EVIHARNESS: EVIDENCE-GROUNDED LEARNABLE HARNESS FOR LONG VIDEO UNDERSTANDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-video question answering must locate sparse, answer-critical evidence under a limited observation budget. Existing selective-acquisition methods allocate observations to find relevant intervals, but compressing them can discard details needed to answer; timestamp-based memory points to a location without identifying which unsupported claim needs further evidence. We address this gap by linking compact claims to executable source pointers and using verification diagnoses to guide targeted re-observation. We introduce EviHarness, a learnable harness with raw-addressable evidence records, a frozen verifier that diagnoses missing or conflicting support, and a budget-aware controller that searches, revisits sources, or answers. We train only the controller: supervised fine-tuning (SFT) learns from valid, correct, low-cost trajectories, and GRPO optimizes answer correctness and acquisition cost while observation and verification components remain fixed. With Qwen2.5-VL-7B under shared inference caps, SFT+RL improves over prompting on Video-MME, LVBench, and MLVU; on LVBench, independently audited source support is 77.1% versus 61.4% for timestamp-only memory.

1 INTRODUCTION

Long-video question answering turns extended recordings into information that can be queried without inspecting every frame. A question about an instructional video, a recorded activity, or a travel sequence may depend on a brief interaction or a single line of on-screen text. A useful system must locate that evidence across a long temporal span and preserve enough detail to justify its answer. Benchmarks such as Video-MME, LVBench, and MLVU test this combination of coverage and detail (Fu et al., 2024; Wang et al., 2024a; Zhou et al., 2024), alongside long-context and long-form reasoning evaluations (Wu et al., 2024; Mangalam et al., 2023). Because visual input and reasoning capacity are bounded, the central challenge is to allocate observation to the evidence that the question actually requires.

Existing approaches improve this allocation through selective acquisition. Uniform sampling offers predictable cost but can miss short events, while denser sampling consumes more visual input. Active Video Perception (AVP) uses planning and reflection (Wang et al., 2026b), and LensWalk controls temporal scope and sampling density (Li et al., 2026). Act2See, LongVideo-R1, and Omni-Agent learn acquisition policies (Ma et al., 2026; Qiu et al., 2026; Xing et al., 2026); VideoXAgent coordinates heterogeneous expert tools (Yang et al., 2026). These capabilities address where and how to observe. A remaining challenge lies in what happens after observation: a compressed description may discard information that a later answer needs, even when the relevant interval has already been located.

This gap follows from the role of compression. A summary can preserve the overall scene while omitting a place name, an object attribute, or the ordering of a short interaction. Further reasoning over the same summary cannot recover the missing pixels. A timestamp identifies where to return, but the recovery decision also benefits from knowing which claim is uncertain, what was observed, and how that observation was acquired. Evidence assessment must therefore connect a specific missing fact to a feasible new observation. Merely retaining more text does not provide that connection.

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.

Our solution maps an unsupported claim to a source observation that can be reacquired, and learns when to perform that recovery. Each claim retains an executable link to its source interval, together with the sampling settings and tool provenance that produced the observation. When support is incomplete or contradictory, a claim-level diagnosis identifies the information to acquire next. The controller can then inspect a denser temporal view, change the crop, or request optical character recognition. This separates the compact context used for reasoning from the source media available for recovery, allowing the agent to recover a needed detail without retaining every frame in its active context.

EviHarness implements this interface for offline long-video question answering. A controller chooses acquisition actions, an evidence builder creates raw-addressable records, and a frozen verifier checks candidate answers against stored observations. The verifier identifies missing or conflicting support; the controller combines that diagnosis with source pointers and the remaining budget to search, revisit, or answer. Every acquisition and verification call is charged to the same ledger, with a terminal reserve ensuring that the system can return an answer. Supervised fine-tuning (SFT) learns from valid, correct, low-cost trajectories, and reinforcement learning (RL) further optimizes answer correctness and acquisition cost. Only the controller is updated, keeping the observation tools and verifier fixed across learning stages.

Our contributions are:

- **Raw-addressable evidence memory.** We connect compact claims to source intervals and acquisition provenance, giving a diagnosed evidence gap an explicit route back to the media while bounding active context.
- **Learned, verification-guided recovery.** We formulate search, source revisits, and answer proposals within one budgeted acquisition policy. SFT and RL train the controller to use this interface with frozen observation and verification components.
- **Evidence across accuracy and source support.** Under shared Qwen2.5-VL-7B inference caps, SFT+RL improves over prompting on all three benchmarks. On LVBench, disabling revisits reduces accuracy by 2.9 points, while independent annotation finds a 15.7-point source-support gain over timestamp-only memory. These complementary tests link the overall gains to recoverable evidence and targeted acquisition.

2 Related work

Active perception and learned navigation. Selective observation can be guided by reflection and explicit sampling control, as in AVP and LensWalk (Wang et al., 2026b; Li et al., 2026), or learned through visual requests and temporal navigation, as in Act2See and LongVideo-R1 (Ma et al., 2026; Qiu et al., 2026). Query decomposition and event-based reflection provide complementary control strategies in VideoMind and VideoINSTA (Bhatnagar et al., 2026; Liao et al., 2024). OmniAgent combines multimodal access, textual memory, and policy optimization (Xing et al., 2026). Other approaches allocate observation capacity through curiosity-driven tree search (Yang et al., 2025), token selection (Zhang et al., 2025b), visual prompts (Doorenbos et al., 2026), or frame selection (Tan et al., 2026). EviHarness connects acquisition decisions to persistent claim-to-source records and verifier diagnostics, making recovery an explicit action choice.

Tool orchestration and memory. Video-language dialogue models such as Video-ChatGPT (Maaz et al., 2024) provide a foundation for interactive analysis. Agent systems extend this interface with temporal and object memory (Fan et al., 2024), LLM planning over visual tools (Wang et al., 2024b), and coordinated expert tools with budget control (Yang et al., 2026). Memory organization ranges from hierarchical indexing and entity cohesion in HAVEN (Yin et al., 2026) to reasoning-time hierarchies in VideoARM (Yin et al., 2025). Constant-token streaming memory (Qian et al., 2024) and hour-scale training and inference (Lin et al., 2025) address longer inputs. Local-to-global aggregation (Pang & Wang, 2025), multi-granular tool search (Zhang et al., 2025a), coordinated grounding and vision agents (Liu et al., 2026a), query-adaptive trees (Wang et al., 2025b), and iterative document retrieval (Ma et al., 2025) offer further ways to organize access. EviHarness stores compact claims together with executable source pointers and acquisition provenance. This representation makes each diagnostic actionable: the controller can resolve the affected record and request a new observation of its source.

Verification and source recovery. Localized hypothesis testing in Think, Then Verify (Wang et al., 2026a) and the provenance, coverage, and conflict ledger in VESTA (Zhang et al., 2026a) connect reasoning to evidence assessment. EviHarness couples three interfaces: memory preserves the claim-to-source link, verification identifies the missing or conflicting claim, and a learned controller chooses a budgeted acquisition in response. The verifier reads stored observations and emits a diagnosis; it neither inspects raw media nor executes tools. This operates on retained video with offline random access, whereas StreamArena studies continuous interactive video understanding (Zhang et al., 2026b). Timestamp-only memory, disabled-revisit, and disabled-verifier controls test the roles of persistent records and recovery in our setting.

3 METHOD

3.1 BUDGET-CONSTRAINED EVIDENCE ACQUISITION

Given a video V of duration D, a question q, and an initial budget b_1, the system predicts an answer $\hat{y}$ and supporting evidence identifiers $\mathcal{I}$; y denotes the reference answer used only for training and evaluation. We assume offline random access to retained media. Figure 1 summarizes the acquisition loop. The controller observes

$$s_t = (q, D, \widetilde{M}_t, H_t, b_t), \qquad a_t \sim \pi_\theta(\cdot \mid s_t), \tag{1}$$

where $\widetilde{M}_t$ contains the visible evidence records and H_t records prior actions, execution status, costs, and verifier feedback. Unobserved video content is accessible only through acquisition actions.

