSchema, both tools operate at 2 FPS but differ in their temporal coverage pattern. All benchmarks share the same planning and verification procedure, without auxiliary subtitles or ASR transcripts.

Table 5: Temporal coverage and sampling rates for caption generation and dual-scale visual inspection. Coarse coverage denotes the total duration of the selected segments.

Benchmark	Caption	Coarse	Fine
LVBench	60 s / 2.0 FPS	$\leq$360 s / 0.5 FPS	60 s / 2.0 FPS
Video-MME	60 s / 2.0 FPS	$\leq$360 s / 0.5 FPS	60 s / 2.0 FPS
LongVideoBench	30 s / 2.0 FPS	$\leq$180 s / 0.5 FPS	30 s / 2.0 FPS
EgoSchema	10 s / 2.0 FPS	$\leq$60 s / 2.0 FPS	60 s / 2.0 FPS

Inference Procedure. We summarize the inference procedure of VidDeduce in Algorithm 1, including the initial caption-based answer attempt, iterative planning, observation and verification, and final deduction when the search budget is exhausted. Table 6 lists the used notation.

Table 6: Notation used in Algorithm 1.

Symbol	**Description**
V, Q, A	Input video, question, and output answer.
$\mathcal{D}$, $T_{\max}$	Coarse caption database and maximum number of iterations.
$\mathcal{B}$	History of search and verification records.
$\mathcal{K}, \mathcal{U}$	Accumulated evidence and unresolved gaps.
$\mathcal{C}_Q$	Captions selected under question-derived temporal guidance.
$\mathcal{C}, \mathcal{C}_{\text{match}}$	Planner caption context and captions retrieved for the current search.
$\mathcal{T}_Q, \mathcal{T}_{\text{gap}}$	Question-derived and gap-derived temporal scopes.
f_Q, f_{gap}	Temporal selection flags: 1 (restricted) retains only clips overlapping the specified intervals; 0 (prioritized) selects overlapping clips first and supplements them with other matches if needed.
q_{search}, q_{visual}, q_{clue}	Planner search query, visual inspection query, and verifier-derived search clue.
s, τ	Caption summary and selected visual tool (coarse or fine).
V_{selected}, O	Video segments selected for inspection and the combined observation.
V_{NLI}, r	NLI verification signal and its supporting reasoning.

Algorithm 1: Verification-Guided Search in VidDeduce

Input: Video V, question Q, caption database $\mathcal{D}$, iteration budget $T_{\max}$
Output: Answer A

$(\mathcal{T}_Q, f_Q) \leftarrow \mathrm{ParseTime}(Q, V)$ `// Question-derived temporal guidance`
$\mathcal{C}_Q \leftarrow \mathrm{SelectCaptions}(\mathcal{D}, \mathcal{T}_Q, f_Q)$ `// Temporal selection`
$A \leftarrow \mathrm{DirectAnswer}(Q, \mathcal{C}_Q)$ `// Caption-based answer`
if *A is valid and judged sufficiently supported* **then**
  return A
$\mathcal{B} \leftarrow [\,]$ `// Initialize history`
$\mathcal{C} \leftarrow \mathrm{Retrieve}(\mathcal{D}, Q, \mathcal{T}_Q, f_Q)$ `// Initialize planner caption context`
for $t = 1, \ldots, T_{\max}$ **do**
  $(\mathcal{K}, \mathcal{U}, q_{\text{search}}, \mathcal{T}_{\text{gap}}, f_{\text{gap}}) \leftarrow \mathrm{Plan}(Q, \mathcal{B}, \mathcal{C})$
  $\mathcal{T}_{\text{gap}} \leftarrow \mathcal{T}_Q$ if unspecified `// Default temporal scope`
  $\mathcal{C}_{\text{match}} \leftarrow \mathrm{Retrieve}(\mathcal{D}, q_{\text{search}}, \mathcal{T}_{\text{gap}}, f_{\text{gap}})$
  $(s, \tau, V_{\text{selected}}, q_{\text{visual}}) \leftarrow \mathrm{Observe}(V, \mathcal{C}_{\text{match}}, q_{\text{search}}, \mathcal{U})$
  $O \leftarrow \mathrm{Concat}\big(s, \mathrm{VLM}_\tau(V_{\text{selected}}, q_{\text{visual}})\big)$ `// Caption and visual evidence`
  $(V_{\text{NLI}}, r, q_{\text{clue}}) \leftarrow \mathrm{Verify}(Q, \mathcal{K}, \mathcal{U}, O)$ `// NLI feedback`
  $\mathcal{B} \leftarrow \mathrm{Append}(\mathcal{B}, (\mathcal{K}, \mathcal{U}, q_{\text{search}}, O, V_{\text{NLI}}, r))$ `// Store iteration record`
  if $V_{\text{NLI}} =$ GLOBAL ENTAILMENT *and the answer is valid* **then**
    return the verifier's answer
  if $q_{\text{clue}} \neq \varnothing$ **then**
    $\mathcal{C} \leftarrow \mathrm{Retrieve}(\mathcal{D}, q_{\text{clue}}, \mathcal{T}_Q, f_Q)$ `// Refresh planner caption context`
$A \leftarrow \mathrm{FinalDeduction}(Q, \mathcal{K}, \mathcal{U}, \mathcal{C}_Q)$ `// Budget exhausted`
return A

