VIDEOGYM: TRAINING MEMORY-AUGMENTED VIDEO AGENTS WITH MODALITY-AWARE CURRICULA

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-video QA involves sparse evidence distributed over extended multimodal contexts, making retrieval-based memory a natural interface for revisiting relevant cues instead of relying on fixed keyframes. We instantiate this interface as VideoAgent, a memory-augmented agent that indexes a video into text, graph, and visual stores and retrieves evidence on demand through a multi-turn planner, with the planner trained by reinforcement learning (RL) to coordinate retrieval across modalities. This training, however, is limited by the long-tail modality distribution of video QA training data: in our source question pool, text-heavy questions dominate at $\approx$58% of samples, while graph-, visual-, and time-range questions appear in only $\approx$15%, $\approx$17%, and $\approx$10% respectively. Fixed-distribution RL therefore concentrates updates on the already-frequent text mode and under-trains the remaining modalities, and reweighting the raw questions alone recycles the same rare-modality prompts rather than adding new ones. We introduce VideoGym, a curriculum-based training environment that adapts VideoAgent's training distribution online: it casts per-period sample selection as an adversarial bandit over (modality, mode) arms and applies the EXP3 update to shift sampling toward arms with low answer accuracy. To supply the augmented arms, a TaskGenerator-Verifier pipeline produces harder variants of existing questions that remain answerable within the same video. Across five long-video benchmarks, VideoGym lifts RL accuracy from 60.4% to 64.8% on average. Gains concentrate on the under-represented modalities: $+10.5$ pp on graph-heavy and $+7.5$ pp on visual-heavy questions, at $3.3\times$ sample efficiency.

1 INTRODUCTION

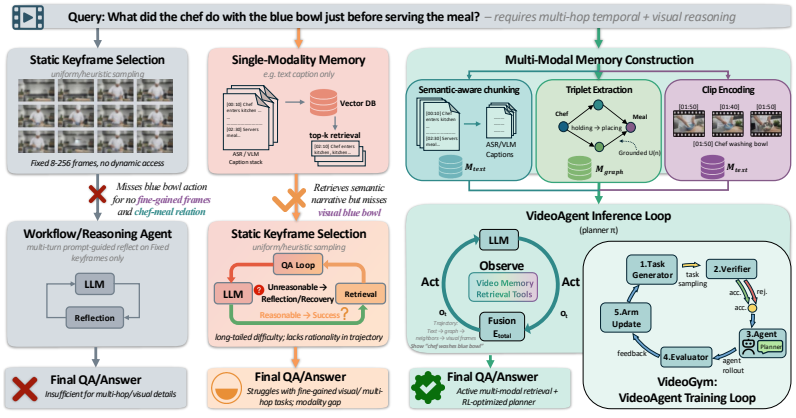

Figure 1: **From passive video input to curriculum-based evidence management.**

Long-form video understanding remains a fundamental challenge for current multimodal large language models (LLMs). In hour-scale videos, where relevant evidence is dispersed across distant temporal segments (Figure 1), methods that pre-select a fixed set of frames or compress them into a fixed token budget cannot reliably cover the signal a query depends on (Fu et al., 2025; Wang et al., 2025a). Such single-pass designs handle closed-ended or descriptive queries adequately, but break

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.

down on questions that require active evidence-seeking, such as locating a specific object behavior, tracking an entity's state across scenes, or verifying a temporal claim spanning non-contiguous segments. The limitation is one of observation strategy: a single-pass agent processes whatever the sampler returns and cannot revisit or re-query the video when its initial observation falls short.

A growing body of work reframes long-video understanding as a retrieval problem over structured memory, where the agent indexes the video offline and queries it on demand rather than ingesting the full stream at once (Ren et al., 2026; Zhang et al., 2025a; Long et al., 2026; Yeo et al., 2026). Building on this paradigm, we instantiate VIDEOAGENT with three complementary memory modalities, each targeting a distinct kind of evidence: text memory over captions and transcripts handles factual recall and narrative summarization, graph memory over entity-relation triples supports multi-hop reasoning and cross-segment entity tracking, and visual memory over frame-level embeddings enables appearance verification and fine-grained temporal localization. A think-act-observe planner (Yao et al., 2023) coordinates these memories across turns, deciding which modality to query next based on accumulated evidence and terminating when the question becomes answerable.

However, equipping an agent with retrieval tools does not ensure it learns to use them effectively under standard training. Conan (Ouyang et al., 2026), for example, trains the retrieval policy with outcome-based reinforcement learning with verifiable rewards (RLVR) on a fixed question distribution, and video QA training data has a long tail over retrieval modalities. In our source question pool, text-based queries account for ≈58% of training samples, while graph-heavy and visual-heavy questions constitute only ≈15% and ≈17%, respectively. The consequence for trained model behavior is direct: supervised fine-tuning (SFT) yields 70.1% accuracy on text-heavy questions but only 44.2% on graph-heavy and 46.8% on visual-heavy questions (Table 2). The performance gap tracks the imbalance in the training distribution.

We address this distributional gap with VIDEOGYM, a curriculum-based RL training environment for VIDEOAGENT that reformulates training allocation as a sequential decision problem: at each curriculum period, the choice of which (modality, mode) arm to draw from is itself a learned action. Its design responds to two properties of long-video QA that distinguish it from the text-based domains in which adaptive curricula were originally developed (Guo et al., 2026; Zeng et al., 2026; Lu et al., 2026; Huang et al., 2025). First, the accuracy of each arm changes as the policy learns. We therefore cast per-period arm selection as an adversarial bandit problem and apply the EXP3 update (Auer et al., 2002b) to redirect sampling weight toward arms with the lowest current accuracy. Second, the bandit can only redistribute the prompts it already has, so once it upweights the graph- and visual-heavy arms it keeps drawing the same questions. We therefore augment these arms with harder variants. Unlike math operands or code templates, synthesized questions must remain grounded in the source video to preserve answerability. We address this with a TASKGENERATOR, an offline LLM that rewrites existing questions conditioned on the source video's captions and entity graph, and a VERIFIER, an offline LLM-based filter that rejects rewrites failing answerability or duplication checks before they enter training.

Our main contributions are as follows:

1. We build **VIDEOAGENT**, a memory-augmented video agent that decomposes a long video into text, graph, and visual indices and retrieves evidence on demand through a multi-turn planner.
2. We introduce **VIDEOGYM**, which casts multi-modal training allocation as an adversarial bandit over modality-augmentation arms and applies the EXP3 update to continuously redirect training budget toward underperforming retrieval modalities. A TASKGENERATOR–VERIFIER pipeline supplies the upweighted arms with harder rewrites validated for answerability against the source video.
3. Across five long-video benchmarks, **VIDEOGYM** yields gains of +10.5 pp on graph-heavy and +7.5 pp on visual-heavy questions over fixed-distribution RL at 3.3× sample efficiency, with the largest improvements concentrated in the modalities that fixed sampling under-trains.

2 RELATED WORK

Memory-augmented video agents. Recent work has explored structured memory and retrieval mechanisms for long-form video understanding. VideoRAG (Ren et al., 2026) constructs graph-grounded textual and visual indices for retrieval-augmented generation over hours-long videos. Deep

Video Discovery (Zhang et al., 2025a) represents videos as a multi-granularity database that can be navigated by an agent loop. M3-Agent (Long et al., 2026) maintains entity-centric episodic and semantic memory through continuous observation and trains multi-turn memory retrieval with reinforcement learning. WorldMM (Yeo et al., 2026) further builds heterogeneous memory across multiple temporal scales for adaptive cross-modal retrieval. Conan (Ouyang et al., 2026) further advances this direction by training the agent with progressive RL to improve how it navigates evidence. VIDEOGYM targets modality imbalance by maintaining an *online*, feedback-driven curriculum over modality types.

Agentic RL and training environments. Research on agent learning spans outcome-based reasoning (Guo et al., 2025a), multi-turn interaction and learning across diverse environments (Yao et al., 2023; Xi et al., 2025), and environment-centric training where task and feedback generation are co-designed with the policy (Huang et al., 2025; Guo et al., 2026; Zeng et al., 2026; Lu et al., 2026). In video settings, the frequency of retrieval modality types depends strongly on the data source and is rarely aligned with the modalities that carry the greatest remaining learning signal. To address this long-tail modality skew, we are inspired by the multi-armed bandit (MAB) framework, which balances exploration and exploitation in dynamic resource allocation (Auer et al., 2002a). Stochastic bandit rules such as KL-UCB and Thompson sampling (Garivier & Cappé, 2011; Agrawal & Goyal, 2012) assume a fixed reward distribution per arm, and switching-bandit variants (Garivier & Moulines, 2011) allow only occasional changes. Here the per-arm reward changes with every policy update, so we cast per-period allocation as an adversarial bandit (Audibert & Bubeck, 2009) and use the exponential-weight update of EXP3 (Auer et al., 2002b). EXP3 lets VIDEOGYM rebalance the training distribution online by shifting sampling weight toward underperforming modality arms based on correctness feedback.