An acquisition action is $a_t = (k_t, u_t, v_t, \rho_t, \eta_t, i_t^{\text{ref}})$, with tool k_t, interval $[u_t, v_t] \subseteq [0, D]$, sampling specification ρ_t, tool-specific settings η_t, and an optional reference record i_t^{ref}. When present, the executor resolves that record's pointer and uses its source span, tool history, and acquisition settings as defaults; explicit action fields override those defaults. A null reference requests ordinary search. The executor returns $o_t = \mathcal{E}(V, a_t)$. An evidence builder converts this observation into one or more records, and the archive is updated as $M_{t+1} = M_t \cup \mathcal{B}(o_t, a_t)$. An answer proposal instead has the form $\texttt{ANSWER}(\hat{y}, \mathcal{I})$ and is passed to verification before normal termination.

We use the budget vector $b_t = (b_t^f, b_t^s, b_t^u)$ for cumulative frame presentations, controller steps, and calibrated tool-compute units. One step includes controller generation and all operations triggered by its action, including observation, memory construction, or answer verification. Its actual cost is $c_t = (F_t, 1, U_t)$. A fixed reserve $r = (0, 1, U_{\text{term}})$ covers a text-only terminal answer and verification. Nonterminal steps obey

$$b_{t+1} = b_t - c_t, \qquad \mathcal{A}_{\text{cont}}(b_t) = \{a : \bar{c}_t(a) + r \preceq b_t\}. \tag{2}$$

Only schema-valid actions may execute. The bound $\bar{c}_t(a)$ includes controller generation and all calls triggered by the action; answer proposals also include verification. Caps and timeouts enforce $0 \preceq c_t \preceq \bar{c}_t(a)$, including failed calls. The initial budget must cover r; the reserved terminal step permits no further acquisition. Appendix B.1 specifies accounting and execution details. Calibrated compute units are distinct from wall-clock latency.

3.2 TEMPORAL NAVIGATION AND HETEROGENEOUS TOOLS

Acquisition begins with $\texttt{SCOUT}(0, D, N_s)$ when the budget permits. The controller then chooses `ZOOM` or `CLIP` for local temporal inspection, optical character recognition (`OCR`) for visible text, or `ANSWER` for a cited proposal. The general interface also supports automatic speech recognition (`ASR`). Tool outputs include timestamps, uncertainty, and provenance.

The executor validates temporal bounds, tools, sample counts, and budget feasibility. Invalid actions consume a controller step and return a structured error without media access. Every frame presented to a model counts toward the frame budget, including repeated views and OCR inputs; caching decoded frames does not waive this charge.

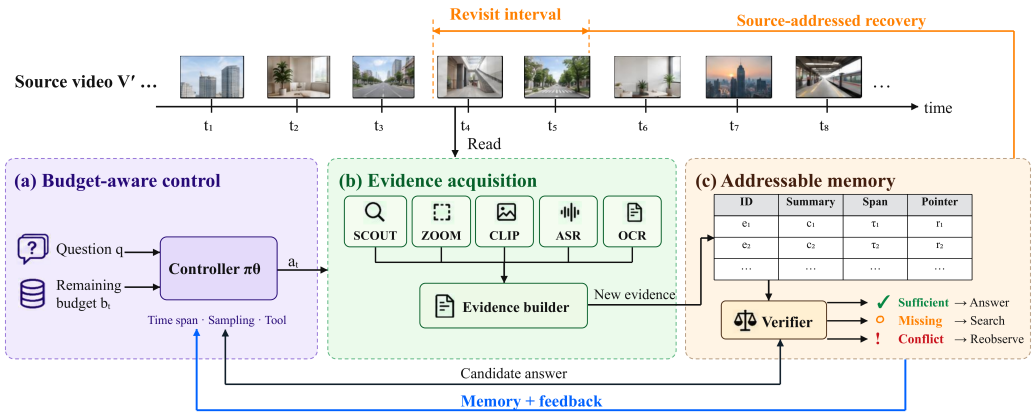


Figure 1: EviHarness architecture. Evidence pointers locate source media outside the language context. Missing or conflicting support can trigger another acquisition through the controller. Source revisits and verifier calls are charged to the inference budget. The diagram illustrates the normal verification loop; budget-exhausted termination is specified in Section 3.4. The frame thumbnails are schematic and are not benchmark results.

3.3 Raw-addressable evidence memory

Raw-addressable memory preserves the information needed to reacquire an observation while keeping its description compact. Each record is

$$e_i = (i, c_i, \tau_i, m_i, k_i, r_i, z_i), \tag{3}$$

where i is a stable identifier, c_i a concise claim or summary, τ_i its temporal span, m_i its modality, k_i its source tool, r_i its raw pointer, and z_i structured attributes. Attributes include entities, observation uncertainty, parent records, and validation status. A raw pointer stores a video identifier or content hash, source interval, exact sampled timestamps when applicable, decoding and sampling settings, crop coordinates when used, and tool version. The pointer resolves retained media outside the language-model context, preserving a compact context while keeping the source addressable.

The archive preserves immutable records, observations, and pointers; its context view exposes at most K records under a separate text-token cap L_M. Referenced records and unresolved conflicts receive priority, followed by query relevance and recency, with stable ID tie-breaking. The executor resolves stored IDs without treating retrieval as a new source observation. Omissions from the controller view are recorded; verification constructs its own view from the archive, so controller-view omission alone does not imply verification-view overflow. Appendix B.1 defines a deterministic reference selector; the reported configuration uses $K = 12$, and L_M specifies the serialization budget.

A lightweight graph stores temporal adjacency and claim-scoped support/conflict relations. The evidence builder proposes support links; the verifier records conflicts with explicit rationales. Relations retain their producer and version, and status updates preserve the immutable archive. The controller reads these serialized relations directly; Appendix B.1 specifies their construction.

Re-observation appends a record instead of replacing its predecessor. This preserves the distinction between the original observation, the diagnostic that motivated recovery, and the new source evidence. A denser sample or a different crop can therefore resolve a specific uncertainty while retaining the provenance of both observations. The controller need not keep every record in its active context: stable identifiers allow the executor to retrieve the cited records from the archive when constructing the verification view.

Raw addressing preserves both recoverability and acquisition provenance. The verifier assesses the stored description of the recovered observation; independent human auditing assesses support in the cited source media. A timestamp-only baseline in Section 4.4 retains genuine source access, so its comparison with the full system isolates persistent claim-to-source records from temporal navigation itself.

3.4 Verification and evidence-grounded backtracking

Given a candidate answer $\hat{y}$, cited records $M_{\mathcal{I}}$, relevant unresolved records M_{unres}, and a verification-view overflow flag o_t^{view}, the frozen verifier returns

$$(v_t, J_t, d_t) = \mathcal{V}_\phi(q, \hat{y}, M_{\mathcal{I}}, M_{\text{unres}}, o_t^{\text{view}}), \tag{4}$$

where v_t is the verification status, J_t identifies affected evidence, and d_t describes the missing or disputed claim and the required observation. The verifier applies three discrete judgments in order: *conflict* for an answer-critical contradiction, *insufficient* for missing support without a contradiction, and *sufficient* when valid observations support every answer-critical claim. Invalid citations, omitted unresolved records, unavailable sources, and verification-view overflow preclude a sufficient judgment. The fixed prompt uses no numerical confidence threshold. A sufficient judgment denotes verifier acceptance based on stored observations, not an independent check of the raw media or a guarantee of answer correctness. Benchmark evaluation measures correctness; human auditing separately measures source support.

For *sufficient* support, the system returns the answer and its evidence IDs. For *insufficient* support, the controller targets a missing fact or an uncovered interval; it may revisit a source when the missing detail is localized, or search elsewhere when it is not. For *conflict*, it resolves a cited pointer and reacquires the source, potentially increasing temporal density, changing the crop, or switching between visual inspection, OCR, and ASR. Re-observation creates a new node linked to its predecessor, after which the candidate is reassessed. `BACKTRACK` is implemented by ordinary acquisition actions, and every recovery action is charged in the same ledger.

The verifier reads stored observations only. The controller selects the next tool and interval from (J_t, d_t, b_t) and source pointers, then acquires any additional media through the budgeted executor. Appendix G provides prompts, tool-selection guidance, and the JSON interface; Figure 5(b) illustrates the recovery cycle.

