EGSD: EVENT-GROUNDED SELF-DISTILLATION FOR STREAMING VIDEO UNDERSTANDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Real-time video understanding requires incrementally maintaining a memory of streaming content, and optimizing this requires dense process signals. On-Policy Self-Distillation (OPSD), which lets one model serve as both teacher and student with the teacher receiving additional privileged information such as the question and ground-truth (GT) answer, can supply such token-level signals. However, applying it directly to streaming video raises two problems. (1) The student cannot be optimized end-to-end, where memory is written before the question arrives, yet the teacher scores it with the question-and-GT privilege, misaligning their preferences. (2) Effective-entity memory collapses, where the question-and-GT privilege makes the teacher favor only question-relevant entities, and token-mean averaging over a memory renders its signal invariant to how many entities that memory covers, both driving memory against the streaming need for diversity. To address these issues, we propose **Event-Grounded Self-Distillation (EGSD)**, which characterizes streaming memory as an incremental update over verifiable Events (key visual entities, actions, and details) and targets the two problems on this basis. For problem (1), we adapt the OPSD signal into a multiplicative weight combined with the outcome reward; for problem (2), we re-weight the teacher with Events as privileged information to counter its question-relevance bias, and add an entity-coverage reward to supply the coverage preference the token-mean teacher lacks. Extensive experiments on mainstream online and offline benchmarks show EGSD achieves strong performance, reaching 79.8% on Streaming-Bench and 73.4% on the OVO-Bench Real-Time track, while memory analysis shows effective-entity recall rises 17.4% at only 6.8% more memory length.

1 INTRODUCTION

Video Large Language Models (VideoLLMs) have achieved significant success in offline video understanding (Chen et al., 2025c; Wang et al., 2025a; Qin et al., 2025). However, real-world applications such as live-stream QA and embodied intelligence (Chen et al., 2024; 2025a; Driess et al., 2023) demand real-time processing of streaming input, bound by temporal causality and a limited context window. This raises a central challenge, incrementally memorizing long-horizon visual context while producing causal responses in an online manner (Lin et al., 2026).

Such incremental memory of the streaming history admits different representations. One way compresses visual information at the token or KV-cache level or into hidden states, and is mainly optimized by training-free or lightweight-adapter inference that prunes visual tokens by inter-frame similarity, attention importance, and similar criteria (Kim et al., 2026; Xie et al., 2026b; Xu et al., 2026); yet such memory is not human-readable, so whether it retains the information a future query needs cannot be measured explicitly. The other way writes the history as natural-language text, which is reviewable and is the memory this work adopts. Existing methods optimize such memory with an SFT-RL pipeline (Guan et al., 2026; Xie et al., 2026a), but its reward is verifiable only at the end of a long trajectory, leaving the model passively driven by final-answer correctness.

On-Policy Distillation (OPD) (Gu et al., 2023; Agarwal et al., 2024) is an important post-training paradigm that supplies dense, token-level process signals within a trajectory. Vanilla OPD needs a larger same-family teacher, bounding gains by the teacher's capability at high cost; On-Policy Self-Distillation (OPSD) (Zhao et al., 2026) instead lets a student holding privileged information act as

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.

its own teacher, needing no larger model and balancing training cost against performance. Recent work explores this paradigm across image-text and mathematical reasoning with RLSD (Yang et al., 2026), multi-turn agents with StepOPSD (Zhang et al., 2026), multimodal reasoning with OPLD (Zhu et al., 2026), and privileged visual evidence with Video-OPSD (Wang et al., 2026), differing in what privileged information the teacher sees and how the resulting signal re-weights tokens; among these, the mainstream privileged information is the final question and the ground-truth answer.

However, these OPSD methods mostly target single-turn or offline settings, whereas streaming video understanding with natural-language memory has two distinctive traits. First, the intermediate memory turns run before any question, so the model can only record the salient visual content it currently observes. Second, for low-latency answering the model must reply from its existing textual memory and the current video clip alone, so the diversity of effective visual entities in memory is critical. Directly applying existing OPSD methods to streaming video thus raises two problems. (1) **The student cannot be optimized end-to-end.** In single-turn QA the teacher adds the ground-truth answer to the shared question, so distilling it aligns with answering correctly; in streaming, memory is written before the question arrives, so the student writes from its own visible state while the teacher scores with the question known, misaligning the teacher's preference with optimizing memory for final performance. (2) **Effective-entity memory collapses.** As shown in Figure 1, the student memory retains far fewer visual entities as training proceeds, driven by two causes. First, the existing question-and-GT privilege makes the teacher favor question-useful entities and discriminate against equally real but irrelevant ones. Second, an intrinsic OPSD limitation is that, under token-mean averaging over a memory, the teacher signal is invariant to how many effective entities a rollout grounds, so it cannot prefer a higher-coverage memory over one grounding fewer entities. Both push memory opposite to the streaming goal of retaining as much key visual content as possible.

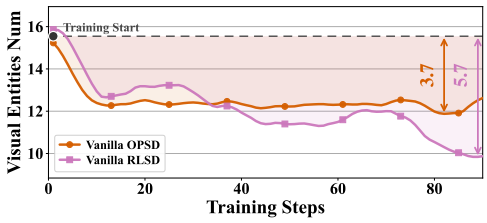


Figure 1: Applying OPSD/RLSD to streaming video collapses memory diversity.

To address these problems, we propose Event-Grounded Self-Distillation (EGSD). Inspired by the event-segmentation theory in cognitive science (Zacks & Tversky, 2001; Zacks et al., 2007), we argue that streaming video understanding based on incremental textual memory is essentially an incremental update over Events, comprising three complementary parts: key visual entities, actions, and details (e.g., spatial relations and OCR). Unlike methods that finely optimize the memory format (Song et al., 2024; Diko et al., 2025), an Event is a deterministic signal that can be extracted offline, verified, and defined independently of any question, giving a stable grounding basis for the teacher and reward. EGSD strengthens the diversity of streaming memory and the final answering accuracy end-to-end by targeting the two problems. For problem (1) we introduce **Multiplicative teacher gating**, where we analyze why the OPSD signal fails in the streaming-video setting and establish that combining it as a multiplicative weight with the outcome reward is the right form for end-to-end streaming optimization that keeps the student learnable from its own memory. For problem (2) we introduce **Event-privileged teacher re-weighting**, where on top of the question and GT answer we add a verifiable set of Event facts to the teacher's privileged information, preventing the teacher's reward from attending only to question-relevant entities and thereby avoiding memory collapse, together with an **Entity-coverage reward** that explicitly prefers rollouts covering more of the clip's effective entities, supplying the coverage preference the token-mean teacher lacks.

EGSD's mechanisms are complementary in streaming video: the Event-based reward drives the model to write more effective entities, while multiplicative gating keeps this optimization aligned with the final answer, so their interaction lifts performance. Our contributions are as follows:

- We analyze and verify the causes of end-to-end optimization failure and memory collapse when vanilla OPSD is applied directly to streaming video understanding.
- We propose EGSD, which identifies verifiable Events as the essential content of streaming memory and strengthens memory diversity and answering accuracy end-to-end through multiplicative gating, Event-privileged re-weighting, and an entity-coverage reward.
- EGSD achieves strong performance on online and offline benchmarks, and memory analysis attributes this to entity recall rising 17.4% at only 6.8% more memory length.

2 Related Work

2.1 Multimodal LLMs for Long Video and Streaming Video Reasoning

Offline long-video LLMs (Yue et al., 2025; Li et al., 2024b) scale along context extension (Chen et al., 2025c) and token compression (Wang et al., 2025a; Qin et al., 2025), but assume the full video is available, whereas streaming exposes neither future frames nor query time. Among causal single-pass methods for streaming video, the history representation determines whether an external function can inspect it. Visual-token or KV compression (Yao et al., 2025; Kim et al., 2026; Xie et al., 2026b; Xu et al., 2026) drops content irrecoverably; hidden-state memory (Chen et al., 2024; Qian et al., 2025; Zeng et al., 2026; Wang et al., 2025b; Sun et al., 2025) is compact but unreadable and unscorable; natural-language memory (Guan et al., 2026; Xie et al., 2026a) is readable and reusable at a token cost. However, none of these imposes a trainable, reward-checkable objective on the memory content itself, which is the premise of our Event-grounded memory.

2.2 Video Reinforcement Learning

Recent methods for RL in long-video reasoning augment the outcome reward with an additional process reward, since over a long horizon the final answer alone cannot tell whether a failure comes from looking at the wrong moment or from flawed reasoning. Yet these rewards score "where to look" rather than "what to remember", whether via key-frame grounding (Cao et al., 2025), zoom-in (Fu et al., 2025b), temporal grounding (Liu et al., 2026), or sampler-MLLM joint optimization (Tan et al., 2026), all evaluating a coordinate against interval annotations. In contrast, none rewards the content of streaming memory, which is exactly where our process reward applies.

2.3 Multimodal On-Policy Self-Distillation

OPSD lets a privileged copy of the student act as its own teacher, giving dense supervision without a larger model. Across all existing instantiations, and this is what matters for streaming, the tokens the teacher scores are the same tokens that produce the answer being evaluated, whether the target is a single-turn image (Zhu et al., 2026), a reasoning chain (Yang et al., 2026), an agent rollout (Zhang et al., 2026), or an evidence-grounded video answer (Wang et al., 2026), so the teacher judges tokens the student already produced toward a question it already knows. Streaming memory writing breaks this premise, as memory turns are written before any question arrives and useful content is defined by future queries, so no existing OPSD signal can supervise it, the gap this work closes.

3 Method

3.1 Analysis of the OPSD Signal in the Streaming-Video Setting

Streaming protocol. We adopt a multi-turn streaming protocol that splits the video stream into K sequentially arriving clips by cumulative visual-token capacity. For the first $K-1$ clips, without knowing the final question q, the model generates a streaming thought z_k from the current visual content c_k and the historical memory m_{k-1} and updates the memory, and when the K-th clip arrives and q becomes visible it produces the final answer a from the accumulated memory,

$$z_k \sim \pi_\theta(\cdot \mid c_k, m_{k-1}), \qquad m_k = \mathrm{Update}(m_{k-1}, z_k). \tag{1}$$

OPSD as a teacher-provided process signal. Vanilla OPD supplies a per-token process signal through the reverse KL between a student P_S and a same-family teacher P_T conditioned on the same prior trajectory,

$$\mathrm{KL}(P_S \,\|\, P_T) = \mathbb{E}_{P_S}[\log P_S(y_t \mid y_{<t}) - \log P_T(y_t \mid y_{<t})]\,. \tag{2}$$

OPSD instantiates the teacher as the student itself additionally conditioned on privileged information ψ, i.e., $P_S(z_k) = \pi_\theta(z_k \mid s_k)$ and $P_T(z_k) = \pi_\theta(z_k \mid s_k, \psi)$ with $s_k = (c_k, m_{k-1})$ the state visible to the student at clip k, so the per-memory signal becomes

$$\delta_k = \log P_T(z_k \mid s_k, \psi) - \log P_S(z_k \mid s_k), \tag{3}$$

which measures the log-probability gain the privileged information brings to this memory.