3 METHOD

3.1 PROBLEM FORMULATION

We formulate long-video QA as memory-grounded, multi-step evidence retrieval. A training instance is denoted by $s = (\mathcal{V}, x, y, c)$, where $\mathcal{V}$ is a long video, x is the question, y is the reference answer, and $c \in \mathcal{C}$ denotes the dominant evidence type required by the question. We use

$$\mathcal{C} = \{\textit{text}, \textit{graph}, \textit{visual}, \textit{time-range}\}.$$

Episode-level interaction. Given $\mathcal{V}, x$, an episode-level policy π_θ interacts with an external memory $\mathcal{M}(\mathcal{V})$ of $\mathcal{V}$ through actions $u_k \in \mathcal{A}_{\text{agent}}$. At turn k, the policy observes the context $(x, h_{<k})$ with history $h_{<k} = (u_1, e_1, \ldots, u_{k-1}, e_{k-1})$, selects u_k, and either receives evidence e_k or terminates the episode by emitting $u_T = \texttt{answer}$ together with the prediction $\hat{y}$. The resulting evidence trajectory is

$$\tau = (u_1, e_1, \ldots, u_{T-1}, e_{T-1}, u_T, \hat{y}), \qquad u_T = \texttt{answer},$$

with episode length $T \le T_{\max}$ capped by a maximum turn budget. We denote by $\mathcal{A}_{\text{ret}} \subset \mathcal{A}_{\text{agent}}$ the set of retrieval actions, defined in Section 3.2, and define the number of retrieval turns as

$$T_{\text{search}} = \sum_{k=1}^{T-1} \mathbf{1}[u_k \in \mathcal{A}_{\text{ret}}].$$

The action space $\mathcal{A}_{\text{agent}}$ and the memory $\mathcal{M}(\mathcal{V})$ are concretized in Section 3.2.

Curriculum-level decision. Training π_θ further requires deciding, at each curriculum period, which training instance to roll out. We define the curriculum arm space

$$\mathcal{A}_{\text{gym}} = \mathcal{C} \times \{\textit{raw}, \textit{aug}\},$$

with arms $z = (c, m) \in \mathcal{A}_{\text{gym}}$. The first factor selects the question type c. The second selects between an existing instance from a per-type bank $\mathcal{B}_c$ ($m = \textit{raw}$) and a model-generated variant ($m = \textit{aug}$). We use distinct symbols u_k and z for the planner action and the curriculum sampling decision, since they live in different spaces and serve different roles. Section 3.3 adapts the distribution $p^{(t)}$ over $\mathcal{A}_{\text{gym}}$ online via an adversarial bandit.

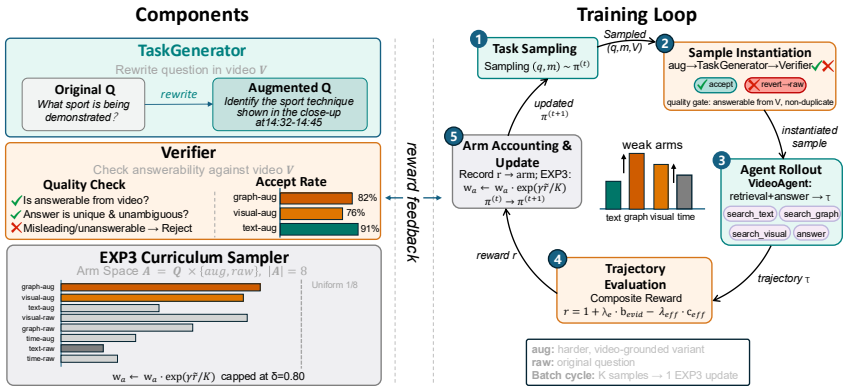

Figure 2: **VIDEOGYM training loop.**

3.2 VIDEOAGENT: MEMORY-AUGMENTED LONG-VIDEO AGENT

As shown in Figure 1, VIDEOAGENT defines the memory and action interface used by the policy. It indexes each video into complementary memory stores and exposes them to a multi-turn planner.

Video memory. For each video $\mathcal{V}$, VIDEOAGENT builds a cached external memory

$$\mathcal{M}(\mathcal{V}) = (\mathcal{M}_{\text{text}}, \mathcal{M}_{\text{graph}}, \mathcal{M}_{\text{visual}}),$$

which is constructed once offline and reused during SFT, RL rollouts, and evaluation. The three stores are organized around the kinds of evidence that long-video questions require. $\mathcal{M}_{\text{text}}$ indexes segment-level captions and ASR transcripts, covering evidence that is explicitly described in speech, narration, or generated captions. $\mathcal{M}_{\text{graph}}$ lifts caption-derived entities and relations into a linked graph, allowing the planner to track recurring entities and reason over relations that appear in non-contiguous segments. $\mathcal{M}_{\text{visual}}$ stores timestamped frame or segment embeddings, preserving appearance-level cues that may be absent from textual descriptions and supporting localized visual re-observation.

Together, these memories provide broad and complementary evidence coverage over the same video. This design allows the planner to match the retrieval interface to the evidence form required by each question, rather than forcing all questions through a single video representation.

Action interface. The planner action space is

$$\mathcal{A}_{\text{agent}} = \{\texttt{search_text}, \texttt{search_graph}, \texttt{search_visual}, \texttt{search_time_range}, \texttt{answer}\}.$$

The first three actions query the corresponding memory store and return ranked evidence snippets. The action `search_time_range` performs a windowed temporal scan, which is useful when the answer depends on a continuous event rather than a single retrieval hit. The action `answer` terminates the episode and emits $\hat{y}$.

Planner loop. VIDEOAGENT follows a Think–Action–Observation interaction pattern. At each turn, the model produces a planner state, emits one structured action tag, and observes the retrieved evidence. The observation is then appended to the evidence buffer and becomes available in later turns.

We apply the same rollout constraints during SFT trace collection, RL training, and evaluation. Repeated retrieval queries are blocked to avoid loops. Malformed action tags trigger a retry or a blocked-action record. If the planner does not emit `answer` within $T_{\max}$ turns, the environment forces an answer from the current evidence buffer.

3.3 VIDEOGYM: MODALITY-AWARE CURRICULUM

Fixed-distribution RL inherits the long-tail modality distribution of the training data: frequent text-heavy questions dominate the update budget, while graph- and visual-heavy questions receive too

Algorithm 1 VIDEOGYM training loop over one curriculum period t

Require: question draws B; group size G; question banks $\mathcal{B} = \{\mathcal{B}_c\}_{c\in\mathcal{C}}$; policy π_θ; arm weights $\{w_z^{(t)}\}$; sampling distribution $p^{(t)}$
Ensure: rollout buffer $\mathcal{D}^{(t)}$; updated weights $\{w_z^{(t+1)}\}$; updated distribution $p^{(t+1)}$

$\mathcal{D}^{(t)} \leftarrow [\,]$ {trajectories and shaped rewards for GRPO}
$\mathcal{S}^{(t)} \leftarrow [\,]$ {arm correctness for EXP3}
draw arms $z_i = (c_i, m_i) \sim p^{(t)}$ independently for $i = 1, \ldots, B$
for $i = 1, \ldots, B$ **do**
  draw seed instance $s_i = (\mathcal{V}_i, x_i, y_i, c_i)$ from $\mathcal{B}_{c_i}$
  if $m_i =$ *aug* **then**
    $\tilde{s}_i \leftarrow \textsc{TaskGenerator}(s_i, \mathcal{M}_{\text{text}}, \mathcal{M}_{\text{graph}})$
    if $\textsc{Verifier}(\tilde{s}_i, \mathcal{M}_{\text{text}}, \mathcal{M}_{\text{graph}})$ **then**
      $s_i \leftarrow \tilde{s}_i$
    else
      $z_i \leftarrow (c_i,$ *raw*$)$ {failed generation is treated as a raw draw}
    end if
  end if
  let $y_i^\star$ be the reference answer of s_i
  for $j = 1, \ldots, G$ **do**
    $\tau_{ij} \leftarrow \textsc{Rollout}(\pi_\theta, s_i, \mathcal{M}(\mathcal{V}_i))$
    $\rho_{ij} \leftarrow \mathbf{1}[\textsc{Match}(\hat{y}_{ij}, y_i^\star) = 1]$ {answer correctness for curriculum}
    $R_{ij} \leftarrow \textsc{CompositeReward}(\tau_{ij}, \rho_{ij})$ {shaped reward for GRPO}
    append $(i, s_i, \tau_{ij}, R_{ij})$ to $\mathcal{D}^{(t)}$
    append (z_i, ρ_{ij}) to $\mathcal{S}^{(t)}$
  end for
end for
compute $\bar{\rho}_z^{(t)}$ for each sampled arm z from $\mathcal{S}^{(t)}$
update observed-arm weights and $p^{(t+1)}$ via Eqs. (1)–(3); keep unobserved-arm weights unchanged
clear $\mathcal{S}^{(t)}$
return $\mathcal{D}^{(t)}$, $\{w_z^{(t+1)}\}$, $p^{(t+1)}$

few rollouts to improve reliably. As shown in Figure 2, VIDEOGYM addresses this by adapting the training distribution online. At each curriculum period, it decides which evidence type to train on and whether to draw a raw question or a verified augmented variant.