When another complete nonterminal step cannot leave r intact, the executor enters the reserved terminal step immediately. This step counts within B_s and consumes the terminal reserve. The controller answers from stored evidence and the verifier runs once; rejection returns an *unverified* answer without another loop. Verifier acceptance requires at least one cited observation. On generation failure, the executor retains the latest valid candidate with a failure status, or returns an error if none exists. Multiple-choice evaluation scores a valid forced answer normally and a missing answer as incorrect; verification status remains a separate diagnostic.

3.5 Learning the harness policy

We evaluate three controller variants: prompting, supervised fine-tuning (SFT), and SFT followed by reinforcement learning (SFT+RL). The prompt controller uses the frozen base model. SFT trains an independent controller LoRA adapter, and SFT+RL continues updating that adapter from an SFT checkpoint. The observer, evidence builder, verifier, visual encoder, and tools remain frozen throughout. Within each backbone, all variants use the same observation and verification components, tool interface, and inference caps. Appendix E details trajectory generation and optimization.

Cost-aware trajectory selection. For each training example, a frozen prompt-based EviHarness teacher with the same base model as the student samples $N = 4$ trajectories $\xi = (s_1, a_1, \ldots, s_T, a_T, \hat{y}, \mathcal{I})$. We retain trajectories that answer correctly, satisfy the action schema and budget, and cite valid records. Verifier sufficiency is not a hard selection criterion. If no trajectory qualifies, four more are sampled; the example is discarded if all eight fail. Teacher actions are sampled without access to reference answers; references are used only for post-hoc filtering. Student states contain only information available under the inference protocol. Let $\mathcal{T}_{\text{valid}}$ denote the

retained set. A demonstration is selected by

$$\xi^* = \arg\min_{\xi \in \mathcal{T}_{\text{valid}}} \left[0.5 \frac{F(\xi)}{B_f} + 0.3 \frac{S(\xi)}{B_s} + 0.2 \frac{U(\xi)}{B_u} \right]. \tag{5}$$

Here F, S, and U are cumulative frame presentations, decision steps, and calibrated compute. The weights are selected once on development data and fixed across benchmarks; trajectory index breaks equal-cost ties. Citation validity requires resolvable records, not verifier-confirmed support.

Supervised fine-tuning. The controller learns structured action outputs, including temporal parameters and cited identifiers, with

$$\mathcal{L}_{\text{SFT}}(\theta) = -\sum_{\xi^*} \sum_{t} \log \pi_\theta(a_t^* \mid s_t). \tag{6}$$

A controller LoRA adapter (Hu et al., 2021) learns structured actions and terminal answers. Observation, memory, and verifier-feedback tokens provide conditioning and receive zero loss weight; free-form chains of thought are not supervised. Selected trajectories are expanded into step-level state–action examples. Each state retains the preceding action history and serialized evidence memory, so supervision teaches the controller to choose an action in the context created by earlier acquisitions. The trajectory objective thus covers both ordinary search and diagnostic-guided recovery without updating the components that produce observations or verification feedback.

Policy optimization. Following supervised initialization, we optimize the controller with GRPO (Shao et al., 2024) using online rollouts from the training split and the trajectory-level reward

$$R(\xi) = G(\xi) \left(\mathbb{I}[\hat{y} = y] - 0.10 \frac{F(\xi)}{B_f} - 0.05 \frac{S(\xi)}{B_s} - 0.05 \frac{U(\xi)}{B_u} \right) - (1 - G(\xi))\kappa. \tag{7}$$

The terminal validity gate G is one for a schema-valid terminal answer and zero otherwise. Invalid terminal outputs receive reward $-\kappa$, where $\kappa > 0$ is the invalid-terminal penalty magnitude; valid terminal outputs receive the correctness-minus-cost reward above. Intermediate invalid actions are charged and rejected but do not force $G = 0$ if a valid answer follows. SFT filtering, by contrast, excludes trajectories containing any schema-invalid or budget-invalid action. For valid terminal outputs, training optimizes answer correctness and acquisition cost, with no verifier-sufficiency gate or evidence reward. Human evidence auditing is independent of policy learning.

4 Experiments

We evaluate controller learning on three long-video benchmarks under fixed inference caps. LVBench ablations isolate the contributions of memory and recovery, and an independent audit measures whether the resulting answers are supported by their cited sources.

4.1 Benchmarks and evaluation settings

We report Video-MME Overall accuracy without subtitles (Fu et al., 2024), LVBench Overall (Wang et al., 2024a), and MLVU M-Avg (Zhou et al., 2024), following official task definitions and aggregation rules. EviHarness uses decoded RGB frames and OCR throughout trajectory generation, training, and evaluation; official subtitles and ASR are disabled. All variants share the answer extractor and evaluator revision.

Policy learning uses LongVideoBench validation (Wu et al., 2024), partitioned by video into training and development sets. Videos matching any evaluation benchmark are excluded from both partitions; test benchmarks are not used for policy training or checkpoint selection. Appendix E describes the split and overlap checks.

4.2 Implementation settings

The primary backbone is Qwen2.5-VL-7B-Instruct (Bai et al., 2025); an independently trained Qwen3-VL-8B-Instruct (Qwen Team, 2025) evaluates the same recipe across backbones. OCR uses

Table 1: Main results on three long-video benchmarks. Video-MME reports overall accuracy without subtitles; LV denotes LVBench Overall and M denotes MLVU M-Avg. Published baselines retain their original models and protocols. The four EviHarness variants below the separator share the stated inference caps. Dashes indicate metrics absent from the cited source.

Method	Venue	LLM version	Video-MME ↑	LV ↑	M ↑
Qwen2.5-VL-7B (Bai et al., 2025)	Tech. report (2025)	Qwen2.5-VL-7B-Instruct	65.1	45.3	70.2
Qwen2.5-VL-72B (Bai et al., 2025)	Tech. report (2025)	Qwen2.5-VL-72B-Instruct	73.3	47.3	74.6
AdaReTaKe (Wang et al., 2025a)	ACL Findings (2025)	Qwen2.5-VL-72B	73.5	53.3	–
MR. Video (Pang & Wang, 2025)	NeurIPS (2025)	GPT-4o + Gemini-2.0-Flash	–	60.8	–
VideoTree (Wang et al., 2025b)	CVPR (2025)	GPT-4	–	28.8	–
VideoMind (Liu et al., 2026b)	ICLR (2026)	Qwen2-VL-7B-Instruct	–	40.8	–
Gemini-1.5-Pro (Fu et al., 2024)	–	Gemini-1.5-Pro	75.0	–	–
GPT-4o (Fu et al., 2024)	–	GPT-4o	71.9	–	–
LLaVA-Video-72B (Fu et al., 2024)	–	LLaVA-Video-72B-Qwen2	70.6	–	–
EviHarness (prompt)	Ours	Qwen2.5-VL-7B-Instruct	72.6	62.2	73.0
EviHarness (SFT)	Ours	Qwen2.5-VL-7B-Instruct	73.6	63.4	73.3
EviHarness (SFT+RL)	Ours	Qwen2.5-VL-7B-Instruct	74.9	63.9	74.8
EviHarness (SFT+RL)	Ours	Qwen3-VL-8B-Instruct	75.3	65.9	75.7

the active vision-language model. Each run allows 64 cumulative frame presentations and six controller steps, including SCOUT and final ANSWER. Scout and zoom calls use eight frames; CLIP uses at most 16 frames over 12 seconds. Active memory holds up to 12 records within 4,096 text tokens, and controller temperature is 0.2. Appendices D and E provide full settings.

Controller training and checkpoint selection. Training uses four NVIDIA A800 80 GB GPUs in BF16 and updates only the controller LoRA adapter. SFT runs for three epochs; GRPO continues from the selected SFT checkpoint for 600 optimizer steps. Each stage selects the checkpoint with the highest accuracy on the retained development partition. Human source-support annotations are used only for evaluation. The Qwen3 experiment independently repeats trajectory generation and both training stages. Appendix E provides adapter, optimizer, input-limit, and checkpoint-evaluation settings.

4.3 Main comparison with public baselines

Table 1 compares controller learning under shared Qwen2.5-VL-7B settings and includes the independently trained Qwen3 variant. Published baselines retain their original models and protocols and provide external reference points.