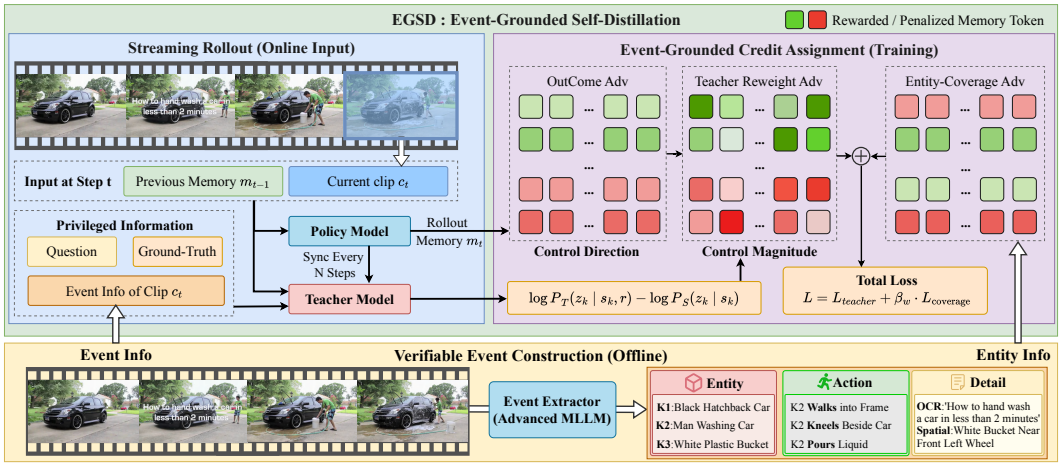


Figure 2: Overview of EGSD. The student writes free-text memory per clip without seeing the question, while a frozen large model extracts a per-clip Event fact set offline. The Event-privileged teacher re-weights student tokens into a multiplicative advantage, and an Event-based entity-coverage reward prefers higher-coverage memories, guiding memory toward grounded diversity.

Multiplicative OPSD fits streaming video. The distinctive challenge of streaming is that the student writes memory before the question arrives, whereas the teacher scores it with the question already known, so δ_k prefers an answer-aware memory that is misaligned with the student's and cannot drive end-to-end optimization on its own. The difficulty is thus how to propagate the outcome to early memory decisions while preserving the teacher's clip-level resolution. Additive schemes such as VOLD (Bousselham et al., 2026), which sum a distillation loss with a GRPO objective, fall short here, as the loss ratio is hard to tune into cooperation toward end-to-end optimization.

We thus propose the key viewpoint that credit assignment in streaming video should let the final result decide the update direction and the privileged teacher the credit magnitude across memory decisions. Accordingly, we adopt outcome-conditioned multiplicative teacher modulation,

$$\hat{A}_k = A_R \cdot \rho_k, \qquad \rho_k > 0, \tag{4}$$

where the positivity of ρ_k keeps $\hat{A}_k$ aligned in sign with A_R, so A_R alone sets the update direction while the teacher gain δ_k, through ρ_k, only modulates the credit magnitude across memory decisions.

3.2 EGSD: Event-Privileged Teacher Re-weighting and Entity-Coverage Reward

§3.1 established multiplicative teacher modulation as the right form of credit assignment for streaming memory but left open what the teacher should be conditioned on. As analyzed in §1, mainstream OPSD conditions the teacher on the question and GT answer, rewarding only question-relevant memory and progressively collapsing the student's output space, at odds with the streaming need for diverse memory. We therefore add a verifiable set of objective facts, termed the Event, to the teacher's privilege, extending its discriminative axis from question relevance to faithfulness to what has happened. As shown in Figure 2, §3.2.1 introduces the Event representation, §3.2.2 derives a privileged teacher anchored on fact grounding, and §3.2.3 develops a coverage reward that ranks memories by how many effective entities they cover.

3.2.1 Event: A Triple-Component Verifiable Signal

Free-text memory z_k is a fluent carrier of a video clip, yet a plain narrative carries no structure verifiable against the visual content, so both the teacher and the reward are left to discriminate on an unverifiable textual surface. We therefore decompose the Event of the k-th clip into three complementary components, extracted offline by a frozen large model and collected into a per-clip fact set,

$$\mathcal{F}_k = \mathcal{E}_k \cup \mathcal{A}_k \cup \mathcal{D}_k, \tag{5}$$

where $\mathcal{E}_k$ (ENTITY) are key visual entities such as people, things, and text signs, $\mathcal{A}_k$ (ACTION) are temporally ordered behaviors among them, and $\mathcal{D}_k$ (DETAIL) are verbatim-checkable attributes such as OCR text, counts, and spatial relations; detailed field definitions and the prompt used to extract Events are in Appendix D. This decomposition breaks a single free-text narrative into typed fact units that can be verified independently, and we use $\mathcal{F}_k$ as the privileged information on which the teacher of §3.2.2 and the coverage reward of §3.2.3 give cleaner, more grounded signals.

3.2.2 EVENT-BASED PRIVILEGED TEACHER

Adding Events to the privileged context. We augment the teacher's privileged context with the per-clip fact set $\mathcal{F}_k$ from §3.2.1, so the teacher and student conditions are

$$P_T(\cdot \mid y_{<t}) \triangleq \pi_\theta(\cdot \mid s_k, \mathcal{F}_k, q, a^\star, y_{<t}), \qquad P_S(\cdot \mid y_{<t}) \triangleq \pi_\theta(\cdot \mid s_k, y_{<t}), \tag{6}$$

which share the same student state s_k and differ only in whether the teacher sees $(\mathcal{F}_k, q, a^\star)$. Adding $\mathcal{F}_k$ extends the discriminative axis of the gain δ_k from question relevance to fact grounding.

Teacher-weight construction and non-decaying gating strength. Substituting this Event privilege into the multiplicative basis of §3.1, we adapt RLSD's sign-modulated positive weight (Yang et al., 2026), the stop-gradient teacher-student probability ratio modulated by the advantage sign,

$$w_t = \left(\frac{P_T(y_t \mid y_{<t})}{P_S(y_t \mid y_{<t})} \right)^{\operatorname{sign}(A_R)}, \tag{7}$$

where $w_t > 1$ amplifies the credit of tokens the Event-privileged teacher favors and $w_t < 1$ suppresses the rest, always in the direction set by A_R. Realizing the memory-level gate ρ_k of §3.1 at token resolution, we combine the clipped weight with the identity weight at gating strength λ,

$$\hat{A}_t = A_R \cdot \rho_t, \qquad \rho_t = (1-\lambda) + \lambda \ \mathrm{clip}(w_t,\, 1-\epsilon_w,\, 1+\epsilon_w). \tag{8}$$

Unlike RLSD, which decays λ to transition back to standard GRPO, streaming memory decisions face the invisible-question gap throughout training, so we keep λ constant at λ_0 and never fade the teacher gate. The clipped surrogate objective of this branch is thus

$$\mathcal{L}_{\text{teacher}}(\hat{A}) = -\,\mathbb{E}\left[\frac{1}{G} \sum_{i=1}^{G} \frac{1}{|y^{(i)}|} \sum_{t=1}^{|y^{(i)}|} \min\left(w_t A_R^{(i)},\ \hat{A}_t^{(i)} \right) \right]. \tag{9}$$

3.2.3 EVENT-BASED ENTITY-COVERAGE REWARD

The privileged teacher of §3.2.2 only re-weights tokens the student has already written, and under token-mean averaging it is invariant to how many reference entities a memory covers, so it cannot prefer a higher-coverage rollout over a lower-coverage one, as proved in Appendix E. To supply this missing preference we introduce an Event-based entity-coverage reward that ranks rollouts directly by how many effective entities they ground. For the k-th clip, the student memory holds N_k^S entities and the Event fact set $\mathcal{F}_k$ supplies N_k^E reference entities; matching the two by one-to-one semantic correspondence, detailed in Appendix D.3, yields N_k^+ effective entities the student has grounded. We then define a grounding rate g_k and a coverage rate cov_k,

$$g_k = \frac{N_k^+}{N_k^S}, \qquad \mathrm{cov}_k = \frac{N_k^+}{N_k^E}. \tag{10}$$

We combine them into a coverage-biased harmonic signal $r_{\text{mem}}(k)$,

$$r_{\text{mem}}(k) = \frac{(1+\beta_e^2)\, g_k\, \mathrm{cov}_k}{\beta_e^2\, g_k + \mathrm{cov}_k}, \qquad \beta_e > 1, \tag{11}$$

where $\beta_e = 1.5$ weights coverage above grounding, rewarding entity recall while still penalizing ungrounded expansion. We center r_{mem} within the group at each clip level to obtain A_{mem}, added to the loss as an independent auxiliary policy-gradient signal,

$$\mathcal{L} = \mathcal{L}_{\text{teacher}}(\hat{A}) + \beta_w \cdot \mathcal{L}_{\text{coverage}}(A_{\text{mem}}). \tag{12}$$

Table 1: Results on the real-time subtasks of OVO-Bench and StreamingBench. Best overall results are in bold and the best results among training-free methods are underlined.