Agent Prompts. We provide the prompts for caption generation, temporal-reference extraction, planning, observation scheduling, NLI verification, and final deduction in Tables 7–13. The corresponding output formats are summarized in Table 14. Prompts are condensed for clarity.

The captioning prompt produces clip descriptions that provide temporal context for subsequent search, while temporal-reference extraction identifies constraints expressed in the question. Guided by this context and prior NLI feedback, the planner revises accumulated evidence, clarifies unresolved gaps, and specifies the next search. The observer summarizes relevant caption clues and selects coarse or fine visual inspection according to the remaining evidence requirements. The verifier then assesses the resulting observations to determine whether the evidence supports an answer or requires further investigation. If the search budget is exhausted, the final-deduction prompt synthesizes the accumulated evidence and caption context to select an answer.

Table 7: Prompt for offline clip-caption generation.

System

```
You are a helpful assistant.
```

User

```
Please observe and understand the given video carefully. Describe all
the details of this video as comprehensively as possible in a smooth
and coherent passage. Do not omit any details or prominent
information. In addition, if there are any texts, subtitles, text
overlays, or voice-overs in the video, you must explicitly and in
detail describe them.
```

Table 8: Prompt for extracting temporal constraints from the question.

Task

```
Extract temporal constraints for video question answering.
```

Inputs

```
Question: {question}
Video duration: {duration}
```

Instructions

```
1. Extract only explicit numeric timestamps as [start, end] pairs in
   HH:MM:SS or MM:SS format. Return [] if none are provided. Do not
   convert relative expressions into numbers.
2. Set start, middle, and end flags only for references to the video's
   structure. Exclude event-relative expressions and ordinal
   references such as "the first time he appears."
3. Set use_time_ref_only=true for a strict temporal boundary.
   Otherwise, prioritize the reference region while allowing retrieval
   from elsewhere in the video.
```

Output

```
Return time_reference, video_structure_flags, use_time_ref_only, and
rationale in the prescribed schema.
```

Table 9: Planner prompt for revising the evidence state and specifying the next search.

Task

```
Maintain the evidence state and formulate the next search.
```

Inputs

```
Question: {question}
Video duration / clip length: {duration} / {clip_secs}
Temporal reference: {time_reference}
Caption context: {global_context}
Verification history: {verifier_feedback}
```

Instructions

```
1. Review caption context and prior verification feedback:
   PARTIAL_ENTAILMENT: retain useful clues and refine the gaps.
   CONTRADICTION: revise conflicting assumptions and redirect the
   search accordingly.
   NEUTRAL: review unsuccessful requests and change the visual
   keywords, search direction, or temporal focus.
2. Update aligned_knowns and unknown_gaps, preserving the subjects and
   spatial constraints specified in the question. Even when captions
   suggest an answer, formulate a gap requiring verification through
   direct visual inspection.
3. Write search_query as an objective visual description targeting the
   unresolved gap and the relevant subject.
4. Specify gap_time_reference and use_gap_time_reference_only. Set the
   flag to true to restrict retrieval to these intervals, or false to
   prioritize them while allowing additional matches from elsewhere in
   the video.
```

Output

```
Return aligned_knowns, unknown_gaps, search_query, gap_time_reference,
and use_gap_time_reference_only.
```

Table 10: Observer prompt for selecting a visual tool and inspection intervals.