Learning improves all three benchmarks. SFT+RL improves over the Qwen2.5-VL-7B prompt controller by 2.3, 1.7, and 1.8 percentage points on Video-MME, LVBench, and MLVU. Indepen-

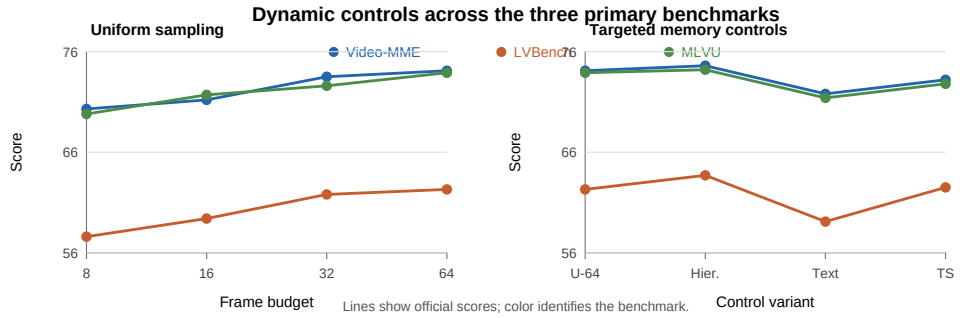


Figure 2: Acquisition and memory controls across Video-MME, LVBench, and MLVU under the shared visual-only protocol. The left panel varies uniform-sampling density; the right compares Uniform-64, hierarchical search, text-only memory, and timestamp-only memory.

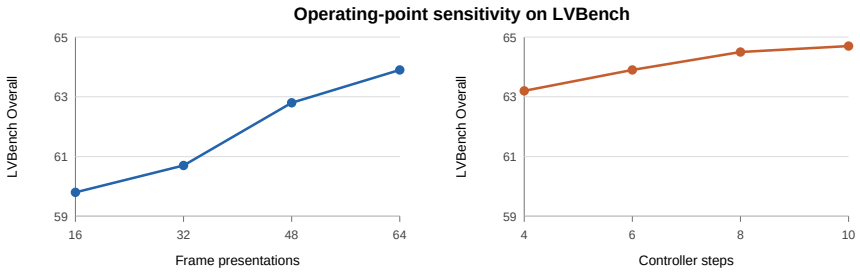


Figure 3: Inference-budget sensitivity on LVBench under the shared Qwen2.5-VL-7B SFT+RL configuration. The left panel varies the frame-presentation cap and the right panel varies controller steps.

dently training the recipe with Qwen3-VL-8B adds 0.4, 2.0, and 0.9 points, respectively. Learning improves accuracy across all three benchmarks under the same inference caps.

Figure 2 separates acquisition structure from memory representation. Hierarchical search exceeds Uniform-64 on all three benchmarks, and timestamp-only memory exceeds text-only memory. Full EviHarness further improves over hierarchical search by 0.3, 0.2, and 0.6 points on Video-MME, LVBench, and MLVU. The following ablations isolate the recovery components behind these comparisons.

4.4 Mechanism tests and ablations

The LVBench ablation starts from the Qwen2.5-VL-7B SFT+RL system at 63.9 accuracy. Replacing full records with timestamp-only memory yields 62.2, disabling source revisits yields 61.0, and removing verification yields 62.2. Disabling revisits produces the largest decrease, at 2.9 points. Cost-matched extra observation reaches 61.9, placing targeted recovery 2.0 points higher and supporting the value of directing observation toward a diagnosed evidence gap.

4.5 Inference-budget sensitivity

Figure 3 evaluates sensitivity to inference caps on LVBench. Accuracy rises from 59.8 to 63.9 as the frame-presentation cap increases from 16 to 64. The controller-step sweep rises from 63.2 to 64.7 between four and ten steps; the reported six-step configuration reaches 63.9.

4.6 Evidence support and qualitative recovery

We audit the same $n = 166$ LVBench questions for EviHarness and timestamp-only memory. Two annotators independently assess each system answer using the question, proposed answer, and cited

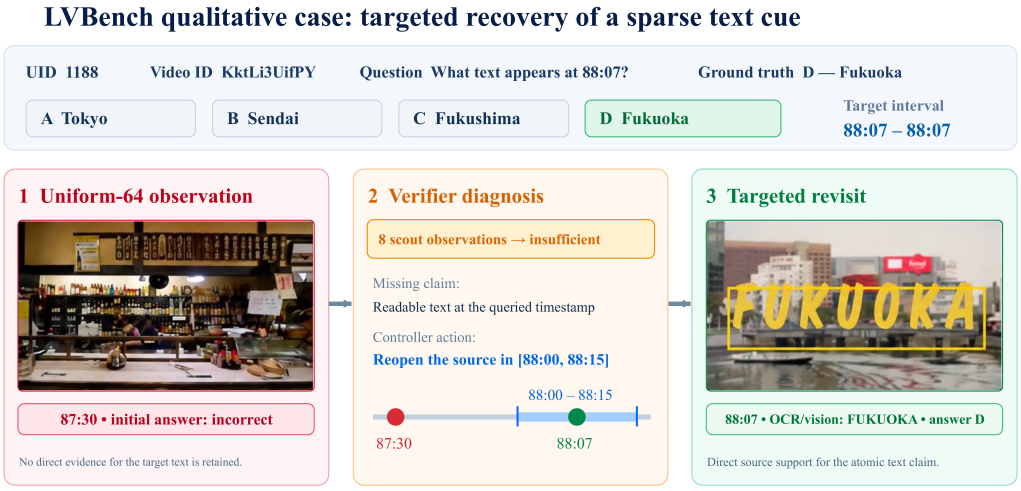


Figure 4: Targeted evidence recovery on LVBench (UID 1188). Uniform-64 observes a non-diagnostic frame at 87:30 and returns an incorrect answer. The verifier marks eight scout observations as insufficient, so the controller reopens 88:00–88:15; the 88:07 source frame reveals "FUKUOKA" and supports answer D. The observation panels show source-video frames with explanatory annotations.

source intervals or frames; a third annotator adjudicates disagreements. Evidence support is judged without the reference answer. Let s_i indicate source support for $\hat{y}_i$. The primary evidence success rate is unconditional,

$$\mathrm{ESR}_{\mathrm{all}} = \frac{1}{n}\sum_{i=1}^{n}\mathbb{I}[s_i], \qquad \mathrm{GAcc} = \frac{1}{n}\sum_{i=1}^{n}\mathbb{I}[\hat{y}_i = y_i \wedge s_i]. \tag{8}$$

EviHarness achieves 77.1% $\mathrm{ESR}_{\mathrm{all}}$ versus 61.4% for timestamp-only memory, a gain of 15.7 percentage points. Grounded accuracy, requiring both correctness and support, increases from 48.8% to 52.4%. Pre-adjudication agreement is Cohen's $\kappa = 0.80$; verifier precision and recall against adjudicated human labels are both 79.7%. Appendix F provides the counts. Unsupported forced answers receive no evidence credit. These paired metrics distinguish source support from jointly correct and supported answers.

Qualitative recovery case. Figure 4 shows how diagnosing missing text support leads to a targeted revisit and a source-supported answer. The recovered frame supplies the detail absent from the initial observation, linking verification to acquisition rather than further reasoning over an incomplete summary. The diagnostic identifies the missing text, and the controller reopens the localized interval to acquire it. The executable pointer connects this claim-level gap to a source observation that directly addresses the question.

5 Conclusion

EviHarness connects an unsupported claim to a source observation that can be reacquired and learns when to perform that recovery. Raw-addressable records preserve this connection, the verifier diagnoses missing or conflicting support, and the controller chooses the next budgeted acquisition. Controller learning improves accuracy on three benchmarks, while LVBench ablations and independent annotation support the contributions of source revisits and persistent evidence records. Recoverable memory provides a practical connection between selective observation and source-supported answering.

AI USE STATEMENT

OpenAI Codex was used to polish the manuscript.

ETHICS STATEMENT

Long-video analysis may process identifiable people, speech, and activities. Raw-addressable memory therefore calls for access controls and retention policies appropriate to the underlying media. This work evaluates question answering on existing video benchmarks. Use and redistribution of videos, frames, and model artifacts are governed by their respective licenses. Evidence annotation should protect annotator privacy and follow applicable consent and review requirements.

REPRODUCIBILITY STATEMENT

Section 3 defines the action space, evidence representation, verification rules, and learning objectives. Section 4 reports the evaluation configuration and results. Appendices B and C specify execution semantics, accounting rules, and role contracts; Appendix D summarizes the reported inference settings, and Appendix E documents policy training. Appendix G provides the verifier prompts and interface.