Method	Size	StreamingBench											OVO Real-Time						
		OP	CR	CS	ATP	EU	TR	PR	SU	ACP	CT	Avg.	OCR	ACR	ATR	STU	FPD	OJR	Avg.
Open-source Offline models																			
LLaVA-Video [TMLR'25]	7B	–	–	–	–	–	–	–	–	–	–	–	69.1	58.7	68.8	49.4	74.3	59.8	63.4
LLaVA-OV [TMLR'25]	7B	80.4	74.2	76.0	80.7	72.7	71.7	67.6	65.5	65.7	45.1	71.1	66.4	57.8	73.3	53.4	71.3	62.0	64.0
LongVU [ICML'25]	7B	–	–	–	–	–	–	–	–	–	–	–	53.7	53.2	62.9	47.8	68.3	59.8	57.6
LongVA [TMLR'25]	7B	70.0	63.3	61.2	70.9	62.7	59.5	61.1	53.7	54.7	34.7	60.0	–	–	–	–	–	–	–
Qwen2.5-VL [arXiv'25]	7B	73.0	74.2	85.8	82.4	76.9	80.4	82.4	69.5	65.2	48.2	74.1	84.6	65.1	65.5	54.5	72.3	63.0	67.5
Qwen3-VL [arXiv'25]	8B	71.7	74.2	**89.6**	84.6	68.1	85.1	83.3	**75.6**	72.0	41.5	75.7	89.3	**73.4**	74.1	**64.0**	73.3	64.7	73.1
Open-source Online methods																			
Dispider [CVPR'25]	8B	74.9	75.5	74.1	73.1	74.4	59.9	76.1	62.9	62.2	45.8	67.6	57.7	49.5	62.1	44.9	61.4	51.6	54.5
TimeChat-Online [MM'25]	7B	80.2	82.0	79.5	83.3	76.1	78.5	78.7	64.6	69.6	**58.0**	75.4	75.2	46.8	70.7	47.8	69.3	61.4	61.9
FluxMem [CVPR'26]	7B	80.2	81.1	81.4	85.3	78.0	83.8	80.6	65.9	69.6	52.1	76.4	81.2	59.6	70.7	53.4	75.2	63.0	67.2
StreamForest [NeurIPS'25]	7B	83.1	82.8	82.7	84.3	77.5	78.2	76.9	69.1	**75.6**	54.4	77.3	68.5	53.2	71.6	47.8	65.4	60.9	61.2
Streamo [CVPR'26]	7B	–	–	–	–	–	–	–	–	–	–	–	77.2	66.1	76.7	45.5	66.3	**72.8**	67.4
VST [ECCV'26]	7B	**85.4**	82.0	86.4	**89.1**	74.2	**87.2**	82.4	73.1	73.9	47.3	79.5	80.5	55.1	72.4	55.1	76.2	64.1	67.2
OPSD-based methods																			
Qwen2.5-VL + SFT	7B	73.8	75.0	85.8	82.6	73.1	86.0	75.9	72.4	68.8	49.7	75.4	81.2	70.6	67.2	52.8	77.2	63.0	68.7
+ OPSD [ICLR'24]	7B	74.7	71.9	82.3	85.2	71.2	81.3	73.2	74.0	70.5	52.3	74.8	84.6	68.8	68.1	52.8	74.3	66.3	69.2
+ RLSD [arXiv'26]	7B	76.3	76.6	86.1	84.6	75.6	82.2	73.2	72.4	72.5	51.8	76.7	83.2	64.2	69.0	53.4	76.2	65.2	68.5
+ EGSD (ours)	7B	79.3	80.5	88.3	81.6	**84.4**	83.2	84.3	72.0	73.7	54.9	78.4	84.6	67.9	71.6	53.9	78.2	63.6	70.0
Qwen3-VL + SFT	8B	74.9	74.2	85.5	86.6	76.3	86.0	78.7	75.2	72.0	48.7	76.9	89.3	67.0	75.9	57.9	78.2	67.4	72.6
+ OPSD [ICLR'24]	8B	75.2	78.9	86.4	80.3	80.0	80.7	79.6	72.4	72.2	50.3	76.0	90.6	64.2	73.3	58.4	77.2	67.4	71.9
+ RLSD [arXiv'26]	8B	82.0	**84.4**	84.5	86.2	77.5	81.6	82.4	73.6	73.1	51.8	78.2	86.6	68.8	**77.6**	59.0	**80.2**	67.9	73.4
+ EGSD (ours)	8B	83.7	79.7	89.3	87.5	81.9	83.5	**85.2**	74.0	72.8	53.9	**79.8**	**91.3**	71.6	72.4	60.1	76.2	69.0	**73.4**

Unlike teacher re-weighting, this signal leaves the sign of $\hat{A}$ untouched and ranks rollouts by how many effective entities they ground, so it supplies the coverage preference the token-mean teacher lacks, with independent hyperparameters β_e balancing grounding against coverage and β_w scaling the auxiliary reward against the teacher re-weighting loss. Together, §3.2.2 and §3.2.3 shape streaming memory from two directions, the privileged teacher enforcing truthfulness and the coverage reward enforcing sufficiency. Verbosity gaming gains nothing here: each extra entity must still survive the teacher's gate, so hallucinated additions are down-weighted and the reward accrues to grounded diversity rather than length.

4 Experiments

4.1 Implementation Details

Experiments use **Qwen2.5-VL-7B-Instruct** (Bai et al., 2025b) and **Qwen3-VL-8B-Instruct** (Bai et al., 2025a) as backbones and sample frames uniformly at **1 fps**. Since our memory is text, we first SFT each backbone on the streaming-memory corpus open-sourced by VST (Guan et al., 2026) to keep the intermediate memory concise and high-quality; all subsequent RL runs start from this same base checkpoint. Under the streaming protocol, a split is triggered and a memory write performed every 5000 visual tokens. All methods are evaluated under this same streaming protocol. During testing we cap each inference step, including streaming-think and the final answer, at 8,192 video tokens and limit the thinking count to 4 for efficient evaluation. Low latency is an inherent advantage of this streaming paradigm rather than an optimization target of our Events; we report it in Appendix B. Further training configurations and the RL training data are given in Appendix C.

4.2 Benchmarks and Baselines

Benchmarks. We evaluate comprehensively on five video-understanding benchmarks. Among them, **StreamingBench** (Lin et al., 2026) and **OVO-Bench** (Niu et al., 2025) are used for online video understanding, examining online reasoning ability and temporal perception; **Video-MME** (Fu et al., 2025a) is a comprehensive offline benchmark covering multiple domains and durations;

Table 2: Memory-dependent results on the online video understanding benchmarks OVO-Bench Backward and Forward, as well as on the offline benchmarks VideoMME (w/o sub.), LongVideoBench (LVB), and VideoHolmes (VH).

Method	Size	Online Video								Offline Video			
		OVO-Bench Backward				OVO-Bench Forward				VideoMME		LVB	VH
		EPM	ASI	HLD	Avg.	REC	SSR	CRR	Avg.	Long	Overall		
Open-source Offline models													
LLaVA-Video [TMLR'25]	7B	56.2	57.4	7.5	40.4	34.1	70.0	60.4	54.8	–	63.3	**61.3**	–
LLaVA-OV [TMLR'25]	7B	54.2	55.4	21.5	43.7	25.6	67.1	58.8	50.5	–	58.2	56.3	–
LongVA [TMLR'25]	7B	–	–	–	–	–	–	–	–	47.6	54.3	56.3	–
Video-R1 [NeurIPS'25]	7B	–	–	–	–	–	–	–	–	–	61.4	–	36.5
LongVILA-R1 [NeurIPS'25]	7B	–	–	–	–	–	–	–	–	55.2	65.1	58.0	–
REVISOR [CVPR'26]	7B	–	–	–	–	–	–	–	–	<u>56.2</u>	<u>65.7</u>	57.5	–
Qwen2.5-VL [arXiv'25]	7B	50.5	67.6	<u>42.5</u>	53.5	29.7	50.6	48.3	42.9	46.7	54.7	56.8	33.6
Qwen3-VL [arXiv'25]	8B	55.2	**73.0**	22.6	50.3	28.4	68.4	53.3	50.0	53.8	62.6	59.6	35.9
Open-source Online methods													
Dispider [CVPR'25]	8B	48.5	55.4	4.3	36.1	18.1	37.4	48.8	34.7	–	57.2	–	–
TimeChat-Online [MM'25]	7B	55.9	59.5	9.7	41.7	31.6	38.5	40.0	36.7	48.4	62.4	55.4	–
StreamForest [NeurIPS'25]	7B	58.9	64.9	32.3	52.0	32.8	70.6	57.1	52.5	–	61.4	–	–
InfiniPot-V [NeurIPS'25]	7B	–	–	–	47.6	–	–	–	47.9	53.4	59.3	56.5	–
Streamo [CVPR'26]	7B	55.6	58.1	33.9	49.2	30.8	57.6	**82.5**	**57.0**	–	–	–	–
VST [ECCV'26]	7B	56.9	64.9	**48.4**	**56.7**	33.0	66.9	62.1	54.0	55.3	64.9	58.0	41.9
OPSD-based methods													
Qwen2.5-VL + SFT	7B	53.2	66.9	26.9	49.0	33.1	49.9	57.9	47.0	49.7	58.2	58.6	36.7
+ OPSD [ICLR'24]	7B	53.5	61.5	23.1	46.0	24.5	62.5	56.3	47.8	52.0	60.8	57.8	43.5
+ RLSD [arXiv'26]	7B	54.2	64.9	30.1	49.7	29.8	63.0	57.1	50.0	53.0	61.5	59.6	43.8
+ EGSD (ours)	7B	60.9	<u>68.9</u>	36.0	55.3	<u>34.4</u>	67.9	61.3	54.5	56.1	65.2	60.9	44.5
Qwen3-VL + SFT	8B	<u>61.3</u>	64.9	32.3	52.8	24.8	<u>70.8</u>	64.6	53.4	54.6	63.4	60.4	43.1
+ OPSD [ICLR'24]	8B	59.6	62.8	28.0	50.1	23.8	70.6	<u>65.4</u>	53.3	54.6	63.6	59.7	<u>44.8</u>
+ RLSD [arXiv'26]	8B	58.6	61.5	32.3	50.8	30.7	**72.7**	63.8	55.7	53.8	62.6	60.4	44.3
+ EGSD (ours)	8B	**62.6**	66.2	39.8	<u>56.2</u>	**35.2**	**72.7**	62.1	<u>56.7</u>	**57.1**	**66.5**	<u>61.1</u>	**46.2**

LongVideoBench (LVB) (Wu et al., 2024) targets long-video understanding ability; and **VideoHolmes (VH)** (Cheng et al., 2025) focuses on logical reasoning over video content.