Let $\mathcal{B} = \{\mathcal{B}_c\}_{c\in\mathcal{C}}$ be the raw question bank partitioned by dominant evidence type. Using the curriculum arm space defined in Section 3.1, an arm $z = (c, m)$ specifies both the evidence type c and the sample mode m. If $m =$ *raw*, VIDEOGYM draws an existing instance from $\mathcal{B}_c$. If $m =$ *aug*, it draws a seed instance from $\mathcal{B}_c$ and attempts to generate a harder question grounded in the same video.

Adaptive sampling. Per-arm reward distributions are non-stationary across curriculum periods, since π_θ is updated continuously during training. VIDEOGYM therefore formulates allocation as an adversarial bandit and applies the EXP3 update (Auer et al., 2002b). The curriculum signal is the binary final-answer correctness $\rho_{ij} = \mathbf{1}[\textsc{Match}(\hat{y}_{ij}, y_i) = 1]$ of rollout j on instance i, where MATCH is the benchmark's answer-matching procedure.

VIDEOGYM maintains weights $\{w_z^{(t)}\}$ and distribution $p^{(t)}$ over $\mathcal{A}_{\text{gym}}$, initialized uniformly with $w_z^{(0)} = 1$ and $p_z^{(0)} = 1/K$, where $K = |\mathcal{A}_{\text{gym}}| = |\mathcal{C}| \times 2 = 8$. For an arm with $N_z^{(t)} > 0$ attributed trajectories in period t, its accuracy is $\bar{\rho}_z^{(t)} = (N_z^{(t)})^{-1} \sum_{i:z_i=z} \sum_{j=1}^{G} \rho_{ij}$. Unobserved arms keep their weights. The bandit converts accuracy into a capped weakness signal and applies the EXP3 update:

$$\ell_z^{(t)} = \min\big(1 - \bar{\rho}_z^{(t)},\, \delta\big), \qquad \tilde{\ell}_z^{(t)} = \ell_z^{(t)} / p_z^{(t)}, \tag{1}$$

$$w_z^{(t+1)} = w_z^{(t)} \exp\big(\gamma\, \tilde{\ell}_z^{(t)} / K\big), \tag{2}$$

$$p_z^{(t+1)} = (1-\gamma)\, \frac{w_z^{(t+1)}}{\sum_{z'} w_{z'}^{(t+1)}} + \frac{\gamma}{K}, \tag{3}$$

with $\gamma = 0.1$ and learnability cap $\delta = 0.80$ applied *before* the importance-weight divide in Eq. (1), bounding the period-wise error signal (Appendix B.3). The update directs sampling toward lower-accuracy arms, and the γ/K mixing keeps every arm reachable.

Source-grounded augmentation. For an augmentation arm $z = (c, \textit{aug})$, VIDEOGYM draws a seed instance $s = (\mathcal{V}, x, y, c)$ from $\mathcal{B}_c$ and uses TASKGENERATOR to produce a candidate $\tilde{s} = (\mathcal{V}, \tilde{x}, \tilde{y}, c)$. The generator is conditioned on the caption and entity-graph stores of the source video and prompted to preserve the evidence type while increasing difficulty. VERIFIER then rejects candidates that are malformed, unsupported, or near-duplicate. For rejected candidates, VIDEOGYM falls back to the seed raw instance and reassigns the arm to $(c, \textit{raw})$. This keeps the rollout budget fixed and ensures that augmentation-arm statistics reflect valid generated questions rather than generation failures. The prompt schemas for both modules are given in Appendices C.2 and C.3. Algorithm 1 summarizes the curriculum period, with two output streams of a rollout buffer for GRPO optimization (Section 3.4) and per-arm accuracy statistics for the curriculum update.

3.4 POLICY OPTIMIZATION

We train VIDEOAGENT with two-stage SFT followed by GRPO. First, Planner SFT supervises intermediate turns from teacher trajectories: given the question and current evidence history, the model learns to emit the next structured retrieval action. Second, Answer SFT supervises only the final answer conditioned on the accumulated evidence. This separation discourages the model from bypassing retrieval and answering directly from parametric knowledge.

Starting from the SFT checkpoint, we optimize the policy with GRPO. For each sampled instance, VIDEOGYM selects the curriculum arm and constructs the rollout input, while GRPO updates the policy from trajectory-level rewards. The reward used for policy optimization is distinct from the binary correctness ρ used by EXP3. We define

$$R(\tau) = \rho\,(1 + \lambda_e\, b_{\text{evid}} - \lambda_{\text{lazy}}\, \mathbf{1}[T_{\text{search}} = 0]) - \lambda_{\text{eff}}\, c_{\text{eff}}, \tag{4}$$

where $b_{\text{evid}} \in [0, 1]$ denotes the fraction of retrieval turns whose returned snippet supports the ground-truth answer. It is applied only to correct episodes so that lucky guesses do not receive evidence credit. The term $c_{\text{eff}} \in [0, 1]$ is defined as a weighted sum of the normalized turn count, the forced-termination indicator, the blocked-tag ratio, and the format-retry ratio. We use $\lambda_e{=}\lambda_{\text{eff}}{=}0.3$ and $\lambda_{\text{lazy}}{=}0.2$ (Appendix B.4). The composite form, rather than a binary ρ, can preserve within-group GRPO advantages on arms where SFT has already saturated accuracy, sustaining gradient signal on the remaining hard cases. Thus VIDEOGYM controls which modality and sample mode the agent trains on, while the composite reward shapes how the agent retrieves evidence within an episode. The two levels of feedback are decoupled by construction: the curriculum reads only answer correctness, and the reward is used only by the policy update.

4 EXPERIMENTS

Our evaluation addresses two questions. *(i) Does* VIDEOGYM *improve the agent over both the underlying architecture and standard training?* (Section 4.2). *(ii) If so, does the gain come from the long-tail modalities that the curriculum reweights?* We decompose by retrieval modality (Section 4.3), ablate components (Section 4.4), and analyze curriculum dynamics (Section 4.5).

4.1 EXPERIMENTAL SETUP

Benchmarks and baselines. We evaluate on five benchmarks spanning broad long-video understanding (LVBench (Wang et al., 2025a), Video-MME (Long) (Fu et al., 2025)) and reasoning-intensive video QA (EgoLifeQA (Yang et al., 2025), Ego-R1 Bench (Tian et al., 2026), HippoVlog (Lin et al., 2026)), including egocentric, memory-heavy, and multi-hop settings. Questions are labeled by dominant evidence type using the same protocol as the per-type banks $\{\mathcal{B}_c\}_{c \in \mathcal{C}}$. We compare with long-video LLMs, RAG-based methods, and memory-based video agents, all re-evaluated under the same protocol from released checkpoints.

Training data. Both SFT and RL draw prompts from the multiple-choice portion of LLaVA-Video-178K (Zhang et al., 2025b), whose source question pool exhibits a long-tail across $\mathcal{C}$ (text 58%,

Table 1: **Main results on five long-video benchmarks.** Accuracy (%). All systems are evaluated by us under the same multiple-choice protocol from their released code and checkpoints. Bold: best result per column. Blue rows : our system across training stages. RL ± is the three-seed standard deviation of the five-benchmark mean.

Method	Size	EgoLife QA	Ego-R1 Bench	Hippo Vlog	LV Bench	Video-MME (Long)	Avg.
Long-video models							
VideoChat-Flash (Li et al., 2026)	7B	34.6	41.3	61.2	47.9	54.8	48.0
Time-R1 (Wang et al., 2025b)	7B	45.1	43.6	57.4	40.2	52.7	47.8
Video-RTS (Wang et al., 2025c)	7B	45.7	44.8	62.3	45.1	54.4	50.5
RAG-based methods							
LightRAG (Guo et al., 2025b)	–	43.2	42.5	47.1	31.6	42.4	41.4
Video-RAG (Luo et al., 2025)	7B	43.9	41.7	59.3	38.5	54.9	47.7
HippoRAG (Gutiérrez et al., 2024)	–	55.2	52.4	60.6	45.3	48.7	52.4
Memory and retrieval agents							
EgoRAG (Yang et al., 2025)	–	36.1	40.4	56.2	34.6	43.5	42.2
Conan (Ouyang et al., 2026)	7B	38.7	40.9	63.4	38.8	52.6	46.9
M3-Agent (Long et al., 2026)	8B	50.4	47.6	66.2	43.7	50.1	51.6
Ego-R1 (Tian et al., 2026)	3B	44.6	45.9	64.3	43.5	63.8	52.4
HippoMM (Lin et al., 2026)	–	52.3	50.6	**77.2**	42.4	48.3	54.2
WorldMM-8B (Yeo et al., 2026)	8B	53.6	49.5	67.4	52.3	63.5	57.3
Ours: training progression							
VIDEOAGENT (training-free)	8B	54.1	51.5	66.8	49.1	54.0	55.1
+ SFT	8B	55.8	54.3	68.2	51.8	59.4	57.9
+ RL w/o VIDEOGYM	8B	57.4	55.6	71.5	53.7	64.0	60.4 ± 0.5
+ RL w/ VIDEOGYM	8B	**61.5**	**59.9**	75.5	**58.3**	**69.0**	$\mathbf{64.8} \pm 0.6$

Table 2: **Per-modality accuracy breakdown.** RL rows are three-seed means. Δ: gain of RL w/ VIDEOGYM over RL w/o VIDEOGYM. Bold Δ: >5pp.