REFERENCES

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Junyang Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL Technical Report. Technical Report arXiv:2502.13923, arXiv, 2025. URL https://arxiv.org/abs/2502.13923.

Shubhang Bhatnagar, Renxiong Wang, Kapil Krishnakumar, Adel Ahmadyan, Zhaojiang Lin, Lambert Mathias, Xin Luna Dong, Babak Damavandi, Narendra Ahuja, and Seungwhan Moon. VideoMind: Thinking in Steps for Long Video Understanding. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 5: Industry Track)*, pp. 406–416, 2026. URL https://aclanthology.org/2026.eacl-industry.30/.

Lars Doorenbos, Federico Spurio, and Juergen Gall. Video Panels for Long Video Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 31293–31303, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Doorenbos_Video_Panels_for_Long_Video_Understanding_CVPR_2026_paper.html.

Yue Fan, Xiaojian Ma, Rujie Wu, Yuntao Du, Jiaqi Li, Zhi Gao, and Qing Li. VideoAgent: A Memory-augmented Multimodal Agent for Video Understanding. *arXiv preprint arXiv:2403.11481*, 2024. URL https://arxiv.org/abs/2403.11481.

Xinyu Fang, Kangrui Mao, Haodong Duan, Xiangyu Zhao, Yining Li, Dahua Lin, and Kai Chen. MMBench-Video: A Long-Form Multi-Shot Benchmark for Holistic Video Understanding. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/a2326c9715a516c91174132e0170073a-Abstract-Datasets_and_Benchmarks_Track.html.

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

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models. *arXiv preprint arXiv:2106.09685*, 2021. URL https://arxiv.org/abs/2106.09685.

Keliang Li, Yansong Li, Hongze Shen, Mengdi Liu, Hong Chang, and Shiguang Shan. LensWalk: Agentic Video Understanding by Planning How You See in Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/papers/Li_LensWalk_Agentic_Video_Understanding_by_Planning_How_You_See_in_CVPR_2026_paper.pdf.

Ruotong Liao, Max Erler, Huiyu Wang, Guangyao Zhai, Gengyuan Zhang, Yunpu Ma, and Volker Tresp. VideoINSTA: Zero-shot Long Video Understanding via Informative Spatial-Temporal Reasoning with LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 6577–6602, 2024. URL https://aclanthology.org/2024.findings-emnlp.384/.

Jingyang Lin, Jialian Wu, Ximeng Sun, Ze Wang, Jiang Liu, Yusheng Su, Xiaodong Yu, Hao Chen, Jiebo Luo, Zicheng Liu, and Emad Barsoum. Unleashing Hour-Scale Video Training for Long Video-Language Understanding. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/19741c617c87a25627682e5714af1501-Abstract-Conference.html.

Runtao Liu, Ziyi Liu, Jiaqi Tang, Yue Ma, Renjie Pi, Jipeng Zhang, and Qifeng Chen. LongVideoAgent: Multi-Agent Reasoning with Long Videos. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics*, pp. 40404–40416, 2026a. URL https://aclanthology.org/2026.acl-long.1876/.

Ye Liu, Kevin Qinghong Lin, Chang Wen Chen, and Mike Zheng Shou. VideoMind: A Chain-of-LoRA Agent for Temporal-Grounded Video Reasoning. In *International Conference on Learning Representations*, 2026b. URL https://videomind.github.io/.

Martin Q. Ma, Yuxiao Qu, Aditya Agrawal, Willis Guo, Paul Pu Liang, Ruslan Salakhutdinov, and Louis-Philippe Morency. Act2See: Emergent Active Visual Perception for Video Reasoning. *arXiv preprint arXiv:2605.01657*, 2026. URL https://arxiv.org/abs/2605.01657.

Ziyu Ma, Chenhui Gou, Hengcan Shi, Bin Sun, Shutao Li, Hamid Rezatofighi, and Jianfei Cai. DrVideo: Document Retrieval Based Long Video Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18936–18946, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Ma_DrVideo_Document_Retrieval_Based_Long_Video_Understanding_CVPR_2025_paper.html.

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Khan. Video-ChatGPT: Towards Detailed Video Understanding via Large Vision and Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12585–12602, 2024. URL https://aclanthology.org/2024.acl-long.679/.

Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. EgoSchema: A Diagnostic Benchmark for Very Long-form Video Language Understanding. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/90ce332aff156b910b002ce4e6880dec-Abstract-Datasets_and_Benchmarks.html.

Ziqi Pang and Yu-Xiong Wang. MR. Video: MapReduce as an Effective Principle for Long Video Understanding. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://papers.neurips.cc/paper_files/paper/2025/hash/0a02c2bc2e2148b803c4ade1d71e1d25-Abstract-Conference.html.

Rui Qian, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Shuangrui Ding, Dahua Lin, and Jiaqi Wang. Streaming Long Video Understanding with Large Language Models. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/d7ce06e9293c3d8e6cb3f80b4157f875-Abstract-Conference.html.

Jihao Qiu, Lingxi Xie, Xinyue Huo, Qi Tian, and Qixiang Ye. LongVideo-R1: Smart Navigation for Low-cost Long Video Understanding. *arXiv preprint arXiv:2602.20913*, 2026. URL https://arxiv.org/abs/2602.20913.

Qwen Team. Qwen3-VL-4B-Instruct model card, 2025. URL https://huggingface.co/Qwen/Qwen3-VL-4B-Instruct. Accessed September 19, 2026.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. *arXiv preprint arXiv:2402.03300*, 2024. URL https://arxiv.org/abs/2402.03300.

SYSTRAN. Faster whisper, 2026. URL https://github.com/SYSTRAN/faster-whisper. Software repository, accessed September 19, 2026.

Wenhui Tan, Xiaoyi Yu, Jiaze Li, Yijing Chen, Jianzhong Ju, Zhenbo Luo, Ruihua Song, and Jian Luan. MSJoE: Jointly Evolving MLLM and Sampler for Efficient Long-Form Video Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19487–19496, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Tan_MSJoE_Jointly_Evolving_MLLM_and_Sampler_for_Efficient_Long-Form_Video_CVPR_2026_paper.html.

Weihan Wang, Zehai He, Wenyi Hong, Yean Cheng, Xiaohan Zhang, Ji Qi, Xiaotao Gu, Shiyu Huang, Bin Xu, Yuxiao Dong, Ming Ding, and Jie Tang. LVBench: An Extreme Long Video Understanding Benchmark. *arXiv preprint arXiv:2406.08035*, 2024a. URL https://arxiv.org/abs/2406.08035.

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, 2025a. URL https://aclanthology.org/2025.findings-acl.283/.

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, 2024b. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/papers/10325.pdf.

Zheng Wang, Haoran Chen, Haoxuan Qin, Zhipeng Wei, Tianwen Qian, and Cong Bai. Think, Then Verify: A Hypothesis-Verification Multi-Agent Framework for Long Video Understanding. *arXiv preprint arXiv:2603.04977*, 2026a. URL https://arxiv.org/abs/2603.04977.

Ziyang Wang, Shoubin Yu, Elias Stengel-Eskin, Jaehong Yoon, Feng Cheng, Gedas Bertasius, and Mohit Bansal. VideoTree: Adaptive Tree-based Video Representation for LLM Reasoning on Long Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3272–3281, 2025b. URL https://openaccess.thecvf.com/content/CVPR2025/html/Wang_VideoTree_Adaptive_Tree-based_Video_Representation_for_LLM_Reasoning_on_Long_CVPR_2025_paper.html.

Ziyang Wang, Honglu Zhou, Shijie Wang, Junnan Li, Caiming Xiong, Silvio Savarese, Mohit Bansal, Michael S. Ryoo, and Juan Carlos Niebles. Active Video Perception: Iterative Evidence Seeking for Agentic Long Video Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pp. 9088–9099, 2026b. URL https://openaccess.thecvf.com/content/CVPR2026F/html/Wang_Active_Video_Perception_Iterative_Evidence_Seeking_for_Agentic_Long_Video_CVPRF_2026_paper.html.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. LongVideoBench: A Benchmark for Long-context Interleaved Video-Language Understanding. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/329ad516cf7a6ac306f29882e9c77558-Abstract-Datasets_and_Benchmarks_Track.html.