Baselines. For *OPSD-based methods*, we take SFT as the base and train + OPSD (Agarwal et al., 2024), + RLSD (Yang et al., 2026), and + EGSD each independently on top of the same SFT checkpoint, and also report the bare backbone Qwen3-VL-8B, evaluating all rows under one script. *Open-source online* baselines are Dispider (Qian et al., 2025), TimeChat-Online (Yao et al., 2025), StreamForest (Zeng et al., 2026), Streamo (Xia et al., 2026), VST (Guan et al., 2026), and the training-free FluxMem (Xie et al., 2026b) and InfiniPot-V (Kim et al., 2026), the latter two both built on Qwen2.5-VL-7B. Open-source offline models (REVISOR (Li et al., 2026), etc.) and the definition of every benchmark subtask are deferred to Appendix C.

4.3 Main Results

Results on Online Video Understanding. As shown in Table 1, EGSD achieves strong performance in real-time evaluation. Since most streaming baselines build on Qwen2.5-VL, we compare on that backbone against the strongest baseline VST. EGSD does not surpass VST on every StreamingBench subtask, as expected since its Event optimization targets cross-time memory, not single-frame perception. It wins on the four subclasses that require accumulating events across segments, event understanding (EU, +10.2), temporal counting (CT, +7.6), clip summarization (CS, +1.9), and prospective reasoning (PR, +1.9) over VST, covering 778 of 2498 questions (31%). On the remaining single-frame subclasses it largely holds or improves over the backbone, so stronger memory does not erode perception, and on the OVO-Bench Real-Time track it leads VST by 2.8 on average.

Results on Memory-Dependent Understanding. Table 2 evaluates the streaming OVO-Bench Backward and Forward tracks, whose queries target events already streamed past, together with the offline VideoMME, LongVideoBench, and VideoHolmes, where our core advantage concentrates. Comparing on Qwen2.5-VL against the strongest baseline VST, EGSD improves the Backward sub-

sets EPM by 4.0 and ASI by 4.0. On Forward it leads by 0.5, and on the offline benchmarks it improves VideoMME-Overall by 0.3, LongVideoBench by 2.9, and VideoHolmes by 2.6. Against its own SFT start the gains are far larger, with the Backward average up 6.3, the Forward average up 7.5, VideoMME-Overall up 7.0, and VideoHolmes up 7.8. Because VideoHolmes is a purely offline reasoning benchmark, this shows the memory from teacher distillation and the entity-coverage reward transfers beyond streaming without trading off offline quality.

Effect of Model Size and the OPSD Signal. The gains persist across backbone scale. On Qwen3-VL-8B, EGSD leads every memory benchmark on its overall metric, lifting the OVO-Bench Backward / Forward averages from 52.8 / 53.4 to 56.2 / 56.7, VideoMME-Overall from 63.4 to 66.5, LongVideoBench from 60.4 to 61.1, and VideoHolmes from 43.1 to 46.2. Comparing the OPSD and RLSD rungs then isolates our first design choice. Vanilla OPSD often falls below its SFT start on the memory columns, with Backward dropping 49.0 to 46.0 on Qwen2.5-VL, because its teacher scores memory with the dual question and GT known while the student wrote it before any question arrived, misaligning its preference with optimizing memory for the answer. RLSD instead combines the same signal multiplicatively with the outcome reward and turns most memory columns positive again, confirming the multiplicative form is the right way to apply the OPSD signal under the streaming protocol; EGSD builds on this form and pushes the gains further.

4.4 Ablation Studies of EGSD Components

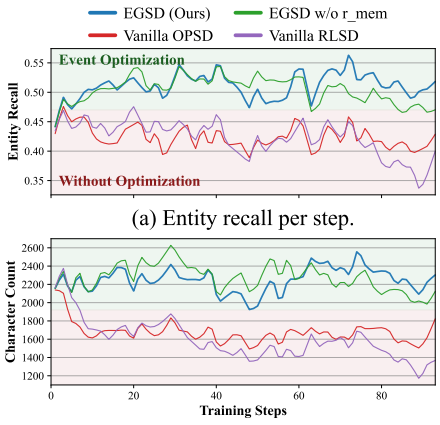
As shown in Table 3, starting from Vanilla RLSD, removing the decay-rate term lifts OVO-Bench, Video-MME, and VideoHolmes by consistent margins, showing the teacher should supervise the whole streaming process. Adding the Event privileged teacher gives the largest single jump, raising OVO-Backward, OVO-Forward, and their average. The complete EGSD then adds the entity-coverage reward and reaches the ladder-highest OVO-Bench average and Video-MME with VideoHolmes level. The three components are complementary and synergistic.

Table 3: Ablation starting from Vanilla RLSD, enabling components one by one. Backbone Qwen3-VL-8B.

Configuration	OVO-Bench			Video-MME	VideoHolmes
	BT	FW	Avg.		
Vanilla RLSD	50.8	55.7	53.3	62.6	44.3
+ remove decay rate	52.1	55.3	53.7	63.8	44.9
+ privileged Event	55.3	56.1	55.7	65.6	**46.5**
+ entity-coverage (EGSD)	**56.2**	**56.7**	**56.5**	**66.5**	46.2

4.5 Memory Quality Analysis

This section characterizes the quality of the memory itself. To separate memory getting longer from memory getting better, we record per step two intrinsic indicators, entity recall, the fraction of the reference Event entities of §3.2.3 that the student trajectory grounds, and the average memory character count for length overhead. The four compared methods are Vanilla OPSD, Vanilla RLSD, EGSD, and EGSD without r_{mem}, the ablation of §4.4. Beyond these training-time curves, Appendix A gives inference-time case studies confirming the same memory-quality advantage on long videos.



(a) Entity recall per step.

(b) Memory length per step.

Figure 3: Per-step entity recall (a) and memory character count (b) for four methods sharing the same SFT start.

4.5.1 Recall and Memory-Length Analysis

Figure 3 shows the per-step curves of entity recall and memory character count. On recall, the Event-privileged teacher and coverage reward r_{mem} jointly fill the teacher's blind spot, so EGSD climbs to 0.518 ($+17.4\%$ over the SFT start) while Vanilla RLSD instead drops to 0.400 (-9.4%), a 29.5% gap at convergence. This gain does not come from writing more: EGSD's memory grows only to 2303 characters, $+6.8\%$ over the shared SFT start, with no runaway growth. Removing r_{mem} confirms its role, recall falls from 0.518 to 0.470 (-9.3%) and length from 2303 to 2124 characters, so r_{mem} raises recall while adding only marginal memory.

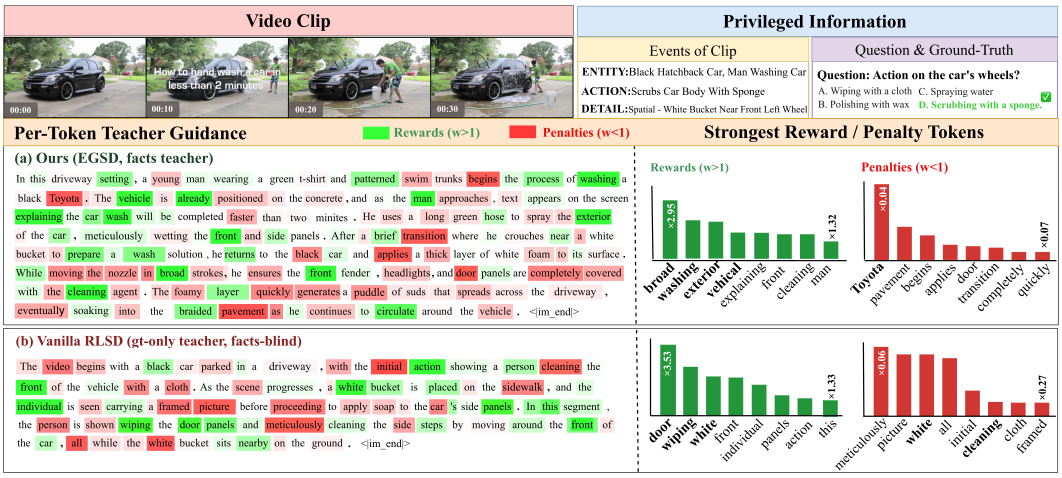


Figure 4: Case study of per-token teacher guidance on the same clip. (a) EGSD teacher with Event facts; (b) Vanilla RLSD teacher with only the question and ground-truth answer.

4.5.2 Tokens Rewarded Analyse

This section analyzes the tokens rewarded by the teacher, using a macro visual-entity ratio across training together with a per-token guidance case study.

Table 4: Reward composition: visual-entity ratio of rewarded tokens across training steps.

Step	10	30	50	70	90
Vanilla RLSD	0.204	0.280	0.237	0.288	0.300
EGSD (ours)	**0.366**	**0.325**	**0.306**	**0.380**	**0.356**

Visual-entity ratio of rewarded tokens. We define visual entities as the union of `ENTITY` words registered for the clip in the fact cache and measure the fraction of teacher-rewarded content words landing on this union (Table 4). EGSD's visual-entity ratio is clearly higher than Vanilla RLSD throughout training, with the gap stably maintained. We never set this ratio as a training target; it emerges after re-weighting the teacher with Event facts, which pushes gradient weight toward entities truly present in the scene and shapes memory-writing habits early.

Case study. We place the Vanilla RLSD and EGSD teachers on the same clip, differing only in privileged information. On a clip of a man hand-washing a black car, asked what action the person performs on the wheels, the registered facts cover the black car, the man in green, a bucket, a hose, and wiping the body with a sponge. As shown in Figure 4, the two teachers reward very different tokens. EGSD concentrates reward on the valid visual entities and actions truly present in the scene, giving high weight to the registered washing action (`broad`$\times$2.95, `wash`$\times$2.05, `exterior`$\times$2.00) and to grounded objects, so it encourages the student to record more effective entities, while its deepest penalty $\times$0.04 falls only on the brand hallucination `Toyota`. Vanilla RLSD instead penalizes salient visual entities only weakly tied to the question, scoring `car`$\times$0.806 and `cleaning`$\times$0.713 where EGSD rewards the same tokens at `cleaning`$\times$1.569 and `car`$\times$1.249. Its guidance is even unstable within one clip, rewarding `white` in the white bucket at $\times$1.996 on first mention but $\times$0.079 on the second, since a question-and-GT privilege cannot judge whether an entity should be recorded for being salient in the scene or only for being relevant to the question. Lacking effective facts, the vanilla teacher penalizes salient entities irrelevant to the question and collapses memory, whereas the Event fact teacher grounds the gradient in real facts and lifts entity recall.