Method	Text-heavy	Graph-heavy	Visual-heavy	Time-range
Training-free	67.3	41.8	43.9	53.2
+ SFT	70.1	44.2	46.8	56.4
+ RL w/o VIDEOGYM	72.6	47.0	49.5	58.8
+ RL w/ VIDEOGYM	73.9	57.5	57.0	64.2
Δ	+1.3	**+10.5**	**+7.5**	**+5.4**

visual 17%, graph 15%, time-range 10%). A GPT-4o-backed teacher emits multi-turn agent traces, and structural quality filtering retains ∼96K accepted trajectories for two-stage SFT (Planner then Answer). RL trains on a much smaller subset of ∼2K rollout prompts randomly sampled from the same pool, with annotation rules in Appendix A.2.

Implementation. We initialize from Qwen3-VL-8B (Bai et al., 2025). Video memory is built with Whisper (Radford et al., 2023) (ASR), Qwen3-VL (captioning), and ImageBind (Girdhar et al., 2023) (visual embeddings). Two-stage SFT trains for 2 epochs on trajectories from a GPT-4o-backed teacher. RL uses GRPO (Shao et al., 2024) with group size $G = 8$, $T_{\max} = 10$, and EXP3 exploration $\gamma = 0.1$. Reward weights $\lambda_e = \lambda_{\text{eff}} = 0.3$. RL w/o VIDEOGYM samples the same ∼2K prompts at their natural distribution. All RL variants start from the same SFT checkpoint and use the composite reward of Eq. (4). All training and evaluation run on a single 8×H100-80G node. Per-stage compute and wall-clock are reported in Appendix E.1.

4.2 MAIN RESULTS

Table 1 traces our system through its training stages. The training-free agent reaches 55.1% average accuracy, two-stage SFT raises it to 57.9%, and RL without the curriculum raises it to 60.4%. Adding VIDEOGYM raises it to **64.8%**, a gain of 4.4 pp that is larger than the 2.5 pp from RL alone. It is also the highest average in Table 1, 7.5 pp above the strongest baseline, WorldMM-8B. Fixed-distribution

Table 3: **VIDEOAGENT component ablation (inference-time).** Each row disables one memory at inference. "Single-shot" removes the planner loop. Averages over the two benchmarks shown only (seed-1 checkpoint).

Configuration	EgoLifeQA	LVBench	Avg.
Full system (RL w/ VIDEOGYM)	62.1	58.7	60.4
w/o Graph Memory	60.3	55.2	57.8
w/o Visual Memory	61.2	56.8	59.0
w/o Text Memory	58.4	51.2	54.8
Single-shot (no planner)	58.9	52.4	55.7

Table 4: **VIDEOGYM component ablation.** Average accuracy over five benchmarks. "Uniform" replaces EXP3 with uniform sampling. "No aug" disables augmentation. "No verifier" disables quality filtering.

Configuration	Avg.
RL w/o VIDEOGYM	60.4
VIDEOGYM: uniform sampling	62.2
VIDEOGYM: EXP3, no aug	63.4
VIDEOGYM: aug, no verifier	64.3
VIDEOGYM: full (ours)	**64.8**

RL spends its gradient in proportion to the data distribution. The curriculum concentrates it on the under-represented modalities, as Section 4.3 shows.

4.3 LONG-TAIL MODALITY BREAKDOWN

If the curriculum works as designed, the gain should be largest on the modalities where the SFT agent starts weakest. Table 2 decomposes the 4.4 pp improvement by dominant retrieval modality: $+10.5$ pp on graph-heavy, $+7.5$ pp on visual-heavy, $+5.4$ pp on time-range, and $+1.3$ pp on text-heavy questions. This order is the reverse of the SFT accuracies in the same table, which is what the EXP3 update (Eqs. (1)–(3)) produces, since arms with lower accuracy receive more sampling weight. Figure 3(b) shows the same picture over training: the `graph-aug` and `visual-aug` arms rise well above uniform in the early periods while `text-raw` stays at or below it. Time-range questions gain less. Their retrieval action is a windowed scan over $\mathcal{M}_{\text{visual}}$ rather than a separate store, and their arms rise only mildly.

4.4 ABLATION STUDIES

VIDEOAGENT memory and planner (Table 3). Each memory store contributes, with a distinct profile. Removing text memory produces the largest drop (-5.6pp Avg), consistent with its role as default for text-heavy questions, the most common type. Graph memory removal disproportionately hurts LVBench (-3.5pp vs -1.8pp on EgoLifeQA), reflecting LVBench's narrative structure. Collapsing the multi-turn planner into single-shot retrieval reduces average performance by 4.7 pp. This drop highlights the benefit of composing memories across turns.

VIDEOGYM components (Table 4). Table 4 removes one component at a time on all five benchmarks. Replacing EXP3 with uniform sampling over the same arms costs 2.6 pp, the largest drop, so most of the gain comes from where the budget is spent. Removing augmentation costs 1.4 pp. Once the sampler shifts toward the rare modalities, the same raw questions are drawn again and again, and the generated variants give those arms new material. Removing the verifier costs 0.5 pp, as unfiltered rewrites add noise to the per-arm signal.

4.5 CURRICULUM DYNAMICS, DATA EFFICIENCY, AND GRPO ADVANTAGES

Arm weight dynamics. Figure 3(b) shows weak arms (`graph-aug`, `visual-aug`) upweighted substantially above uniform within the first $\sim$20 curriculum periods, then reverting toward uniform after period $\sim$30. The curriculum thus behaves as adaptive rather than static oversampling: it raises the probability of an arm while that arm lags and retracts it as the arm catches up.

Data efficiency. Figure 3(c) shows VIDEOGYM attaining the accuracy of the 2000-question no-curriculum baseline with about 600 unique raw questions at the same rollout budget ($\sim 3.3\times$ sample efficiency). Guo et al. (2026) report an analogous improvement for text-agent curricula.

GRPO advantages. Under a binary reward, a GRPO group whose trajectories share the same correctness outcome yields zero advantage. The composite reward separates such trajectories by evidence use and efficiency. Figure 3(d) compares the advantage standard deviation under the two

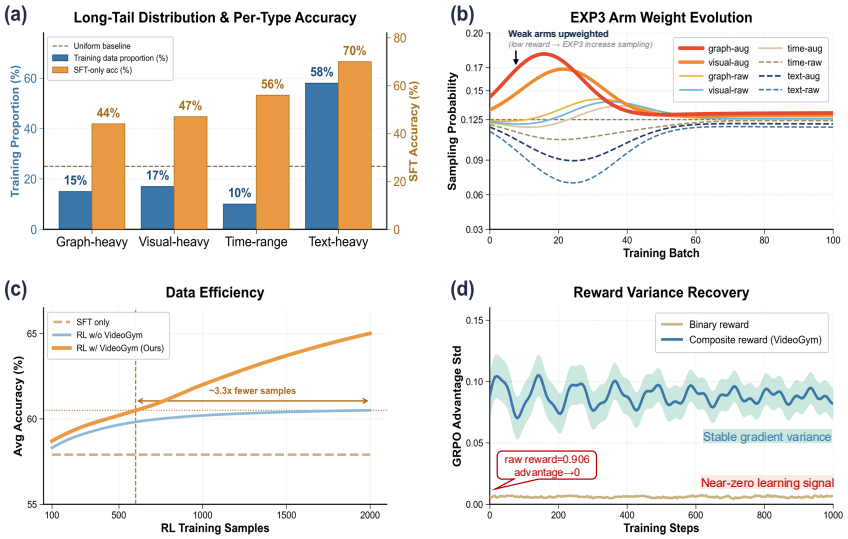

Figure 3: **VIDEOGYM training dynamics.** **(a)** Source-pool distribution and SFT accuracy by question type. **(b)** EXP3 arm probabilities over training. **(c)** Accuracy versus unique raw questions at the same rollout budget. **(d)** GRPO advantage standard deviation under binary and composite rewards.

rewards. Appendix F presents one representative trajectory per memory store and the main failure modes.

5 CONCLUSION

We introduced VIDEOGYM, a curriculum-based training environment that treats the allocation of RL training across retrieval modalities as an adversarial bandit. VIDEOGYM couples the EXP3 update with source-grounded augmentation and verifier filtering to correct the long-tail distribution under which fixed-distribution RL under-trains the rare retrieval modalities. Across five long-video benchmarks, VIDEOGYM lifts RL accuracy from 60.4% to 64.8% on average, with the largest gains on graph-heavy (+10.5 pp) and visual-heavy (+7.5 pp) questions, and reaches the 2000-question no-curriculum baseline with about 3.3× fewer unique raw questions at the same rollout budget.

Limitations. The pipeline relies on three offline-built memory indices whose quality bounds downstream retrieval, and on English-language ASR and captioning. The curriculum operates over a fixed arm partition rather than a learned one. Detailed treatment of these constraints, together with broader-impact considerations, is given in Appendix G.

AI USE STATEMENT

Generative models are used in our research pipeline to produce video captions for memory construction, generate teacher trajectories for supervised fine-tuning, and generate and filter augmented training questions through TASKGENERATOR and VERIFIER. These uses are described in the method and implementation sections and in the prompt templates (Appendices A and C). Generative AI tools were also used to assist with manuscript writing and editing. The authors are responsible for the final text, methodology, results, references, and all other artifacts, including material prepared with AI assistance.