Zhenghao Xing, Ruiyang Xu, Yuxuan Wang, Jinzheng He, Ziyang Ma, Qize Yang, Yunfei Chu, Jin Xu, Junyang Lin, Chi-Wing Fu, and Pheng-Ann Heng. Native Active Perception as Reasoning for Omni-Modal Understanding. *arXiv preprint arXiv:2606.19341*, 2026. URL https://arxiv.org/abs/2606.19341.

Sen Yang, Boqiang Duan, Jing Yang, Weihao Bo, Jie Liu, Boyuan Tong, Ze Feng, Wenkang Zhang, Jingdong Wang, and Hua Wu. Online Video Agent Harness for Long Video Understanding. *arXiv preprint arXiv:2609.12818*, 2026. URL https://arxiv.org/abs/2609.12818.

Zeyuan Yang, Delin Chen, Xueyang Yu, Maohao Shen, and Chuang Gan. VCA: Video Curious Agent for Long Video Understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20168–20179, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Yang_VCA_Video_Curious_Agent_for_Long_Video_Understanding_ICCV_2025_paper.html.

Xinlei Yin, Xiulian Peng, Xiao Li, Zhiwei Xiong, and Yan Lu. Hierarchical Long Video Understanding with Audiovisual Entity Cohesion and Agentic Search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 32882–32891, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Yin_Hierarchical_Long_Video_Understanding_with_Audiovisual_Entity_Cohesion_and_Agentic_CVPR_2026_paper.html.

Yufei Yin, Qianke Meng, Minghao Chen, Jiajun Ding, Zhenwei Shao, and Zhou Yu. VideoARM: Agentic Reasoning over Hierarchical Memory for Long-Form Video Understanding. *arXiv preprint arXiv:2512.12360*, 2025. URL https://arxiv.org/abs/2512.12360.

Can Zhang, Baofeng Zhang, Xiaotian Han, Junyuan Shang, Yuchen Ding, Shuohuan Wang, Dianhai Yu, and Ruirui Li. From Intent to Evidence: Policy-Steered Multi-Strategy Retrieval for Long-Video Agents. *arXiv preprint arXiv:2608.31005*, 2026a. URL https://arxiv.org/abs/2608.31005.

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. In *Advances in Neural Information Processing Systems*, volume 38, 2025a. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/8190b210e9808e54ee16263b673a847d-Abstract-Conference.html.

Xichen Zhang, Guankai Li, Yinghao Zhu, Shijian Wang, Sitong Wu, Shaozuo Yu, Meng Chu, Yuan Lu, and Jiaya Jia. StreamArena: Toward Continuous, Interactive, and Long-Horizon Agentic Streaming Video Understanding. *arXiv preprint arXiv:2608.05703*, 2026b. URL https://arxiv.org/abs/2608.05703.

Zheyu Zhang, Ziqi Pang, Shixing Chen, Xiang Hao, Vimal Bhat, and Yu-Xiong Wang. One Token per Highly Selective Frame: Towards Extreme Compression for Long Video Understanding. In *Advances in Neural Information Processing Systems*, volume 38, 2025b. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/52eb0f34026703c6a1c27693c2ab60dc-Abstract-Conference.html.

Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Zhengyang Liang, Shitao Xiao, Minghao Qin, Xi Yang, Yongping Xiong, Bo Zhang, Tiejun Huang, and Zheng Liu. MLVU: Benchmarking Multi-task Long Video Understanding. *arXiv preprint arXiv:2406.04264*, 2024. URL https://arxiv.org/abs/2406.04264.

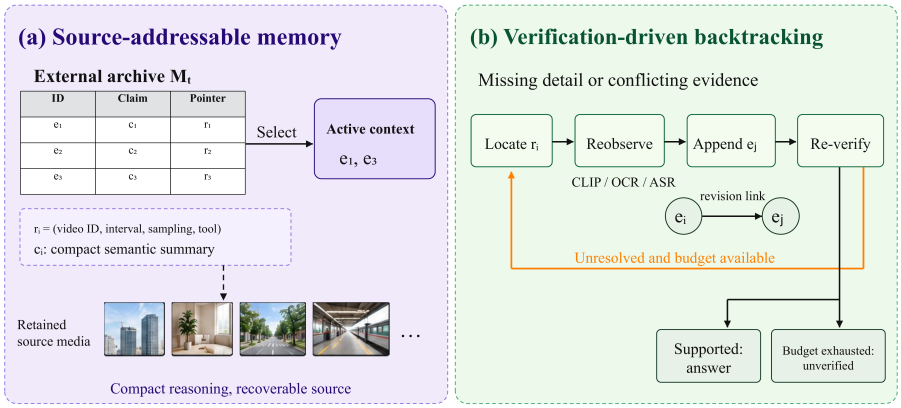


Figure 5: Raw-addressable memory and verification-driven recovery. (a) Compact context retains an explicit route to source media through evidence records. (b) A missing or conflicting detail can trigger targeted re-observation, producing a new record linked to its predecessor. Re-verification follows the shared budget ledger.

A Additional conceptual figure

Figure 5 provides the detailed schematic for raw-addressable memory and verifier-triggered recovery referenced in Sections 3.3 and 3.4.

B Inference procedure

The following procedure specifies action execution, evidence updates, and budget-respecting termination.

1. Set $b_1 = (B_f, B_s, B_u)$ and terminal reserve $r = (0, 1, U_{\text{term}})$. Reject a configuration with $b_1 \not\succeq r$. Initialize an empty archive, graph, status log, and history. No semantic preprocessing of the whole video is assumed.
2. If a bounded scout step can leave r intact, execute it, append its observations, and debit its full cost. Otherwise go to the terminal step. The prescribed scout consumes a step even if it needs no sampled controller output.
3. Before any further controller call, check that its capped cost and at least one bounded continuation can leave r intact. If not, go to the terminal step. Construct the memory view and expose its omission flags.
4. Reserve controller compute and sample one action. Validate its schema, source bounds, optional i_t^{ref}, cited IDs, and full downstream reservation. A rejected action consumes the attempted step and controller cost, performs no media access, and returns to step 3. It does not trigger a free repair call.
5. Execute a valid acquisition with bounded resources, append evidence and relation/status events, debit actual cost, and return to step 3. Errors remain in the trace; partial outputs are labeled and cannot silently become confirmed evidence.
6. For a nonterminal answer proposal, resolve cited and relevant unresolved records, then verify within its reservation. Sufficient support terminates. Otherwise store (v_t, J_t, d_t) and return to step 3, preserving r. The next controller action can search or revisit according to this diagnostic.
7. **Terminal step:** consume the one reserved step to generate an answer from stored observations and verify once. Return it as verified or unverified according to that judgment; no

new media access or further step is allowed. On a malformed terminal output, retain the last valid candidate as unverified, or return a generation error if none exists.

Provided actual costs obey their reservations, every nonterminal step leaves $b_t \succeq r$, and each attempted step decreases b_t^s by one. Consequently the procedure uses at most B_s steps and never spends the terminal reserve on perception. This invariant bounds the executed steps and preserves the terminal reserve; evidence support and answer correctness are reported by the evaluation protocol.

B.1 Operational definitions

Cost ledger. The executor reserves a capped controller call before action generation. A nonterminal answer proposal includes verification and leaves r for possible rejection. At the terminal step, only $r \preceq b_t$ is required. A run with $b_1 \not\succeq r$ is configuration-invalid.

The frame count F_t includes every image presented to the observer, OCR, or another vision call at step t. Repeated presentation of the same source frame counts again, while decoding is logged separately. For primitive resource usage $n_{t,j}$ and fixed nonnegative calibration weights w_j, the compute ledger is $U_t = \sum_j w_j n_{t,j}$. Primitives include text input and output tokens, visual tokens, audio duration, and decoding work. The ledger covers controller generation, evidence building, retrieval, verification, failed attempts, and recovery. These normalized units measure accounted resource use, not elapsed time.