5 Conclusion

This paper introduces EGSD, an on-policy self-distillation method that optimizes streaming video memory end-to-end. Characterizing memory as an incremental update over verifiable Events, EGSD combines multiplicative teacher gating, Event-privileged re-weighting, and an entity-coverage reward to counter the optimization failure and memory collapse of vanilla OPSD. Across online and offline benchmarks, EGSD improves both memory diversity and answering accuracy for streaming video understanding, and we hope it offers useful insights for future work in this direction.

AI USE STATEMENT

In this work, generative AI tools were used solely to polish the writing of parts of the paper, correcting grammar and improving fluency of author-written text. They were not used for research ideation, experimental design, implementation, data analysis, or generating any scientific claims or results. All AI-assisted text was reviewed by the authors, who take full responsibility for the final content of this work.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

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*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun 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, 2025b. URL https://arxiv.org/abs/2502.13923.

Walid Bousselham, Hilde Kuehne, and Cordelia Schmid. Vold: Reasoning transfer from llms to vision-language models via on-policy distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26209–26218, 2026.

Xinye Cao, Hongcan Guo, Jiawen Qian, Guoshun Nan, Chao Wang, Yuqi Pan, Tianhao Hou, Xiaojuan Wang, and Yutong Gao. Videominer: Iteratively grounding key frames of hour-long videos via tree-based group relative policy optimization. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 23773–23783. IEEE, 2025.

Joya Chen, Zhaoyang Lv, Shiwei Wu, Kevin Qinghong Lin, Chenan Song, Difei Gao, Jia-Wei Liu, Ziteng Gao, Dongxing Mao, and Mike Zheng Shou. Videollm-online: Online video large language model for streaming video. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18407–18418. IEEE, 2024.

Joya Chen, Ziyun Zeng, Yiqi Lin, Wei Li, Zejun Ma, and Mike Zheng Shou. Live: Learning video llm with streaming speech transcription at scale. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 29083–29095. IEEE, 2025a.

Yukang Chen, Wei Li, Zhijian Zhang, Yuhao Chen, Ligeng Zhu, Yunhao Fang, Song Han, et al. Scaling rl to long videos. *arXiv preprint arXiv:2507.07966*, 2025b.

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

Junhao Cheng, Yuying Ge, Teng Wang, Yixiao Ge, Jing Liao, and Ying Shan. Video-holmes: Can mllm think like holmes for complex video reasoning? *arXiv preprint arXiv:2505.21374*, 2025.

Anxhelo Diko, Tinghuai Wang, Wassim Swaileh, Shiyan Sun, and Ioannis Patras. Rewind: Understanding long videos with instructed learnable memory. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13734–13743. IEEE, 2025.

Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.

Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Junfei Wu, Xiaoying Zhang, Benyou Wang, and Xiangyu Yue. Video-r1: Reinforcing video reasoning in mllms. *Advances in Neural Information Processing Systems*, 38:99114–99137, 2026.

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

Shenghao Fu, Qize Yang, Yuan-Ming Li, Xihan Wei, Xiaohua Xie, and Wei-Shi Zheng. Love-r1: Advancing long video understanding with an adaptive zoom-in mechanism via multi-step reasoning. *arXiv preprint arXiv:2509.24786*, 2025b.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: On-policy distillation of large language models. *arXiv preprint arXiv:2306.08543*, 2023.

Yiran Guan, Liang Yin, Dingkang Liang, Jianzhong Ju, Zhenbo Luo, Jian Luan, Yuliang Liu, and Xiang Bai. Video streaming thinking: Videollms can watch and think simultaneously. In *European Conference on Computer Vision*, pp. 80–99. Springer, 2026.

Minsoo Kim, Kyuhong Shim, Jungwook Choi, and Simyung Chang. Infinipot-v: Memory-constrained kv cache compression for streaming video understanding. *Advances in Neural Information Processing Systems*, 38:138983–139013, 2026.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024a.

Jiaze Li, Hao Yin, Wenhui Tan, Jingyang Chen, Boshen Xu, Yuxun Qu, Yijing Chen, Jianzhong Ju, Zhenbo Luo, and Jian Luan. Revisor: Beyond textual reflection, towards multimodal introspective reasoning in long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5059–5069, 2026.

Zhang Li, Biao Yang, Qiang Liu, Zhiyin Ma, Shuo Zhang, Jingxu Yang, Yabo Sun, Yuliang Liu, and Xiang Bai. Monkey: Image resolution and text label are important things for large multi-modal models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26753–26763. IEEE, 2024b.

Junming Lin, Zheng Fang, Chi Chen, Haoxuan Cheng, Zihao Wan, Fuwen Luo, Ziyue Wang, Peng Li, Yang Liu, and Maosong Sun. Streamingbench: Assessing the gap for mllms to achieve streaming video understanding. In *ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 12147–12151. IEEE, 2026.

Wenqi Liu, Yunxiao Wang, Shijie Ma, Meng Liu, Qile Su, Tianke Zhang, Haonan Fan, Changyi Liu, Kaiyu Jiang, Jiankang Chen, et al. Videotemp-o3: Harmonizing temporal grounding and video understanding in agentic thinking-with-videos. *arXiv preprint arXiv:2602.07801*, 2026.

Junbo Niu, Yifei Li, Ziyang Miao, Chunjiang Ge, Yuanhang Zhou, Qihao He, Xiaoyi Dong, Haodong Duan, Shuangrui Ding, Rui Qian, et al. Ovo-bench: How far is your video-llms from real-world online video understanding? In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18902–18913. IEEE, 2025.

Rui Qian, Shuangrui Ding, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Dahua Lin, and Jiaqi Wang. Dispider: Enabling video llms with active real-time interaction via disentangled perception, decision, and reaction. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24045–24055. IEEE, 2025.

Minghao Qin, Xiangrui Liu, Zhengyang Liang, Yan Shu, Huaying Yuan, Juenjie Zhou, Shitao Xiao, Bo Zhao, and Zheng Liu. Video-xl-2: Towards very long-video understanding through task-aware kv sparsification. *arXiv preprint arXiv:2506.19225*, 2025.

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

Guangzhi Sun, Yixuan Li, Xiaodong Wu, Yudong Yang, Wei Li, Zejun Ma, and Chao Zhang. video-salmonn s: Memory-enhanced streaming audio-visual llm. *arXiv preprint arXiv:2510.11129*, 2025.

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. *arXiv preprint arXiv:2602.22932*, 2026.

Yi Wang, Xinhao Li, Ziang Yan, Yinan He, Jiashuo Yu, Xiangyu Zeng, Chenting Wang, Changlian Ma, Haian Huang, Jianfei Gao, et al. Internvideo2. 5: Empowering video mllms with long and rich context modeling. *arXiv preprint arXiv:2501.12386*, 2025a.

Yuxuan Wang, Yiqi Song, Cihang Xie, Yang Liu, and Zilong Zheng. Videollamb: Long streaming video understanding with recurrent memory bridges. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 24170–24181. IEEE, 2025b.

Ziyue Wang, Shiqi Huang, Weiwen Xu, Bihan Wen, and Xudong Jiang. Video-opsd: Exploiting privileged visual evidence for on-policy self-distillation in video large language models. *arXiv preprint arXiv:2608.27065*, 2026.

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

Jiaer Xia, Peixian Chen, Mengdan Zhang, Xing Sun, and Kaiyang Zhou. Streaming video instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 31219–31229, 2026.

Junlin Xie, Quanlong Zheng, Ruifei Zhang, Kuo Wang, Yanhao Zhang, Jinguo Luo, Haonan Lu, Xiang Wan, and Guanbin Li. Streamrag: Enhancing real-time video understanding with retrieval augmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 38870–38879, 2026a.

Yiweng Xie, Bo He, Junke Wang, Xiangyu Zheng, Ziyi Ye, and Zuxuan Wu. Fluxmem: Adaptive hierarchical memory for streaming video understanding. *arXiv preprint arXiv:2603.02096*, 2026b.

Ruyi Xu, Guangxuan Xiao, Yukang Chen, Liuning He, Kelly Peng, Yao Lu, and Song Han. Streamingvlm: Real-time understanding for infinite video streams. In *International Conference on Learning Representations*, volume 2026, pp. 61463–61475, 2026.

Chenxu Yang, Chuanyu Qin, Qingyi Si, Minghui Chen, Naibin Gu, Dingyu Yao, Zheng Lin, Weiping Wang, Jiaqi Wang, and Nan Duan. Self-distilled rlvr. *arXiv preprint arXiv:2604.03128*, 2026.

Linli Yao, Yicheng Li, Yuancheng Wei, Lei Li, Shuhuai Ren, Yuanxin Liu, Kun Ouyang, Lean Wang, Shicheng Li, Sida Li, et al. Timechat-online: 80% visual tokens are naturally redundant in streaming videos. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 10807–10816, 2025.

Zihao Yue, Zhenru Lin, Yifan Song, Weikun Wang, Shuhuai Ren, Shuhao Gu, Shicheng Li, Peidian Li, Liang Zhao, Lei Li, et al. Mimo-vl technical report. *arXiv preprint arXiv:2506.03569*, 2025.

Jeffrey M Zacks and Barbara Tversky. Event structure in perception and conception. *Psychological bulletin*, 127(1):3, 2001.

Jeffrey M Zacks, Nicole K Speer, Khena M Swallow, Todd S Braver, and Jeremy R Reynolds. Event perception: a mind-brain perspective. *Psychological bulletin*, 133(2):273, 2007.

Xiangyu Zeng, Kefan Qiu, Qingyu Zhang, Xinhao Li, Jing Wang, Jiaxin Li, Ziang Yan, Kun Tian, Meng Tian, Xinhai Zhao, et al. Streamforest: Efficient online video understanding with persistent event memory. *Advances in Neural Information Processing Systems*, 38:75804–75835, 2026.

Peiyuan Zhang, Kaichen Zhang, Bo Li, Guangtao Zeng, Jingkang Yang, Yuanhan Zhang, Ziyue Wang, Haoran Tan, Chunyuan Li, and Ziwei Liu. Long context transfer from language to vision. *arXiv preprint arXiv:2406.16852*, 2024a.

Yanfei Zhang, Xu Lin, and Chenglin Wu. Stepopsd: Step-aware online preference distillation for agent reinforcement learning. *arXiv preprint arXiv:2605.27140*, 2026.

Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Video instruction tuning with synthetic data. *arXiv preprint arXiv:2410.02713*, 2024b.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026.