REPRODUCIBILITY STATEMENT

The method and experimental setup are described in Sections 3 and 4.1. Additional implementation details, hyperparameters, and prompts are provided in Appendices A–C. The training code, configuration files, and prompt templates are provided as supplementary material.

REFERENCES

Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *Proceedings of the 25th Annual Conference on Learning Theory*, volume 23 of *Proceedings of Machine Learning Research*, pp. 39.1–39.26. PMLR, 2012.

Jean-Yves Audibert and Sébastien Bubeck. Minimax policies for adversarial and stochastic bandits. In *Proceedings of the 22nd Annual Conference on Learning Theory*, pp. 217–226, 2009.

Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2–3):235–256, 2002a. doi: 10.1023/A:1013689704352.

Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2002b. doi: 10.1137/S0097539701398375.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24108–24118, 2025.

Aurélien Garivier and Olivier Cappé. The kl-ucb algorithm for bounded stochastic bandits and beyond. In *Proceedings of the 24th Annual Conference on Learning Theory*, volume 19 of *Proceedings of Machine Learning Research*, pp. 359–376. PMLR, 2011.

Aurélien Garivier and Eric Moulines. On upper-confidence bound policies for switching bandit problems. In *Algorithmic Learning Theory*, volume 6925 of *Lecture Notes in Computer Science*, pp. 174–188. Springer, 2011. doi: 10.1007/978-3-642-24412-4_16.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15180–15190, 2023.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, 2025a. doi: 10.1038/s41586-025-09422-z. URL https://doi.org/10.1038/s41586-025-09422-z.

Jiacheng Guo, Ling Yang, Peter Chen, Qixin Xiao, Yinjie Wang, Xinzhe Juan, Jiahao Qiu, Ke Shen, and Mengdi Wang. GenEnv: Difficulty-aligned co-evolution between LLM agents and environment simulators. In *Advances in Neural Information Processing Systems*, 2026. URL https://neurips.cc/virtual/2026/poster/152039. To appear.

Zirui Guo, Lianghao Xia, Yanhua Yu, Tu Ao, and Chao Huang. LightRAG: Simple and fast retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 10746–10761, 2025b. doi: 10.18653/v1/2025.findings-emnlp.568. URL https://aclanthology.org/2025.findings-emnlp.568/.

Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. HippoRAG: Neurobiologically inspired long-term memory for large language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 59532–59569, 2024. doi: 10.52202/079017-1902. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/6ddc001d07ca4f319af96a3024f6dbd1-Paper-Conference.pdf.

Yuchen Huang, Sijia Li, Minghao Liu, Wei Liu, Shijue Huang, Zhiyuan Fan, Hou Pong Chan, and Yi R Fung. Environment scaling for interactive agentic experience collection: A survey. *arXiv preprint arXiv:2511.09586*, 2025.

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*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/b14d7175755b180dc2163e15e3110cb6-Paper-Conference.pdf.

Yueqian Lin, Jingyang Zhang, Qinsi Wang, Hancheng Ye, Yuzhe Fu, Yudong Liu, Hai Helen Li, and Yiran Chen. HippoMM: Hippocampal-inspired multimodal memory for long audiovisual event understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pp. 5968–5977, 2026. URL https://openaccess.thecvf.com/content/CVPR2026F/html/Lin_HippoMM_Hippocampal-inspired_Multimodal_Memory_for_Long_Audiovisual_Event_Understanding_CVPRF_2026_paper.html.

Lin Long, Yichen He, Wentao Ye, Yiyuan Pan, Yuan Lin, Hang Li, Junbo Zhao, and Wei Li. Seeing, listening, remembering, and reasoning: A multimodal agent with long-term memory. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/ec370e882e8a02be955679ddbd2731b8-Paper-Conference.pdf.

Siyuan Lu, Zechuan Wang, Hongxuan Zhang, Qintong Wu, Leilei Gan, Chenyi Zhuang, Jinjie Gu, and Tao Lin. Don't just fine-tune the agent, tune the environment. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://iclr.cc/virtual/2026/poster/10007443.

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. In *Advances in Neural Information Processing Systems*, volume 38, pp. 168008–168033, 2025. doi: 10.52202/085713-5602. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/f593c9c251d4d7cf14d4ab9861dfb7eb-Paper-Conference.pdf.

Kun Ouyang, Yuanxin Liu, Linli Yao, Yishuo Cai, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. Conan: Progressive learning to reason like a detective over multi-scale visual evidence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 41089–41099, 2026.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pp. 28492–28518. PMLR, 2023.

Xubin Ren, Lingrui Xu, Long Xia, Shuaiqiang Wang, Dawei Yin, and Chao Huang. VideoRAG: Retrieval-augmented generation with extreme long-context videos. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1*, pp. 2390–2401, 2026. doi: 10.1145/3770854.3783944. URL http://dx.doi.org/10.1145/3770854.3783944.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y. Wu, et al. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Shulin Tian, Ruiqi Wang, Hongming Guo, Penghao Wu, Yuhao Dong, Xiuying Wang, Jingkang Yang, Hao Zhang, Hongyuan Zhu, and Ziwei Liu. Ego-R1: Agentic chain-of-tool-thought for ultra-long egocentric video reasoning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(10):12116–12131, 2026. doi: 10.1109/tpami.2026.3697367. URL https://doi.org/10.1109/TPAMI.2026.3697367.

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 *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22958–22967, 2025a. URL https://openaccess.thecvf.com/content/ICCV2025/html/Wang_LVBench_An_Extreme_Long_Video_Understanding_Benchmark_ICCV_2025_paper.html.

Ye Wang, Ziheng Wang, Boshen Xu, Yang Du, Kejun Lin, Zihan Xiao, Zihao Yue, Jianzhong Ju, Liang Zhang, Dingyi Yang, et al. Time-R1: Post-training large vision language model for temporal video grounding. In *Advances in Neural Information Processing Systems*, volume 38, pp. 83330–83364, 2025b. doi: 10.52202/085713-2793. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/7801b29c93b599b8d0c44138596bdeed-Paper-Conference.pdf.

Ziyang Wang, Jaehong Yoon, Shoubin Yu, Md Mohaiminul Islam, Gedas Bertasius, and Mohit Bansal. Video-RTS: Rethinking reinforcement learning and test-time scaling for efficient and enhanced video reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 28126–28140, 2025c. doi: 10.18653/v1/2025.emnlp-main.1428. URL https://aclanthology.org/2025.emnlp-main.1428/.

Zhiheng Xi, Yiwen Ding, Wenxiang Chen, Boyang Hong, Honglin Guo, Junzhe Wang, Xin Guo, Dingwen Yang, Chenyang Liao, Wei He, et al. Agentgym: Evaluating and training large language model-based agents across diverse environments. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 27914–27961, 2025. doi: 10.18653/v1/2025.acl-long.1355.

Jingkang Yang, Shuai Liu, Hongming Guo, Yuhao Dong, Xiamengwei Zhang, Sicheng Zhang, Pengyun Wang, Zitang Zhou, Binzhu Xie, Ziyue Wang, et al. EgoLife: Towards egocentric life assistant. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28885–28900, 2025.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*, 2023.

Woongyeong Yeo, Kangsan Kim, Jaehong Yoon, and Sung Ju Hwang. WorldMM: Dynamic multimodal memory agent for long video reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25599–25609, 2026.

Zhiyuan Zeng, Hamish Ivison, Yiping Wang, Lifan Yuan, Shuyue Stella Li, Zhuorui Ye, Siting Li, Jacqueline He, Runlong Zhou, Tong Chen, et al. RLVE: Scaling up reinforcement learning for language models with adaptive verifiable environments. In *International Conference on Machine Learning (ICML)*, 2026. URL https://arxiv.org/abs/2511.07317.

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, pp. 89863–89895, 2025a. doi: 10.52202/085713-3005. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/8190b210e9808e54ee16263b673a847d-Paper-Conference.pdf.

Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. LLaVA-Video: Video instruction tuning with synthetic data. *Transactions on Machine Learning Research*, 2025b. URL https://openreview.net/forum?id=EElFGvt39K.

A Implementation Details

A.1 VideoAgent Memory Construction

Every video $\mathcal{V}$ is processed once offline and cached in a per-video working directory that is reused across training, augmentation, and evaluation. Keyframe extraction combines uniform sampling with shot-boundary detection so that the planner has both regularly spaced anchors and scene-transition cues for its first-turn global perception. The video is then split into fixed-duration segments, and segment-level transcripts are produced by Whisper (Radford et al., 2023). Per-segment captions come from Qwen3-VL, and segment-level visual embeddings come from ImageBind (Girdhar et al., 2023).

Three indices are constructed from these intermediate artefacts. $\mathcal{M}_{\text{text}}$ is a dense chunk index built jointly over captions and ASR transcripts, retrievable by semantic similarity. $\mathcal{M}_{\text{graph}}$ extracts caption-driven entities and relations and links them across segments to expose multi-hop structure that no single segment encodes. $\mathcal{M}_{\text{visual}}$ stores segment-level visual embeddings together with timestamp metadata, supporting both content-based queries and the windowed scans used by `search_time_range`.