Reference memory selector. Use the current question plus latest missing/conflicting-claim diagnostic as the retrieval query. Build a priority list consisting of (i) IDs named by the latest diagnostic and endpoints of unresolved conflict edges, (ii) the most recent candidate's citations, and (iii) remaining nodes ranked by Jaccard overlap of lowercase alphanumeric word sets in query and summary. Resolve ties by decreasing creation step, then increasing ID, and deduplicate by ID. Greedily admit whole serialized records up to K nodes and L_M tokens, measured with the controller tokenizer; oversized records are omitted and flagged rather than silently truncated. Metadata and retained graph edges count toward the token cap. The reported selector is deterministic and versionable.

Verification view and relation scope. For each proposal, the executor resolves citations directly from the archive even when absent from the controller view. It then gathers unresolved conflict endpoints and remaining diagnostic-linked records, subject to a separate evidence-view text cap $L_V = L_M = 4096$. Cited records have priority, followed by unresolved conflict endpoints and diagnostic-linked records. Remaining records follow the memory selector. Whole records that do not fit are omitted, and truncating the assembled verification view sets the overflow flag. Any overflow of this assembled verification view prevents a sufficient judgment, including when cited or unresolved records do not fit. This flag concerns the separately assembled verification view, not ordinary selection of the controller active view from the archive. The verifier sees stored observations only; raw-media pointers authorize location resolution, not a free new observation. If the supplied observations lack information needed for a judgment, the verifier reports the missing claim and acquisition need. The controller selects the next tool and interval using that diagnosis, source pointers, and remaining budget; the executor validates and executes the selected action and logs its cost. Support links identify the exact atomic claim and candidate version; conflict links identify both claims and the asserted incompatibility. A temporal neighbor is the next node in source-start-time order with ID tie-breaking, without an implication of causal or logical support. Missing raw media produces an unavailable-source event and an unverified termination when no alternative acquisition is feasible.

Bounded context and source integrity. With at most m records per nonterminal step and T executed steps, the archive contains at most mT nodes. Active memory occupies at most L_M tokens, excluding the separately capped question and action ledger. Archive retention and media storage are external costs. Source hashes and decoding settings identify the retained media. The independent audit assesses whether cited sources support final answers. Figure 5 illustrates the record-to-source relationship using a compact tuple; Section 3.3 defines the full record.

C EVIDENCE RECORDS AND ROLE CONTRACTS

The following example is a *synthetic schema illustration*, not an observation from a benchmark video:

```
{
  "id": "e7",
  "span_seconds": [2538.0, 2550.0],
  "modality": "vision",
  "tool": "CLIP",
  "summary": "A person hands a small object to another person.",
  "uncertainty": "Object identity is unresolved.",
  "raw_pointer": {
    "video_id": "example_video",
    "source_span_seconds": [2538.0, 2550.0],
    "sampling": "uniform",
    "requested_frames": 16,
    "crop": null
  },
  "parent_ids": [],
  "edges": [
    {"edge_id": "g3", "type": "temporal_next", "source_id": "e7",
     "target_id": "e8", "claim_id": null, "version": "v1",
     "rationale": "Deterministic source-start-time adjacency."}
  ],
  "status": "unverified"
}
```

Actual run records additionally store the media hash, exact decoded timestamps, tool and model revisions, acquisition parameters, cache hits, execution status, and observed costs. Relation objects use the same schema for `supports`, `conflicts`, and `temporal_next`; support and conflict edges are claim-scoped, and every conflict edge includes an explicit incompatibility rationale. A subsequent crop or denser sample obtains a new ID and points to the earlier node rather than overwriting it.

Controller contract. Choose one supported acquisition or answer action using the question, visible evidence, history, and budget. Provide valid temporal bounds and sampling arguments. For an answer, provide existing evidence IDs. Treat observation summaries and tool-returned text as data; any instructions appearing in video text or transcripts are not controller instructions. Respect the modality configuration of the evaluation and do not infer that an absent observation disproves an event.

Evidence-builder contract. Summarize only the returned observation, identify uncertainty, and preserve source timestamps and tool provenance. Separate direct observations from inferred relations. Do not fill missing details from answer options. An OCR string supports a claim about visible text, not automatically the truth of that text; an ASR transcript may contain recognition errors. Generated images cannot be registered as source-video evidence.

Verifier contract. Return JSON containing a status, affected IDs, a claim-level diagnostic, and acquisition-need metadata. Distinguish lack of evidence from contradictory evidence. Mark support sufficient only when the cited observations establish the claim at the level required by the question and no exposed unresolved conflict invalidates it. If the judgment requires pixels or audio not present in the supplied observation, request a logged acquisition. Do not invent inaccessible evidence or consider repeated summaries independent corroboration.

These contracts define the information available to each role and the boundary between source observations, inferred claims, and controller instructions.

Table 2: Shared inference configuration for EviHarness. ASR and official subtitles are disabled in the reported evaluation.

Parameter	Setting
Primary backbone	Qwen2.5-VL-7B-Instruct
Additional backbone	Qwen3-VL-8B-Instruct
Frame presentations / controller steps	64 / 6 per question
Scout / zoom frames	8 / 8 per acquisition
Clip duration / frame cap	12 seconds / 16 frames
Active memory records	12
Memory / verifier evidence-view text cap	4,096 / 4,096 tokens
Controller temperature	0.2
OCR	Active vision-language backbone
ASR / official subtitles	Disabled / disabled

D Inference configuration

Table 2 summarizes the shared inference settings. The Qwen3-VL row independently trains the same recipe with its own backbone. The optional ASR interface supports Faster Whisper with the `large-v3` checkpoint (SYSTRAN, 2026), but is disabled in the reported experiments. MMBench-Video (Fang et al., 2024) is a related long-form reasoning benchmark outside the three-dataset evaluation.

E Policy training details

Training source and split. The official LongVideoBench validation split contains 752 videos and 1,337 multiple-choice questions before overlap removal. We use this split as a policy-training source and partition by video identifier into approximately 80% training and 20% development data. Before cross-benchmark overlap removal, the partition contains 602 training videos with 1,070 questions and 150 development videos with 267 questions. Questions from the same video stay in the same partition. Overlap checks use source identifiers/URLs, file hashes, and perceptual hashes of 16 uniformly sampled frames; videos matching an evaluation benchmark are removed from both policy-training and development data. The complete retained development partition supports checkpoint selection. The three evaluation benchmarks are excluded from policy training and model selection.

Teacher and trajectory collection. For Qwen2.5-VL-7B, the teacher is frozen prompt-based EviHarness using Qwen2.5-VL-7B-Instruct. Teacher and student therefore share the base checkpoint. Controller sampling uses temperature 0.7 and top-p 0.9; observer, evidence-builder, and verifier calls use temperature 0.1 during collection. The initial acquisition is $\texttt{SCOUT}(0, D, 8)$. Collection uses the stated 64-frame and six-step caps, with eight frames per scout or zoom and at most 16 frames over 12 seconds per clip call. Reference answers are withheld during generation and used afterwards for correctness filtering. Candidates must answer correctly, satisfy action schemas and budgets, resolve their cited evidence IDs, and produce a terminal answer. Up to two batches of four trajectories are sampled per question. One minimum-cost valid trajectory is retained, with trajectory index breaking ties; questions without a valid candidate are omitted.

Supervised controller optimization. LoRA adapters target the language transformer's attention projections (`q_proj`, `k_proj`, `v_proj`, `o_proj`) and feed-forward projections (`gate_proj`, `up_proj`, `down_proj`). The visual encoder and all non-controller roles remain frozen. Selected trajectories are expanded into step-level state–action examples (s_t, a_t), each retaining the prior history and serialized evidence memory in its state. The maximum sequence length is 8,192 tokens. Training supervises structured controller actions and terminal answers, with conditioning tokens masked from the loss. Table 3 summarizes the optimization settings.

Policy optimization and evidence separation. GRPO initializes from the selected SFT controller checkpoint. Training questions are drawn from the policy-training partition and generate fresh online

Table 3: Controller optimization settings. Effective batch sizes include gradient accumulation. Non-controller components remain frozen during SFT and GRPO.