Shoutai Zhu, Tianyang Xu, Bin Sun, Mingyuan Xu, Yu Liu, and Qinzhen Guo. Opld: On-policy latent distillation for multimodal reasoning. *arXiv preprint arXiv:2607.28154*, 2026.

A Inference Case Study: Memory Quality

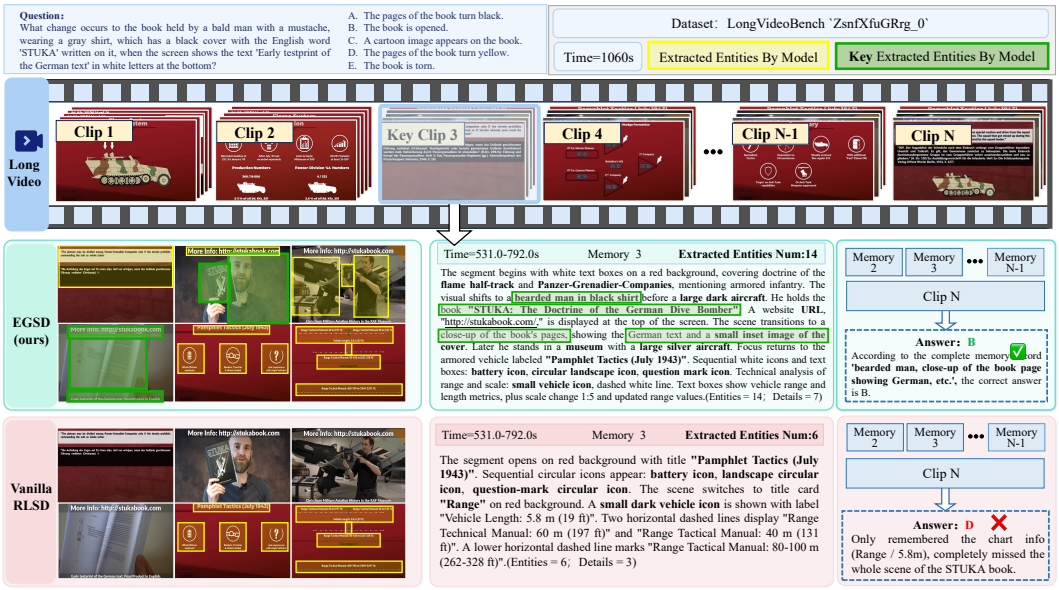


Figure 5: Streaming QA on LongVideoBench sample `ZsnfXfuGRrg_0` (≈18 min). The question-critical evidence lies in the middle Chunk 2 (531.0–792.0 s). Vanilla RLSD collapses this chunk to a few salient mainline entities and misses the book-related evidence, predicting the wrong option D; EGSD additionally retains the transient book-related entities, matches the question, and reaches the ground-truth answer B.

We qualitatively compare the memory quality of EGSD and Vanilla RLSD at inference on two long streaming cases. The two methods share the SFT start and differ only in the RL-stage privileged information; at inference their inputs and outputs are identical, so any divergence in the written memory isolates the effect of the training method. Each case stresses one failure mode: Vanilla RLSD collapses the memory onto a few salient mainline entities, whereas EGSD retains the question-critical evidence. The first case tests whether a transient but question-critical entity buried in a dominant mainline stream can still be recalled (§A.1); the second tests whether a character–prop binding is preserved across time (§A.2).

A.1 Case 1: Recalling Transient Evidence Buried in a Mainline Stream

In sample `ZsnfXfuGRrg_0`, the only supporting evidence sits in the middle chunk, which shows the bearded man in a black shirt and the book *STUKA: The Doctrine of the German Dive Bomber* with its inset cover, while a continuous mainline stream about armored vehicles arrives before and after it.

Vanilla RLSD suffers memory collapse: it keeps only salient mainline entities (*Pamphlet Tactics (July 1943)*, the icon sequence, *Range*, the vehicle icon; 6 entities, 3 details) and discards every book-related entity, so it cannot locate the evidence and predicts the wrong option D. EGSD additionally captures the bearded man, the large dark aircraft, the *STUKA* book with its URL and inset cover, and the museum aircraft (14 entities, 7 details). These low-priority entities survive continual overwriting by later mainline frames, and matching them against the question yields the correct answer B. Recording more valid, question-critical entities is what ultimately makes the answer correct.

A.2 Case 2: Preserving Character–Prop Binding Across Time

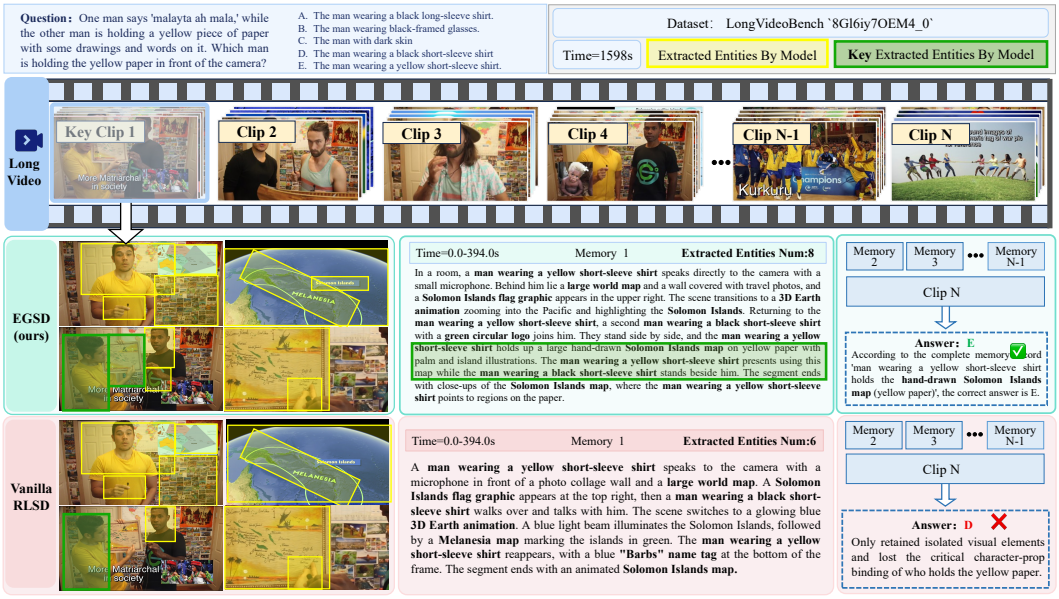


Figure 6: Streaming QA on LongVideoBench sample `8Gl6iy7OEM4_0` (0.0–394.0 s). Answering requires binding the yellow hand-drawn Solomon Islands map to the person who holds it. Vanilla RLSD keeps the salient mainline entities but loses the person–map binding, predicting the wrong option D; EGSD retains both the men and the map together, so it identifies the holder and reaches the ground-truth answer E.

In sample `8Gl6iy7OEM4_0`, the question asks which man holds the yellow sheet of hand-drawn maps, so the answer hinges on a relation rather than a single entity: the man in the yellow shirt, the man in the black shirt, and the hand-drawn Solomon Islands map must stay bound to who holds what.

Vanilla RLSD again suffers memory collapse: it retains only isolated mainline entities (the man in the yellow shirt, the large world map, the Solomon Islands flag graphic, the man in the black shirt, the 3D Earth animation, Solomon Islands; 6 entities) and drifts toward secondary props such as the Melanesia map and the "Barbs" name tag. Dropping the person–map binding, it cannot decide who holds the paper and predicts the wrong option D. EGSD instead preserves the question-critical structure, capturing the same mainline entities plus the green circular logo and the wall of travel photos, and crucially keeps the man in the yellow shirt *holding* the hand-drawn map (8 entities). The relational binding survives temporal overwriting, so matching the memory against the question yields the correct answer E. Beyond recalling more entities, EGSD retains the relational bindings among them.

B Inference Latency

We report two latency notions and, following VST's Table 6, keep them apart. *TTFT* (Time-To-First-Token) is the delay from posing the question to the first answer token, and is the operative metric for direct-answer and online streaming methods. *End-to-end QA latency* (E2E) runs from the end of video input to the last token of a full answer, and is the only faithful metric for offline with-CoT methods, whose reasoning chain is produced *after* the question — reporting TTFT for them would undercount the true cost by an order of magnitude. All "ours" rows are measured on one harness and GPU batch with `failure_rate`=0; the rest are quoted from the VST paper.

Table 5: Inference latency on VideoHolmes (p50), reported for both the Qwen2.5-VL-7B and Qwen3-VL-8B backbones. Offline with-CoT rows report end-to-end QA latency (E2E); direct-answer and online rows report TTFT. Rows shaded green are offline, orange online; numbers not marked "ours" are quoted from the VST paper.

Type	Method	Metric	Latency (s)
Offline	Qwen2.5-VL-7B w/CoT	E2E	5.30
Offline	Video-R1 w/CoT	E2E	8.80
Offline	Qwen2.5-VL-7B w/CoT (ours repro)	E2E	5.68
Offline	Qwen3-VL-8B w/CoT (our 8B backbone)	E2E	5.89
Offline	Qwen2.5-VL-7B direct-answer	TTFT	0.54
Online	Dispider-7B	TTFT	1.10
Online	VST-7B	TTFT	0.56
Online	VST-32B	TTFT	1.40
Online	**EGSD-7B (ours)**	TTFT	0.90
Online	**EGSD-8B (ours)**	TTFT	0.73

Holding the backbone fixed, the two regimes diverge sharply. On Qwen3-VL-8B, an offline with-CoT answer needs 5.89 s end-to-end, whereas our online streaming memory returns a first token in 0.73 s ($\approx 8\times$ faster); on Qwen2.5-VL-7B the contrast is 5.68 vs 0.90 s ($\approx 6\times$). The gain is structural, not a smaller-model artifact: the streaming think is front-loaded, completing asynchronously as clips arrive before the question rather than stacking after it, so our TTFT stays in the same sub-second range as VST-7B (0.56 s) even with the added memory mechanism. Across seven benchmarks the sample-weighted median first token is 0.98 s (8B), on par with the GT-only variant, so the memory shaped by r_{mem} costs no extra latency. Because VST rows come from different hardware, we compare orders of magnitude and the online–offline gap, not absolute seconds.