Task

```
Address the planner's evidence gap through visual inspection.
```

Inputs

```
Unresolved gaps: {unknown_gaps}
Planner query: {planner_search_query}
Retrieved captions: {search_results}
```

Instructions

```
1. Summarize useful caption clues in search_insights.
2. Select exactly one visual tool:
   fine_vlm_tool: inspect one continuous interval of up to 60 seconds
   for detailed examination of a candidate moment.
   coarse_vlm_tool: inspect up to six separate clip intervals for
   broader examination of candidate regions.
3. Copy interval boundaries exactly from the retrieved results.
   Preserve timestamp formats.
4. Formulate one visual query addressing the unresolved gap. Apply
   this query to all segments processed by the tool.
```

Output

```
Return tool_name and the selected tool's parameter object:
search_insights, query, and time_range or time_ranges. Leave the
unused tool's parameter object null.
```

Table 11: Verifier prompt for NLI assessment and subsequent search feedback.

Task

```
Assess new observations against the intended evidence gap and the
accumulated evidence.
```

Inputs

```
Question: {question}
Accumulated evidence: {aligned_knowns}
Unresolved gaps: {unknown_gaps}
New observations: {observations}
```

Decision rules

```
GLOBAL_ENTAILMENT: The observations resolve the remaining gaps and,
together with accumulated evidence, provide sufficient support to
answer the question.

PARTIAL_ENTAILMENT: The observations provide relevant evidence or
clues that partially address the current gaps.

CONTRADICTION: The observations explicitly conflict with an existing
premise or hypothesis. Missing information alone does not constitute a
contradiction.

NEUTRAL: The observations provide no meaningful information about the
gaps without explicitly contradicting the current hypothesis.
```

Output

```
Return nli_status and reasoning. Provide final_answer only for
GLOBAL_ENTAILMENT. When the observation provides a useful clue for
further search, express it in search_clue as an objective visual
description based on the observation. Otherwise, return an empty
search_clue.
```

Table 12: Prompt for final deduction after the search budget is exhausted.

Task

```
Select an answer using the accumulated evidence and caption timeline.
```

Inputs

```
Question: {question}
Accumulated evidence: {aligned_knowns}
Unresolved gaps: {unknown_gaps}
Caption timeline: {filtered_captions}
```

Instructions

```
1. Prioritize accumulated evidence as the factual basis.
2. Use the caption timeline to establish event order and provide
   broader context for the remaining gaps.
3. Infer the most likely answer or eliminate options inconsistent with
   the accumulated evidence.
4. Select one option: A, B, C, or D. Do not return None or Unknown.
```

Output

```
Return final_answer and reasoning in the prescribed schema.
```

Table 13: Condensed prompt for coarse and fine visual inspection of selected temporal intervals.

System

```
You are a helpful assistant.
```

User

```
Inspect the supplied video content from {time_range}.

Please observe and understand the given video carefully. Describe all
the details of this video as comprehensively as possible in a smooth
and coherent passage. Do not omit any details or prominent
information. In addition, if there are any texts, subtitles, text
overlays, or voice-overs in the video, you must explicitly and in
detail describe them.

In addition: {query}
```

Table 14: Structured output formats for temporal-reference extraction and the reasoning agents. Field types and admissible values are shown in abbreviated schema notation.