Indexing is performed once per video and amortized across all downstream uses. During training and evaluation, subsequent operations access only the constructed indices rather than re-reading the raw video file, making per-turn latency primarily governed by LLM decoding rather than memory I/O.

A.2 SFT and RL Data

Source corpus. Both SFT and RL training data are derived from **LLaVA-Video-178K** (Zhang et al., 2025b), an open instruction-tuning corpus that aggregates roughly 178K source videos and 1.3M instruction-response samples (178K detailed captions, 961K open-ended QA, 196K four-option multiple-choice QA). Source videos are pooled from ActivityNet, Charades, NExT-QA, EgoSchema, and PerceptionTest. We use the multiple-choice portion of the corpus, where four-option grounding is directly compatible with answer-option matching and sidesteps the open-ended scoring drift that makes captioning samples unreliable as RL reward signals. The indexing requirement of Appendix A.1, namely that every retained sample carry a fully built three-modality memory, then narrows the candidate pool before teacher tracing.

Question-type and difficulty annotation. Every retained sample is assigned a question type in $\mathcal{C}$ (Section 3.1) by a rule-based classifier that fires in priority order. Strong textual cues such as on-screen text, caption, subtitle, dialogue, or speech mentions map to *text*. Causal, relational, or summarization cues such as "why", "purpose", "best summarizes" map to *graph*. Explicit timestamps or sequential-event language such as "before", "after", "first scene" map to *time-range*. Remaining samples default to *visual*. Items that fire no rule confidently are spot-checked manually and dropped. A scalar difficulty score is then accumulated from question-side signals ($+2$ for graph cues, $+1$ each for time-range, text, counting, negation, multi-constraint phrasing, or Jaccard similarity above 0.5 between the four answer options) and from teacher-trace statistics on accepted trajectories (additional points for higher retrieval-turn count, modality diversity, and segment coverage). Samples are bucketed as *easy* (≤ 1), *medium* (2–3), or *hard* (≥ 4). Difficulty is recorded for stratification but is not used as an arm dimension. The curriculum operates only over $\mathcal{C} \times \{\textit{aug}, \textit{raw}\}$.

Long-tail mixture. The annotated source question pool exhibits the long-tail modality skew reported in §4.1: *text* $\approx 58\%$, *visual* $\approx 17\%$, *graph* $\approx 15\%$, and *time-range* $\approx 10\%$. This is the imbalance that the EXP3 curriculum redirects online.

Teacher trace collection. For every retained prompt we run a GPT-4o-backed teacher agent through the full VideoAgent multi-turn interface with four keyframes per turn and at most $T_{\max}=10$ turns. The teacher generates a trajectory consisting of a sequence of $\langle \text{think}, \text{action} \rangle$ pairs followed by a final answer. We retain only trajectories in which the final choice matches the ground-truth label and the action sequence satisfies a structural quality audit. The audit rejects malformed action tags, repeated-search loops past the dedup guard, and forced-termination at $T_{\max}$ without a confident answer. The accepted set comprises roughly 96K trajectories. The remaining rejections dominantly come from forced-termination on hard samples and from semantically valid but format-noncompliant teacher outputs, which we found cheaper to discard than to repair.

Two-stage SFT decomposition. Each accepted trajectory yields two disjoint supervised samples. The *Planner* stage extracts all intermediate $\langle$think, retrieval-action$\rangle$ pairs and trains the model to predict the next action conditioned on the partial trajectory, with the final answer turn masked out. The *Answer* stage retains only the last turn and supervises the answer conditioned on the accumulated evidence. This decomposition forces the planner to learn what to retrieve next without conflating it with what to answer, so that the SFT model does not start RL by answering from parametric knowledge alone. Both stages are emitted in ShareGPT-style multimodal format and consumed by a standard instruction-tuning pipeline.

RL data layer. A random subset of $\sim$2K of the accepted prompts is serialized in a unified *prompt+label* schema with fields $\{$*video_path*, *ground_truth*, *mode*, *question_type*, *difficulty*$\}$ and converted to a Parquet shard for the RL rollout engine. Because the SFT and RL banks share both prompts and annotations, the two phases draw from the same source pool and the same per-sample metadata, which is what makes per-arm bandit accounting consistent across SFT-init and the subsequent RL updates.

A.3 RL Training Infrastructure

The RL setup couples a multi-turn rollout engine with a curriculum controller through a narrow interface. On the rollout side, an environment wrapper turns planner outputs into environment steps. For each retrieval action it issues a request to the search service of Appendix A.4, validates the response, and maintains guards for duplicate searches, blocked action tags, and forced-answer at $T_{\max}$=10. A rollout-and-reward driver orchestrates per-group rollouts at GRPO group size G=8 and computes advantages from the composite reward of Eq. (4).

On the curriculum side, a single detached process owns all bandit state. Before each question group the controller is queried for the arm $z = (c, m)$ and, if m=*aug*, dispatches TaskGenerator and Verifier (Appendix C). After each rollout, the binary correctness ρ is written back to the arm's period statistics. After $BG = 1024$ trajectories, the controller triggers the EXP3 update of Eqs. (1)–(3), producing the next-period sampling distribution and clearing the period statistics. The arm-accounting interface only requires a scalar outcome per rollout, so the RL algorithm and the curriculum are independently swappable.

All training and evaluation runs use a single 8×H100-80G node. Video memory indexing (Appendix A.1) is a separate offline preprocessing step that is not included in the training compute budget. Once built, the same indices are reused across SFT, RL, and evaluation. RL rollouts dominate the recurring wall-clock. EXP3 updates are $O(K)$=$O(8)$ per curriculum period and negligible.

A.4 Search API and Serving Layer

VideoAgent's retrieval functionality is exposed as an HTTP service so that rollouts, teacher-trace collection, and evaluation share the same execution path. The service exposes three endpoints. A single-shot question-to-answer endpoint runs the full planner loop in one call and returns only a final answer, which is what evaluation uses. A one-step retrieval endpoint executes a single *search* action and returns its evidence, which is what RL rollouts consume turn by turn. A multi-turn agent-search endpoint runs the planner loop while exposing every intermediate think/action/observation, which is what teacher-trace collection records. All three endpoints route a $(\textit{video_path}, \textit{query})$ pair to the pre-built per-video working directory of Appendix A.1 and cache one VideoAgent instance per video so that index loading is not paid on every request.

Decoupling retrieval from the RL framework in this way has two benefits. First, it lets rollouts and evaluation run on the same index without code duplication, which removes a class of train/eval drift that we observed in earlier prototypes that re-implemented retrieval inside the rollout loop. Second, it confines the augmentation rewriter and the verifier to the curriculum side. They never need to instantiate a video index themselves, which keeps the augmentation budget bounded by prompt tokens rather than index construction time.

B VIDEOGYM ALGORITHM DETAILS

B.1 EXP3 SAMPLER: EXTENDED PSEUDOCODE

Algorithm 1 in the main text gives the concise form. The full implementation maintains three quantities per arm: an unnormalized weight w_z, a period-average correctness $\bar{\rho}_z$, and a trajectory count N_z. At the start of training all weights are initialized to $w_z^{(0)}=1$, producing the uniform sampling distribution $p_z^{(0)}=1/K$. During each curriculum period t:

1. Draw $B = 128$ arms $\{z_i\}$ i.i.d. from $p^{(t)}$ and sample one question with replacement from each selected type. Keep $p^{(t)}$ fixed throughout the period.
2. For each sampled arm z_i, assemble the sample (raw or TASKGENERATOR→VERIFIER), collect $G = 8$ trajectories, and record their binary outcomes. Rejected rewrites fall back to the source question, with feedback attributed to its raw arm.
3. After collecting $BG = 1024$ trajectories, compute $\bar{\rho}_z^{(t)}$ as the mean correctness of the trajectories attributed to each observed arm z.
4. Apply the learnability cap $\ell_z^{(t)}=\min(1-\bar{\rho}_z^{(t)},\delta)$ *before* forming the importance-weighted signal $\tilde{\ell}_z^{(t)}=\ell_z^{(t)}/p_z^{(t)}$ (Eq. 1).
5. Update observed-arm weights $w_z^{(t+1)}=w_z^{(t)}\exp(\gamma\,\tilde{\ell}_z^{(t)}/K)$, leave unobserved-arm weights unchanged, and re-normalize into $p^{(t+1)}$ via Eq. (3). Clear the period's correctness statistics and counts.

The cap is applied to the error signal before importance weighting, consistent with Eq. (1).