C Training Configuration and Data

C.1 Post-training Hyperparameters

Table 6 lists the key post-training configuration of the main EGSD run. Training runs on the **verl** framework with **32 × 80G H800**, using GRPO as the advantage estimator, on top of which we add the EGSD multiplicative teacher re-weighting and the entity-coverage reward r_{mem}. The teacher gate is $\rho_t = (1-\lambda) + \lambda \cdot \text{clip}(w_t, 1-c, 1+c)$ with $w_t = (P_T/P_S)^{\text{sign}(A_R)}$; at λ=0.5, c=0.2 the teacher scales a token's policy-gradient contribution by at most $\pm 10\%$. The entity-coverage reward is the F-β score (β=1.5) between the entities mentioned in the student memory and the reference set $\mathcal{F}_k$, matched lexically and normalised within each GRPO group, and enters as an independent additive policy-gradient term.

C.2 Training Data

The main experiment uses an RL training set of 6000 question–video items covering 5688 distinct videos (825-item validation set). Each item carries a self-contained problem (MCQ or numeric/yes-no, wrapped in the streaming-memory instruction), an answer, the videos, and source clip metadata; the task mix is 5706 MCQ, 290 numeric, and 4 yes-no. Video durations span 2.6–2046 s (median 70.8 s, mean 198.2 s, p90 510 s), matching the streaming regime the model serves. The source composition is LLaVA-Video-178K (3413), hdvila (705), Grounded-VideoLLM (352), RepCount (290),

SpaceR (253), Ego4D (230), Charades v1 (219), YouCook2 (141), NExT-GQA (129), Perception-Test (122), MovieChat-1K (110), and Video-Holmes (36).

The privileged fact sets $\mathcal{F}_k$ and the r_{mem} references are read from a precomputed cache (20,075 clip entries over 5688 videos; 15,862 usable for r_{mem} under coverage and minimum-entity gates). Facts are produced by a frozen Qwen3-VL-235B-A22B-Instruct extractor plus one 235B reviewer pass, each clip decoded at `sample_fps`=1.0 with up to 48 frames. The cache is keyed by (video, chunk) under the streaming fingerprint requiring `STREAM_MAX_CHUNKS`=32, so offline chunk boundaries align with the online streaming reader.

Table 6: Key post-training configuration of the main EGSD run.

Parameter	Value
Base / policy model	Qwen3-VL-8B (SFT)
Optimizer / LR	AdamW / 1×10^{-6}
Epochs / max steps	1 / 50
Samples per prompt n	8
Rollout / global batch	64 / 64
Token budget per chunk	5000
Max frames / chunks	64 / 32
Sampling rate	1.0 fps
Teacher	frozen self-copy
Sync interval	every 20 steps
Mixing weight λ / clip c	0.5 / 0.2
r_{mem} weight β_w	0.3
F-β / matcher	β=1.5 / lexical

C.3 Benchmarks and Baselines

Benchmark subtasks. We report each benchmark's official subtask splits and metrics. **StreamingBench** (Lin et al., 2026) scores 2498 real-time questions by doc-level micro accuracy over ten subtasks: object recognition (OP), causal reasoning (CR), clip summarization (CS), attribute recognition (ATP), event understanding (EU), text-rich understanding (TR), prospective reasoning (PR), spatial understanding (SU), action recognition (ACP), and counting (CT). **OVO-Bench** (Niu et al., 2025) asks each question at a fixed timestamp and allows only content streamed up to that point, over three tracks whose per-track score is the macro average of its subtasks: *Real-Time* (837 questions) tests perception of the current frame (OCR, ACR, ATR, STU, FPD, OJR); *Backward* (631) recalls events already streamed past, e.g. episodic memory (EPM), action-sequence identification (ASI), and hallucination detection (HLD); and *Forward* (1567) tracks or extrapolates still-developing events, e.g. long-window recall (REC), SSR, and CRR. For offline understanding, **Video-MME** (Fu et al., 2025a) spans multiple domains and durations, **LongVideoBench (LVB)** (Wu et al., 2024) targets long-video understanding, and **VideoHolmes (VH)** (Cheng et al., 2025) focuses on logical reasoning over video content.

Baselines. Beyond the OPSD-based rows and the streaming baselines described in the main text, Tables 1 and 2 list open-source offline models for reference: LLaVA-Video (Zhang et al., 2024b), LLaVA-OneVision (Li et al., 2024a), LongVA (Zhang et al., 2024a), Video-R1 (Feng et al., 2026), LongVILA-R1 (Chen et al., 2025b), and REVISOR (Li et al., 2026). For methods we do not retrain, quoted numbers follow their original papers or the VST official table.

D Event Triples: Design and Extraction

D.1 Design and Offline Extraction

Why ENTITY / ACTION / DETAIL. The three components form a minimal cover of what a video is later asked. `ENTITY` is a 1–5 word canonical noun phrase plus aliases, the only component feeding r_{mem}, so bare generics (`a man`, `thing`) are rejected. `ACTION` (agent-act-object with an absolute-second interval) carries event order and causality, and `DETAIL` (verbatim OCR, counts, spatial relations, attributes) grounds the perceptual question types; both feed the teacher prompt only. Together they give $\mathcal{F}_k$ what free text lacks: determinism, verifiability, and privilege without leakage—the extractor sees more frames but records only what is confirmable and is never a student input.

Offline extraction prompt. $\mathcal{F}_k$ is produced chunk-by-chunk by a frozen 235B extractor and checked by one 235B reviewer pass, recording only what is verifiably visible in the clip; the question guides prioritisation but is never answered. Facts are line-oriented, not JSON, to allow line-level reviewer patches.

Offline fact extractor — system prompt (core rules)

```
You are a precise visual fact extractor for video.  You are shown ONE clip of a longer
video, plus the question that will later be asked about the whole video.  You extract what
is VERIFIABLY VISIBLE in this clip.
WHAT WILL BE ASKED LATER (extract FOR these; never answer them here):  verbatim on-screen
text (OCR), object attributes incl.  colour and material, spatial relations, object
counts, how many times a repeated action completed, the ORDER of events, and causal
chains.  A fact list that records only "who did what" loses every one of them.  A fact you
invent is worse than a fact you omit.
OUTPUT GRAMMAR --- one fact PER LINE, fields separated by " | " as key=value.  ENTITY
| id=k1 | type=person|object|text|location|group|animal|other | name=<canonical noun
phrase, 1-5 words> | attrs=<visible; single hyphen if none> | aliases=<other surface
forms; single hyphen if none> ACTION | id=a1 | ag=<k1[,k2] or short noun> | act=<verb
phrase> | obj=<k2 or short noun; hyphen if none> | t=<start>-<end> DETAIL | id=d1 |
kind=ocr|count|spatial|attribute|state|temporal | text=<the fact, carrying a concrete
specific> | ref=<k1[,a1]; hyphen if none>
FIELD DISCIPLINE. ENTITY.name is matched by a reward function, so it must be a CONCRETE
noun phrase, never a bare generic; put other surface forms in aliases.  DETAIL.kind=ocr
copies on-screen text VERBATIM in single quotes; never guess a title, name, brand, or
number.  ACTION.t is in ABSOLUTE seconds of the whole video.  SCALE --- aim for 6-18 lines
per clip; prefer FEWER, RICHER lines.  Output ONLY fact lines:  no prose, no JSON, no code
fences, no numbering.
```

D.2 A Full Video's Per-Clip Event Facts

The box above shows the grammar; here we give one clip's $\mathcal{F}_k$ so the reader can see the privileged information the student receives during streaming. The video is the car-wash clip of Figure 4 (ActivityNet, `v_ELiXlJUBzzw`): a man in a green shirt hand-washes a black car, split into 4 chunks. Its question is *What action does the person perform on the car's wheels?* with ground truth **D (scrubs with a sponge)**, supported by the wheel-scrub `ACTION` lines in chunks 1–2; spraying water is a near distractor targeting the body. Each teacher sees only its own chunk's $\mathcal{F}_k$; the student sees none of it.

chunk 0 (t=0.0–41.1s): scene set-up, wheels not yet touched

```
ENTITY | id=k1 | type=object | name=black hatchback car | attrs=black, parked, wet,
soap-covered | aliases=Scion xD, black car
ENTITY | id=k2 | type=person | name=man washing car | attrs=green shirt, patterned shorts,
barefoot
ENTITY | id=k3 | type=object | name=white plastic bucket | ENTITY | id=k4 | name=green
garden hose | ENTITY | id=k5 | name=orange spray bottle
ACTION | id=a2 | ag=k2 | act=kneels beside car | obj=k1 | t=8.0-9.0
ACTION | id=a3 | ag=k2 | act=pours liquid from orange spray bottle onto car | t=9.0-10.0
ACTION | id=a4 | ag=k2 | act=scrubs car body with sponge | t=11.0-23.0
DETAIL | id=d1 | kind=ocr | text='How to hand wash a car in less than 2 minutes!'
DETAIL | id=d2 | kind=spatial | text=white bucket on the ground near the front left wheel
DETAIL | id=d5 | kind=attribute | text=car has black rims and tinted windows
```

Two points recur across the full video. Entities evolve by surface: the same car is aliased `Scion xD` here, `black minivan` and `Hyundai` in later chunks, yet the alias lists resolve them to one entity under r_{mem}. And hallucination is verifiable: the car is never registered as `Toyota` anywhere in $\mathcal{F}$, so the `Toyota` in the RLSD memory of Figure 4 is a genuine fabrication, and the teacher's deepest penalty landing on it is exactly the grounding a pre-existing annotation provides.

D.3 Student-Side Entity Extraction and F-β Matching

The extractor of §D.1 turns the reference into a set of matchable entity units; the other half of r_{mem} extracts visual entities from the student's *own* memory into the same `list[list[str]]` shape, so one scorer applies to both sides. In the loop we use a microsecond-level heuristic extractor (noun-phrase scanning with head-word merging, attribute/location splitting of prepositional phrases, quoted-OCR units, and generic-name dropping); an offline Qwen3-1.7B backend serves as an oracle for ablations.