Temporal-reference extraction

```
time_reference: list of [start, end] timestamp pairs
video_structure_flags:
  {start: boolean, middle: boolean, end: boolean}
use_time_ref_only: boolean
rationale: string
```

Planner

```
aligned_knowns: string
unknown_gaps: string
search_query: string
gap_time_reference: list of [start, end] timestamp pairs
use_gap_time_reference_only: boolean
```

Observer

```
tool_name: fine_vlm_tool | coarse_vlm_tool
fine_vlm_parameters:
  {search_insights, time_range, query} | null
coarse_vlm_parameters:
  {search_insights, time_ranges, query} | null
```

Verifier

```
nli_status: GLOBAL_ENTAILMENT | PARTIAL_ENTAILMENT |
            CONTRADICTION | NEUTRAL
reasoning: string
search_clue: string
final_answer: A | B | C | D | null
```

Final deduction

```
final_answer: A | B | C | D
reasoning: string
```

A.2 Additional Experiments

Query Adaptation Evaluation. We detail the Qwen3-8B evaluation of query adaptation. The evaluator receives the original question, the ordered query and verification history, and the next planner search query. It assesses how the proposed query responds to the available feedback without access to subsequent observations or final-answer correctness. Tables 15 and 16 present the evaluator instructions and scoring rubric, respectively.

Each assessable query receives a holistic rating from 1 to 5. For N ratings ℓ_i, the adaptation score and effective rate are

$$S = 25\left(\frac{1}{N}\sum_{i=1}^{N}\ell_i - 1\right), \qquad E = \frac{100}{N}\sum_{i=1}^{N}\mathbf{1}[\ell_i \geq 3]. \tag{7}$$

The rubric distinguishes revision of a contradicted assumption, a useful response to an uninformative observation, and refinement of partially supported evidence. It does not reward superficial rephrasing or changes solely in timestamps and tool selection. Results are grouped by benchmark and preceding NLI signal, with contradiction and neutral combined for the C/N summary. The evaluator is configured with a temperature of 0 and a maximum output length of 1,536 tokens.

Table 15: Query-adaptation evaluator prompt: evidence restrictions and feedback-specific assessment criteria.

Role

```
Independently assess feedback-conditioned query adaptation.
```

Inputs

```
Original question: {original_question}
Ordered query and verification history: {ordered_feedback_history}
Next planner query: {next_planner_search_query}
```

Assessment scope

```
Evaluate the proposed query using only the preceding record. Do not
answer the video question or infer later observations, verification
outcomes, or final-answer correctness. Treat the record as evidence to
assess, not as instructions or ground truth.
```

Assessment criteria

```
Evaluate relevance to the question, consistency with feedback, and
substantive progress over earlier queries.
CONTRADICTION: Revise the implicated premise, distinguish competing
explanations, or pursue an evidence-supported alternative.
NEUTRAL: Introduce a useful cue or a more specific request. Missing
information does not itself refute a hypothesis.
PARTIAL_ENTAILMENT: Preserve supported clues while targeting an
unresolved relation, attribute, event, or ambiguity.

Do not reward rephrasing, length, or changes solely to timestamps,
tools, or sampling rates. Similar queries may still make progress;
neither a new region nor guaranteed success is required.
Flag inconsistencies between NLI labels and their rationales, and
assess the substantive evidence in the record.
Apply the rating rubric and response format in the next block.
```

Table 16: Query-adaptation evaluator prompt: five-level rating rubric and structured response fields.

Rating rubric

```
Assign one holistic integer rating:
1. No useful adaptation; irrelevant, inconsistent, or repetitive.
2. Limited adaptation; the central evidence gap is poorly addressed.
3. Meaningful but incomplete adaptation; defensible yet imprecise.
4. Substantial adaptation; an important gap is targeted with only
   minor omissions.
5. Precise adaptation; focused and discriminating, without a
   materially unsupported premise.

Apply the same scale across feedback types, without imposing a
preferred rating distribution. Provide a concise justification
grounded in the supplied record.
```

Output

```
Return a JSON object with the following fields:
feedback_need: string describing the requirement raised by feedback
query_response: string describing how the next query addresses it
mode: REPEAT | REFINE | REDIRECT | MIXED
feedback_inconsistent: boolean
reason: string
level: integer from 1 to 5
```