B.2 HYPERPARAMETER SPECIFICATION

Symbol	Value	Role
EXP3 *curriculum*		
K	8	arms $= \mathcal{C} \times \{aug, raw\}$
γ	0.1	exploration rate (Eq. 3)
δ	0.80	learnability cap (Eq. 1)
question draws B	128	sampled per curriculum period
update period	$BG = 1024$	trajectories per EXP3 update
Composite reward		
λ_e	0.3	evidence-utility weight
λ_{eff}	0.3	efficiency-cost weight
λ_{lazy}	0.2	lazy-shortcut penalty (correct$\wedge T_{\text{search}}$=0)
$(\alpha_1,\alpha_2,\alpha_3,\alpha_4)$	$(0.2, 0.4, 0.2, 0.2)$	turn / forced / blocked / retry weights in c_{eff} (α_2 highest)
RL / rollout		
GRPO group size G	8	rollouts per query
max turns $T_{\max}$	10	forced answer beyond this cap
SFT		
epochs	2	each of Stage 1 / Stage 2
backbone	Qwen3-VL-8B	shared across SFT and RL (matches §4.1)

B.3 LEARNABILITY CAP: MOTIVATION

The cap clips the period-wise error signal before importance weighting: $\ell_z^{(t)} = \min(1-\bar{\rho}_z^{(t)},\delta)$. With $\delta = 0.80$, clipping affects arms whose accuracy in the current period is below 0.20, and bounds the weight growth of an arm on which the policy currently fails almost entirely.

B.4 COMPOSITE REWARD: COMPONENTS

Equation (4) decomposes the reward into three additive components plus a lazy-shortcut penalty. We restate their definitions here. Correctness $\rho \in \{0, 1\}$ is determined by matching the predicted option to the ground-truth option. Evidence utility $b_{\text{evid}} \in [0, 1]$ measures the fraction of retrieval turns

whose returned snippet contains information that supports the ground-truth answer. We compute it using evidence matching against the reference answer, and apply it only to correct episodes so that lucky guesses do not receive evidence credit. Efficiency cost $c_{\text{eff}} \in [0, 1]$ is a weighted sum of normalized turn count, the forced-termination indicator, the blocked-tag ratio, and the format-retry ratio, with weights $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)=(0.2, 0.4, 0.2, 0.2)$ that put the largest penalty on forced termination. The lazy-shortcut penalty $\lambda_{\text{lazy}}\,\mathbf{1}[T_{\text{search}}=0]$ lowers the reward of correct answers produced without retrieval.

The evidence and efficiency terms can distinguish trajectories with the same correctness outcome, providing variation in the rewards used by GRPO.

Decoupled use in EXP3 and GRPO. The composite reward $R(\tau)$ is consumed only by the GRPO policy update. The EXP3 curriculum update consumes the binary correctness ρ, not $R(\tau)$. The curriculum therefore reads final-answer correctness, while the policy gradient reads the composite signal that keeps within-group advantages from collapsing.

B.5 Computational Cost

The EXP3 update itself is $O(K)=O(8)$ per curriculum period and accounts for a negligible fraction of wall-clock. The measurable cost of VideoGym relative to plain RL comes from the augmentation path. Selecting arm *aug* requires one LLM call from TaskGenerator and one from Verifier. Rejected samples fall back to the raw arm and are used for rollout, with the rejection cost paid in prompt tokens rather than rollout tokens. In practice the curriculum adds about one hour of training wall-clock (Appendix E.1).

C Prompt Templates

We specify each prompt as a compact schema with input, output, and operative rules.

C.1 Planner

Input question, keyframe panel, accumulated evidence, turn index k.
Output `<think>...</think>` followed by exactly one action tag.
Action grammar `<search_text>`, `<search_graph>`, `<search_visual>`, `<search_time_range>`, `<answer>`.
Rules (i) emit exactly one tag per turn; (ii) duplicate `<search_*>` arguments across turns are blocked by the dedup guard; (iii) if $k{=}T_{\max}$ and no `<answer>` has been emitted, the environment forces one from the current evidence window.

C.2 TaskGenerator (Augmentation Rewriter)

Both TaskGenerator and Verifier use GPT-4o.

Input source question x, its question type c, video captions and entity-relation triples of $\mathcal{V}$.
Output rewritten question $\tilde{x}$ with the same question type c, four options, and one correct option.
Rules (i) the answer to $\tilde{x}$ must exist in $\mathcal{V}$ and be obtainable via the modality corresponding to c; (ii) the rewrite should increase difficulty by adding a multi-hop step, finer temporal granularity, or a rarer entity, without changing the underlying topic; (iii) the output is the question, four options, and correct option only, with no additional commentary.

C.3 Verifier (Quality Gate)

Input candidate question $\tilde{x}$, type c, four options, and correct option, video captions and entity-relation triples of $\mathcal{V}$, existing training bank.
Output ACCEPT or REJECT with a one-line reason code.
Rubric (i) *well-formed*: the question is syntactically valid and unambiguous; (ii) *answerable*: the answer span is supported by $\mathcal{V}$ via the modality c; (iii) *non-trivial*: not a paraphrase duplicate of any existing item in the training bank (string and embedding similarity thresholds).
Fallback rejected samples are returned to the raw arm of $\mathcal{A}_{\text{gym}}$ (Alg. 1), keeping the rollout budget fixed.

C.4 Answer Grading

Answers are scored by exact matching of the predicted option to the ground-truth option.

D Dataset and Benchmark Details

D.1 Training Corpus Composition

The training bank shares its prompt pool with the SFT corpus described in Appendix A.2. The RL rollout engine draws a random $\sim$2K subset of the accepted prompts that supply the two-stage SFT samples, with each prompt serialized as a *prompt+label* record carrying $\{$*video_path*, *ground_truth*, *mode*, *question_type*, *difficulty*$\}$. A single sample therefore flows through SFT data preparation, RL rollouts, augmentation rewriting, and evaluation without ever changing schema, which is the property the per-arm accounting in §3.3 relies on.

The distribution in the source question pool over $\mathcal{C}$ is *text*$\approx$58%, *visual*$\approx$17%, *graph*$\approx$15%, and *time-range*$\approx$10%. The long-tail of text-heavy questions is the distributional motivation for EXP3.

D.2 Question-Type Taxonomy

The four types in $\mathcal{C}$ partition questions by the retrieval modality that is most directly evidence-providing.

Type	Primary modality	Example
text-heavy	$\mathcal{M}_{\text{text}}$	"What was the speaker's main argument about X?"
graph-heavy	$\mathcal{M}_{\text{graph}}$	"Who attended every meeting that A attended?"
visual-heavy	$\mathcal{M}_{\text{visual}}$	"What object is the child holding at the party?"
time-range	$\mathcal{M}_{\text{visual}}$ (time-gated)	"What happens between 22:10 and 22:30?"

time-range reuses $\mathcal{M}_{\text{visual}}$ rather than defining a separate memory. The distinction lives in the retrieval action, not in the stored representation (§3.2). Type annotation follows the rule-based scheme described in Appendix A.2, with manual spot-checks resolving ambiguity. Items where no rule fires confidently ($<3\%$ of the pool) are excluded from per-arm accounting so that bandit reward estimates are not corrupted by mislabeling at the arm boundary.

D.3 Long-Tail Distribution

The proportions in Fig. 3(a) are those of the source question pool, and the $\sim$2K RL prompts are drawn from it at the same proportions. Reweighting alone therefore cycles quickly through the same graph-heavy and time-range prompts. The augmentation arm supplies these arms with harder verified variants.

D.4 Augmentation Acceptance Rate per Arm

Verifier pass rates are tracked per arm throughout training and serve both as a data-quality monitor and as a diagnostic for TaskGenerator. Rejection reasons fall into three buckets. *Malformed* candidates violate the output schema or reference entities absent from $\mathcal{V}$. *Unsupported* candidates have an answer span that cannot be recovered via the required modality. *Duplicate* candidates fail a string-plus-embedding similarity check against the training bank.

D.5 Evaluation Benchmarks

We evaluate on five long-video benchmarks that collectively span the four retrieval demands in $\mathcal{C}$:

- **EgoLifeQA** (Yang et al., 2025): egocentric multi-day QA over household-scale video streams. Stresses long-range graph-relational reasoning about people and events.
- **Ego-R1 Bench** (Tian et al., 2026): multi-hop reasoning over first-person video, with questions requiring composition across several episodes.

- **HippoVlog** (Lin et al., 2026): character and event tracking over hour-long vlogs.
- **LVBench** (Wang et al., 2025a): long-form narrative QA, up to 4 h. Dominated by text-heavy and visual-heavy items.
- **Video-MME (Long)** (Fu et al., 2025): diverse multi-modal QA over long clips, used as a broad-coverage reference.

Answers are scored by matching the predicted option to the ground-truth option. Table 3 averages EgoLifeQA and LVBench, and Table 4 averages all five benchmarks.

E Additional Experimental Results

E.1 Training Compute and Wall-Clock

All training and evaluation use a single $8\times$H100-80G node. Video memory indexing (Appendix A.1) is a separate offline preprocessing step run once and reused across every downstream training, augmentation, and evaluation pass, so we exclude it from the budget below. Table 5 reports the recurring training cost from a fresh SFT initialization to the final RL w/ VideoGym checkpoint.

Table 5: **Recurring training compute on $8\times$H100-80G.** Wall-clock is the elapsed time of a single end-to-end run, and GPU-hours $= 8\times$ wall-clock. Token counts cover the dominant cost (training tokens for SFT; rollout+backward for RL). The VideoGym curriculum adds about one hour on top of RL w/o VideoGym. Offline video indexing is excluded as it is performed once and reused across all downstream runs.