Given student units $\mathcal{S} = \{s_i\}$ and reference units $\mathcal{R} = \{r_j\}$, scoring proceeds in four steps. First, the `name` and `aliases` of a unit describe the same thing, so their content words are pooled into

one matchable word set,

$$W(u) = \bigcup_{\text{surface} \in u} \text{contentwords}(\text{surface}). \tag{13}$$

Second, unit similarity is a soft overlap normalised by the smaller unit, so a long reference name (`elderly man in a blue hospital gown`) and a short student name (`elderly man`) are not penalised as Jaccard would,

$$\text{overlap}(a, b) = \frac{|W(a) \cap W(b)|}{\min\big(|W(a)|,\, |W(b)|\big)}. \tag{14}$$

Third, over all pairs with overlap $\geq \tau$ we take a one-to-one greedy matching in descending score to obtain the hit count n_{hit}; the one-to-one constraint prevents a student from hitting one reference entity five times by writing five surface forms. Fourth, we combine a recall-biased F-β with a precision slack that frees $a = \lceil \text{slack} \cdot n_R \rceil$ unmatched student entities, offsetting the reference's natural under-coverage while still charging memory that dumps whole neighbouring clips,

$$\text{prec} = \frac{n_{\text{hit}}}{\max\big(1,\ \max(n_{\text{hit}},\, n_S - a)\big)}, \qquad \text{rec} = \frac{n_{\text{hit}}}{n_R}, \tag{15}$$

$$r_{\text{mem}} = \frac{(1 + \beta_e^2)\,\text{prec} \cdot \text{rec}}{\beta_e^2\,\text{prec} + \text{rec}}, \qquad \beta_e = 1.5,\ \text{slack} = 0.5,\ \tau = 0.5. \tag{16}$$

$\beta_e = 1.5$ weights recall above precision; a pure-recall reward would not penalise verbose dumping, whereas this F-β does. Chunks with an empty reference return `None` rather than 0.0, so they do not contribute to the reward.

D.4 Validating the Extractor and Matcher

The loop uses a cheap heuristic extractor and a lexical matcher rather than an online model. Three checks confirm this does not distort r_{mem}.

The matcher separates same-clip from cross-clip. On the 722-clip cache we score each clip's entity table by its own aliases (*same*, the recall ceiling) and by a neighbouring clip's entities (*cross*, the leakage floor); the gap must clear 0.370 (Table 7). Lexical wins at 0.523, above embedding (0.406) and their union (0.474): the synonym misses of word overlap (`sofa`/`couch`) cost less than embedding's recall loss, and the union only loosens *cross*. We therefore keep the lexical matcher.

The extractor passes a synthetic self-test. For each clip we build *perfect* (its own facts), *verbose* (correct plus padding), and *wrong* (another clip's) memories, then extract and score: 0.739 / 0.602 / 0.422. The verbosity gap 0.137 and the clip-specificity gap 0.317 both clear their thresholds, so padding and mis-recall are penalised in the directions r_{mem} intends.

Heuristic and model extraction agree. On 800 real step-50 memory lines the heuristic and an offline Qwen3-1.7B extractor reach symmetric F1 $= 0.615$ at near-identical density (19.6 vs 17.3 units per clip), differing only in granularity (`bowl with spoon` vs `bowl+spoon`). Removing the $\sim$3% malformed units leaves group-normalised advantages correlated at 0.985 with the original, since a systematic extractor bias cancels in the GRPO group mean. Model extraction thus adds no signal to justify its cost, so we use the heuristic.

E Why the Token-Mean Teacher Is Coverage-Invariant

§1 claims that a token-mean teacher cannot prefer a memory that grounds more effective entities. We make this precise here.

Let a clip have reference entities $\mathcal{E}$ with $|\mathcal{E}| = N$, and let a memory rollout $y = (y_1, \ldots, y_L)$ cover the set $C(y) = \{e \in \mathcal{E} : e \text{ appears in } y\}$. The teacher weights each token by $w_t \geq 0$ and distills under the token-mean loss

$$\mathcal{L}_{\text{OPSD}}(y) = -\frac{1}{L} \sum_{t=1}^{L} w_t \, \log \pi_\theta(y_t \mid y_{<t}), \tag{17}$$

Table 7: Matcher separation on the 722-clip fact cache. *same* is the recall ceiling (self-query by aliases), *cross* the leakage floor (query by a neighbouring clip); separation = same − cross must exceed 0.370.

Matcher	same ↑	cross ↓	separation ↑	Verdict
Lexical (default)	1.000	0.477	**0.523**	PASS
Embedding (τ=0.72)	0.808	0.402	0.406	PASS
Hybrid (union)	1.000	0.526	0.474	PASS

so the signal a rollout carries reduces to its average weight $\bar{w}(y) = \frac{1}{L}\sum_t w_t$.

Two facts follow. First, w_t exists only for tokens y actually wrote, so an entity $e^* \notin C(y)$ receives no gradient at all—the teacher can reshape written tokens but never push the student to emit a missing one. Second, for an on-policy, locally fluent student the teacher approves most reasonable tokens with a similar weight $w_t \approx c$, so $\bar{w}(y) \approx c$ regardless of $|C(y)|$. Two equal-length rollouts, one covering five entities and one covering two, then carry the same average weight and the same gradient magnitude, even though the former is the better streaming memory. The token-mean teacher is thus blind to coverage.

The entity-coverage reward closes exactly this gap. Writing its coverage component as $r_{\text{mem}}(y) \propto |C(y)|/N$ (the recall term of the F-β signal in §3.2.3; the precision term only rescales it), the return depends on the reference set, not the written tokens, so within a GRPO group the normalized advantage $\hat{A}_i \propto r_{\text{mem}}(y_i) - \bar{r}$ favors the higher-coverage rollout and raises the probability of the entities that lower-coverage rollouts drop. It still cannot invent an entity absent from every rollout—no on-policy signal can—but whenever group rollouts differ in coverage it supplies the preference the teacher lacks.

F Prompt Templates

EGSD uses three prompt roles: the privileged teacher, the student memory writer during training, and the unprivileged evaluation reader. The student reads a video cut into K chunks by cumulative visual-token budget, writing one memory increment per intermediate chunk and answering only when the final chunk delivers the question. All strings below are read from the training and evaluation code.

Teacher prompt (facts-privileged). The main experiment uses the facts teacher, a superset of the pure-GT teacher: it keeps the GT-answer line and the phase-aware split (write-memory on intermediate chunks, answer on the final chunk) and adds this chunk's fact set $\mathcal{F}_k$ (`ENTITY`/`ACTION`/`DETAIL`, pre-extracted offline over the whole video by a frozen VLM at `sample_fps`=1.0, up to 48 frames per clip, then attributed to chunks). Teacher and student consume identical per-chunk visual tokens and write in the same prose register, so the fact set is the teacher's only advantage and the token-level signal stays on the SFT distribution. Because these facts come from denser frames than the streaming student sees, the prompt hard-constrains it to record only what the current frames confirm.

Teacher prompt — intermediate chunk (write memory)

```
{{ question }}
The question this recording will later be used to answer has the known-correct answer: {{
hint }}
=== NEWLY SUPPLIED VISUAL ENTITIES FOR THIS SEGMENT ===
An ADDITIONAL privileged list of the visual entities, actions, and details that a denser
inspection of THIS SAME segment found. The model whose recording is compared with yours
does not receive it. Treat it as a checklist of what deserves attention, not as text to
copy line by line.
{{ facts }}
An ENTITY line names something worth naming explicitly; an ACTION line names who did what
to what and when; a DETAIL line is a concrete specific worth stating (verbatim on-screen
text in quotes, a spatial relation, an attribute such as colour or material, or a count).
Verification. The list was produced from MORE frames than this segment shows you, so
record ONLY what you can confirm in the frames in front of you. Never copy a quoted
string, name, number, time, or count you cannot read or count in these frames.
Record the visual evidence from this segment that supports reaching that answer. Describe
only what is visible --- do not state the answer, do not mention that an answer was
provided, and do not answer the question yet.
```

On the final chunk $\mathcal{F}_k$ is empty and the teacher answers from the accumulated memory plus the reference answer.

Student training prompt. Each turn is assembled from three roles byte-aligned with the SFT base: a `system` role (`You are a helpful assistant.`), a memory role initialised with `[System]\nYou are a Streaming Video Analyst.` and blind-appended per chunk, and a `user` role carrying the current segment. Intermediate turns receive frames plus a write-memory instruction; the final turn adds the question. During RL the question is hidden on intermediate turns (`STREAM_QUESTION_IN_THINK=never`), so the student writes memory before knowing it.

Student prompt — intermediate turn (write memory)

```
Time={start}-{end}s
Watch this video segment and describe all visual evidence that may be useful for answering
future questions. Be concise but preserve key objects, actions, counts, text, directions,
and temporal events.
```

Evaluation prompt. Evaluation runs the prose (VST) memory branch with no privileged information: no GT, no facts, no delta-op hint. Memory parsing is `auto`, enabling structured delta-op only when an `event_memory_sft.json` marker sits beside the checkpoint; ours carries none, so the policy trained to write structured memory instead reads and writes its own prose. The two modes differ in memory surface form, not policy weights. Evaluation reuses the same three-role template and string constants as training (one `PromptConfig`): the `system` role is `You are a helpful assistant.`, the memory role initialises with `[System]\nYou are a Streaming Video Analyst.` and blind-appends one line per segment, and the intermediate write-memory turn is byte-identical to the training box above. When the final turn reads memory, it is prefixed as below.

Evaluation — accumulated memory (previous-text role)

```
Here is the accumulated memory from previous video segments:
[System]
You are a Streaming Video Analyst.
Time={start:.1f}-{end:.1f}s {segment memory the model wrote for this segment}
Time={start:.1f}-{end:.1f}s {segment memory the model wrote for this segment}
...
```

Evaluation — final answer turn

```
Time={q_end:.1f}s {task-native question}
Answer with the option's letter from the given choices directly.
```

The final `Answer with...` line is appended by `task.build_prompt` for multiple-choice tasks only. Segmentation follows the `em_aligned` protocol: StreamingBench and

LongVideoBench split into 4 segments, other tasks fall back to duration quartiles. This role alone is fully unprivileged and runs on prose rather than structured memory.

G Limitations and Future Work

Our exploration is still limited to several settings, left for future work. (1) This work only studies self-distilled OPSD, where the teacher is the student itself holding privileged information, and does not yet involve vanilla OPD with a stronger external model as the teacher; whether the latter can provide complementary and stronger process signals on streaming memory remains to be verified. (2) We use verifiable Event facts as the grounding object of the teacher and the reward; how to generalize this idea to open-domain settings that lack structured fact annotations, or where the facts themselves are highly subjective, is an open question. (3) The experiments concentrate on a single-pass, no-look-back streaming protocol and the current backbone scale; the behavior of the method with larger models, longer training, and integration with online retrieval / tool calling has not yet been examined.