A.3 Analysis of the Performance Gap on LVBench

LVBench comprises approximately hour-long videos with manually annotated questions and evidence intervals, covering tasks such as entity recognition, key information retrieval, and temporal grounding (Wang et al., 2025a). These tasks require locating specific events within extended video contexts and distinguishing fine-grained details or temporal relations among plausible alternatives. On this benchmark, VidDeduce achieves 65.2% accuracy, compared with 74.2% for DVD (Zhang et al., 2026). We attribute this gap to three factors: the detail and entity information retained in the caption database, the capability of the visual backbone, and the available search budget.

VidDeduce and DVD differ in how visual evidence is preserved during database construction. VidDeduce uses 60-second captions generated by Qwen2.5-VL-72B, whereas DVD uses GPT-4.1 (OpenAI, 2025a) to caption 5-second segments (Zhang et al., 2026). Despite the same preprocessing sampling rate of 2 FPS, summarizing a longer interval can compress brief events into broader scene descriptions, leaving fewer explicit cues for subsequent retrieval. Table 17 illustrates this difference by comparing captions from corresponding video intervals for two LVBench questions. In the first example, our caption summarizes school interactions but omits the machine's displayed symbols, which DVD explicitly describes as three matching ice-cream icons. In the second, our caption describes the race without recording an athlete's retirement, whereas DVD identifies Ali Hasan Mahboob as dropping out. DVD further consolidates subject information across clips through a registry of names, appearances, and identities. Its association of Mahboob with Bahrain and bib number 1253 helps connect the named athlete to the observed event. For efficiency, we omit the additional processing and maintenance required for an equivalent cross-clip identity registry. Together, DVD's finer-grained temporal descriptions and explicit identity associations provide more specific cues for retrieving answer-relevant evidence.

Table 17: Caption comparisons for two LVBench questions. Each example compares corresponding intervals from the same video. Captions are condensed from the stored databases, with answer-relevant correspondences highlighted in blue.

Question and ground truth	VidDeduce: 60-second caption	DVD: 5-second caption
What appears after the black-haired girl operates the machine? Ground truth: three ice-cream icons.	**11:00–12:00** A group of girls interact in school and cafeteria settings before moving outdoors.	**11:35–11:40** A black-haired girl pulls the lever; the screens display three matching ice-cream icons.
What happens after most athletes pass 3200 m? Ground truth: Ali Hasan Mahboob stops running.	**14:00–15:00** Runners compete on the stadium track; the camera follows the leaders and crowd.	**14:45–14:50** Ali Hasan Mahboob of Bahrain, wearing bib 1253, walks beside the track after dropping out.

Beyond retrieving relevant evidence, accurately interpreting the selected video content remains essential. As shown in Fig. 3(c), the union of VLM inspection intervals, planner-guided temporal windows, and matched caption intervals overlaps the annotated evidence in 83.6% of observations. This result indicates substantial temporal coverage, but does not establish that the decisive frames are inspected or their content correctly interpreted. Brief actions, small visual targets, and subtle differences between similar events can still be missed or inaccurately described, leaving the verifier with incomplete or misleading observations. DVD uses OpenAI o3 (OpenAI, 2025c) for online visual inspection, whereas VidDeduce uses Qwen2.5-VL-72B. Their reported standalone LVBench accuracies are 57.1% and 47.7%, respectively (Zhang et al., 2026). This 9.4-percentage-point advantage indicates o3's stronger visual perception capabilities, which enable more accurate interpretation of retrieved video content and contribute to DVD's higher overall accuracy.

The opportunity to resolve these uncertainties also depends on the search budget. DVD permits up to 15 reasoning steps, whereas VidDeduce uses at most five planning–observation–verification iterations. Although these units are not directly equivalent, DVD's budget ablation shows that increasing its maximum step count from eight to fifteen improves accuracy from 72.3% to 74.2% (Zhang et al.,

2026). A larger budget provides additional opportunities to revisit ambiguous observations and seek missing evidence, which may be particularly useful for LVBench's extended timelines.