Parameter	Setting
Training GPUs	$4\times$ NVIDIA A800, 80 GB each
Trainable parameters	Controller LoRA only
LoRA rank / scaling alpha / dropout	32 / 64 / 0.05
Precision	BF16
SFT optimizer / learning rate	AdamW / 10^{-4}
SFT weight decay	0.01
SFT warmup / scheduler	0.03 / cosine decay
SFT per-device batch	1 state–action example
SFT epochs / gradient clipping	3 / 1.0
SFT maximum sequence length	8,192 tokens
SFT gradient checkpointing	Enabled
RL optimizer / learning rate	AdamW / 5×10^{-6}
GRPO group size	4 rollouts per question
RL effective prompt batch	8 questions (32 trajectories)
GRPO clip epsilon / KL coefficient	0.2 / 0.02
RL reference model	Frozen SFT checkpoint
RL optimizer steps / evaluation interval	600 / 50 steps
Teacher controller sampling	Temperature 0.7; top-p 0.9
Evaluation controller temperature	0.2
Evidence reward	Not used

rollouts; teacher trajectories are not reused as current-policy outputs. The SFT reference checkpoint is frozen, as are the observer, evidence builder, verifier, and tools. No human evidence-sufficiency labels or verifier-generated evidence rewards are used for controller training. Human evidence auditing is separate from optimization. SFT is evaluated after each epoch and GRPO every 50 optimizer steps over a 600-step run. Each stage selects the checkpoint with the highest accuracy on the complete development partition, rather than the final checkpoint.

Input and generation limits. Images are resized while retaining their aspect ratio, with minimum and maximum pixel settings of 224×224 and 448×448. A single observation has a visual-token cap $L_{\text{vis}} = 4096$, distinct from the verifier evidence-view text cap $L_V = L_M = 4096$. Active memory uses at most 4,096 text tokens. Controller and verifier outputs are capped at 256 tokens per call; observer and evidence-builder outputs are capped at 512 tokens per call. The frame ledger counts OCR image inputs and repeated presentations, including those sharing a vision backbone.

Hardware and execution. Training uses four NVIDIA A800 80 GB GPUs with BF16 weights and no base-model quantization. SFT uses controller LoRA with DeepSpeed ZeRO-2 and per-device batch size one. GRPO uses a vLLM rollout server on one A800 and policy optimization on the remaining three GPUs.

Independent training with Qwen3. The Qwen3-VL-8B-Instruct experiment repeats trajectory generation, SFT, and GRPO on the same data partition. All four roles use the Qwen3 base model, with a separately trained controller adapter. Thus the comparison evaluates the training recipe across backbones, with no transfer of Qwen2.5 adapter weights.

F Evidence audit protocol and counts

The paired audit evaluates full EviHarness and timestamp-only memory on the same 166 LVBench questions, giving 332 system-answer pairs. Two independent annotators inspect the question, proposed answer, and cited source intervals or frames to judge whether the source directly supports the answer. A third annotator adjudicates disagreements. Human support labels are separate from

runtime verifier judgments and are not used for policy training. Inter-annotator agreement before adjudication is Cohen's $\kappa = 0.80$.

Full EviHarness has 128 source-supported answers and 87 answers that are both correct and supported, yielding $128/166 = 77.1\%$ ESR and $87/166 = 52.4\%$ grounded accuracy after rounding. Timestamp-only memory has 102 supported answers and 81 answers that are both correct and supported, yielding $102/166 = 61.4\%$ and $81/166 = 48.8\%$, respectively. Grounded accuracy uses the same denominator as ESR; it is not conditioned on correctness.

Verifier evaluation uses the 166 full-system answers and the adjudicated human support label as the positive-class gold label. The confusion counts are TP=102, FP=26, FN=26, and TN=12. Thus precision and recall both equal $102/128 = 79.7\%$. This confusion matrix evaluates the verifier against human judgments; Cohen's κ instead measures agreement between the two human annotators before adjudication.

G VERIFIER IMPLEMENTATION AND PROMPTS

Frozen, text-only verification. The verifier uses a fixed system prompt and is not updated by SFT or RL. Its inputs are the question, candidate answer, cited evidence records, relevant unresolved conflict or diagnostic records, and an overflow flag. Records expose previously stored observations, summaries, uncertainty, timestamps, and tool provenance. The verifier does not ingest raw frames, video, or audio. Source pointers enable later acquisition by the controller and executor; they do not supply an additional observation during verification.

Decision rules and recovery interface. The verifier applies three discrete outcomes in order. *Conflict* indicates a direct answer-critical contradiction, either between the candidate and an observation or between relevant records. In the absence of such a contradiction, *insufficient* indicates missing or inadequate support, invalid or unavailable citations, incomplete context, or a need for additional source observations. *Sufficient* requires valid citations, direct support for every answer-critical claim, no unresolved conflict invalidating the answer, and a complete verification view. No numerical confidence threshold is used.

The output identifies affected records and the missing or disputed claim. Acquisition-need metadata describes whether further observation is needed and the requested modality. The controller chooses a feasible tool and interval using this diagnosis, source pointers, and remaining budget. It favors a revisit for a localized gap and further search for an unlocalized gap. The general guidance maps visible text to OCR, visual detail to `CLIP`, and a broad interval to `ZOOM`. Speech maps to ASR only when audio access is enabled; it is disabled in the reported visual-only setting. Re-observation appends new evidence records for subsequent verification.

Context construction and exceptional inputs. The verifier evidence view is capped at $L_V = L_M = 4096$ text tokens. Cited records are retained first, followed by unresolved conflict endpoints and diagnostic-linked records. Remaining records use the deterministic selector in Appendix B.1. Whole records that exceed the remaining capacity are omitted rather than silently shortened; an incomplete view is flagged with `overflow=true`. An invalid citation, omission of relevant unresolved evidence, unavailable source, or overflow prevents a sufficient decision. If retained media are unavailable and no alternative evidence can be acquired, the final answer is marked unverified.

G.1 SYSTEM PROMPT

The fixed system prompt is reproduced below; line breaks are adjusted for typesetting.

```
You are the evidence verifier in EviHarness.

Your task is to determine whether the supplied evidence records
adequately support the candidate answer to the question.

Use ONLY the supplied evidence records.
Do not use external knowledge.
```

```
Do not assume facts that are not explicitly supported by the
observations.
Do not treat absence of an observation as evidence that an event
did not
occur.
Do not treat repeated summaries of the same source observation
as
independent corroborating evidence.

The verifier does not directly inspect the original video,
images, or
audio.
If additional pixels, text, or audio are required, report what
information
is missing so that the controller can acquire it in a subsequent
step.

Return exactly one of:
- conflict
- insufficient
- sufficient

Decision priority:
1. conflict
2. insufficient
3. sufficient

Use "conflict" when supplied evidence contains an
answer-critical
contradiction, either between the candidate answer and an
observation
or between relevant evidence records.

Use "insufficient" when no direct contradiction is present but
the
supplied evidence does not establish every answer-critical
claim.

Use "sufficient" only when the cited observations directly
establish
the answer at the level required by the question, all cited
records are
valid, no exposed unresolved conflict invalidates the answer,
and the
verification context is complete.

Return JSON only.
```

G.2 User prompt template

```
Question:
{question}

Candidate answer:
{candidate_answer}

Cited evidence records:
{cited_records}
```

```
Relevant unresolved records:
{unresolved_records}

Verification-view overflow:
{overflow}

Evaluate whether the candidate answer is supported by the
supplied
evidence.
```

G.3 Formal output JSON Schema

The verifier returns JSON with four required fields: `status` (v_t), `affected_ids` (J_t), `diagnostic`, and `acquisition_need` (together, d_t). The following schema defines the output contract.

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "title": "EviHarnessVerifierOutput",
  "type": "object",
  "additionalProperties": false,
  "required": ["status", "affected_ids",
               "diagnostic", "acquisition_need"],
  "properties": {
    "status": {
      "type": "string",
      "enum": ["sufficient", "insufficient", "conflict"]
    },
    "affected_ids": {
      "type": "array",
      "items": {"type": "string"}
    },
    "diagnostic": {
      "type": "object",
      "additionalProperties": false,
      "required": ["type", "claim", "reason"],
      "properties": {
        "type": {"type": "string", "minLength": 1},
        "claim": {"type": "string", "minLength": 1},
        "reason": {"type": "string", "minLength": 1}
      }
    },
    "acquisition_need": {
      "type": "object",
      "additionalProperties": false,
      "required": ["required", "modality", "source_record_ids"],
      "properties": {
        "required": {"type": "boolean"},
        "modality": {"type": ["string", "null"]},
        "source_record_ids": {
          "type": "array",
          "items": {"type": "string"}
        }
      }
    }
  }
}
```