Stage	Wall-clock (h)	GPU-hours	Tokens	Notes
SFT Stage 1 (Planner, 2 epochs)	5	40	0.60 B	intermediate (think, action) pairs
SFT Stage 2 (Answer, 2 epochs)	2	16	0.20 B	final-turn answer only
RL w/o VideoGym	14	112	0.22 B	GRPO, G=8, $T_{\max}$=10
RL w/ VideoGym (full)	15	120	0.23 B	+ TaskGenerator/Verifier
Recurring total (RL w/ Gym)	**22**	**176**	**1.03 B**	SFT_1+SFT_2+$\mathrm{RL}_{\mathrm{Gym}}$

End-to-end recurring training is therefore about 22 hours of wall-clock and 176 GPU-hours, of which the RL phase contributes 120 GPU-hours ($\approx 68\%$). The curriculum adds about one hour ($+8$ GPU-hours) over RL w/o VideoGym.

F Qualitative Case Studies

The traces below are sampled from the LLaVA-Video-178K multiple-choice training pool described in Appendix A.2. Each box shows a real VideoAgent trajectory at small scale, with $\langle$think, action$\rangle$ pairs preserved verbatim from the trace and only minor whitespace trimming. We pick one representative success per modality and one trajectory per failure mode from the same trace pool.

F.1 Visual-Heavy Trace

Q. *What specific action does Johnny Debut perform to motivate the participants?* **Options.** (A) he uses hand gestures (B) he dances (C) he claps his hands (D) he sings a song. **Ground truth.** (A).

T0 *think* The question asks about the specific motivating action performed by Johnny Debut, which requires understanding patterns of interaction or gestures. This demands the relationship between the person named Johnny and the participants, which I can probe with a graph search.
act `<search_graph>` "Johnny Debut motivating participants interaction actions"

T1 *think* The retrieved relation indicates that Johnny Debut motivates participants during cycling workout sessions, but the specifics of how (hand gestures, dancing, clapping, or singing) are not yet narrowed down. I need a visual search to identify the actual action.
act `<search_visual>` "Johnny Debut motivating participants: hand gestures, dancing, clapping, or singing"

T2 *think* The retrieved frames show Johnny Debut motivating participants by speaking animatedly and gesturing with his hands. The captions specifically mention that he "gestures animatedly while speaking to the camera," which matches option (A) hand gestures.
act `<answer>` (A) He uses hand gestures

The trace shows the canonical relational-then-visual pattern. The graph store fixes who Johnny is and what he is doing at the discourse level, after which a single visual call grounds the abstract action label in concrete body motion.

F.2 GRAPH-HEAVY TRACE

Q. *Why does the person use a spoon after pouring the blended mixture into the glass?* **Options.** (A) to clean the glass (B) to add more ingredients (C) to taste the mixture (D) to stir the mixture. **Ground truth.** (D).

T0 *think* The sequence of frames shows the preparation of a beverage using a blender, indicating practical use of tools. Since it is unclear from the images alone why a spoon is used specifically, I will search the textual store for descriptions of spoon usage during this step.
act `<search_text>` "spoon usage after pouring blended mixture into glass"

T1 *think* The captions describe the process of preparing a frozen beverage but do not explicitly state the spoon's purpose. I will probe the relational store for the entity-level relation between the spoon and the mixture pouring step.
act `<search_graph>` "purpose of spoon after pouring blended mixture into glass"

T2 *think* The graph returns entities and relationships related to drink preparation but no explicit action label for the spoon. The spoon could plausibly be for stirring or tasting; visual evidence will disambiguate.
act `<search_visual>` "spoon action after pouring mixture into glass"

T3 *think* The frames show the mixture has a slushie-like consistency that typically requires stirring for even distribution, and the spoon is used to mix the contents in the glass. Combined with the textual and relational context, this points to (D) stirring rather than tasting.
act `<answer>` (D) To stir the mixture

The trace exercises all three retrieval stores in three turns, with each turn closing one ambiguity left by the previous one. Text describes *what is happening*, graph encodes the *relation* between the spoon and the mixture, and visual confirms the actual *motion*.

F.3 TEXT-HEAVY TRACE

Q. *What text appears on the right side of the split-screen?* **Options.** (A) "MAKEUP TRANSFORMATION IN PROGRESS" (B) "WATCH THE MAGIC HAPPEN" (C) "WE ASKED A PROFESSIONAL MAKEUP ARTIST TO TRANSFORM THESE FIVE MEN…" (D) "BEAUTY MAKEOVER". **Ground truth.** (C).

T0 *think* The question involves on-screen text in the video, which is best handled by a textual search over captions and ASR transcripts.
act `<search_text>` "MAKEUP TRANSFORMATION IN PROGRESS, WATCH THE MAGIC HAPPEN, WE ASKED A PROFESSIONAL MAKEUP ARTIST"

T1 *think* The textual store does not surface the specific overlay on the right side of the split-screen. On-screen graphics are often absent from caption and transcript pipelines, so a direct visual search of the relevant frames is the natural fallback.
act `<search_visual>` "text on right side of split-screen: MAKEUP TRANSFORMATION, WATCH THE MAGIC HAPPEN, WE ASKED A PROFESSIONAL MAKEUP ARTIST"

T2 *think* The retrieved frames show that the text on the right side reads "WE ASKED A PROFESSIONAL MAKEUP ARTIST TO TRANSFORM THESE FIVE MEN…" which matches option (C) exactly.
act `<answer>` C

A common failure mode of caption- and ASR-based stores is that on-screen overlays produced post-hoc in editing are not reflected in either the spoken track or the dense captions. The planner correctly diagnoses this gap and falls back to the visual store, which inspects the rendered frames directly. The case illustrates why text-heavy questions still benefit from visual augmentation.

F.4 FAILURE MODES

Curriculum training reduces but does not eliminate four recurring failure modes. We sample one trajectory from each.

Hallucinated retrieval. On a graph-heavy question about a participant's role in a panel, the planner emits a `search_graph` query naming an entity that does not appear in $\mathcal{V}$. The graph store returns no hits, and the planner treats the absence as evidence for the most plausible-sounding distractor rather than retrying with a different entity span.

Early-stop. On a four-option visual identification question, the planner emits `answer` after a single `search_text` call whose evidence is consistent with two of the four options.

Repeated search. On a hard time-range question, the planner issues two `search_time_range` queries with overlapping windows. The dedup guard blocks the second query and the planner has no remaining turns to pivot to a different modality, so it forces an answer from incomplete evidence.

Verifier false-accept. A TASKGENERATOR rewrite that is grammatically valid but answer-shifted (the rewriter has subtly changed the referent of the pronoun) passes the VERIFIER non-trivial check and enters the training pool. Per-arm accuracy in this period dips, and EXP3 responds by upweighting the same arm.

G DISCUSSION

G.1 LIMITATIONS

Dependence on offline index quality. The three memory indices are constructed once and reused. Errors in ASR, captioning, or entity extraction are not corrected at train or inference time, so downstream retrieval inherits an upper bound from the index. Online re-indexing triggered by repeated retrieval failure is a natural follow-up but is outside the scope of this work.

English-centric transcripts. Our experiments use English-language videos and questions. Results on non-English video content are not reported and should not be extrapolated from our numbers without re-evaluating the index pipeline.

Fixed arm space. $\mathcal{A}_{\text{gym}}=\mathcal{C}\times\{aug, raw\}$ discretizes the curriculum along pre-defined question types. Learning the arm partition itself, rather than fixing it, would remove this assumption but introduces identifiability issues we have not resolved.

G.2 BROADER IMPACT

Privacy of long-form video. Memory-grounded video agents, by construction, index and persistently store content from long recordings including egocentric footage. Deployments must honor the consent scope of the original recording and constrain retrieval to that scope. None of the methods in this paper change the underlying privacy surface of the video data.

Compute cost. Offline video index construction is the dominant one-time preprocessing cost. Once built, subsequent training and evaluation reuse the same indices and never re-touch the raw video. Reported recurring training runs fit on a single $8\times$H100-80G node in roughly 22 wall-clock hours per backbone, with full per-stage breakdown in Appendix E.1.

Misuse. The same retrieval-and-reasoning capability that enables useful question answering over long videos also enables surveillance-style queries. We do not release pretrained indices on third-party videos.

G.3 RELATION TO PREFERENCE-BASED TRAJECTORY OPTIMIZATION

Our contribution is orthogonal to that of Conan, which optimizes the video agent with RLVR over full trajectories. EXP3 operates one level above the trajectory optimizer, reshaping the *mixture* of trajectories that the optimizer sees. Concretely, a DPO-based trajectory learner could be substituted into our RL step without changing the VIDEOGYM interface, which only requires a scalar outcome per rollout. Combining the two is a natural direction for future work.

G.4 EXTENSIBILITY BEYOND VIDEO

The decomposition $\mathcal{A}_{\text{gym}}=\mathcal{C}\times\{aug, raw\}$ is not specific to video. It applies whenever an agent's action space decomposes into (i) discrete retrieval modalities over a structured memory and (ii) a controllable augmentation dimension. Candidate settings include: (a) web agents with text / graph / code / table retrieval over a document corpus; (b) embodied agents with semantic-map / metric-map / affordance memories; (c) scientific agents with literature / database / code-execution retrieval. In each case, TASKGENERATOR and VERIFIER would need to be reinstantiated around the domain's content representations, but the curriculum machinery could be evaluated in these settings.