EVISTREAM: EVIDENCE-PRESERVING MEMORY FOR STREAMING VIDEO UNDERSTANDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Streaming video assistants must continuously process an unbounded visual stream before knowing what users may ask in the future, requiring efficient memory without losing evidence needed for later reasoning. Existing approaches mainly reduce visual computation or build retrievable memories, but often leave unclear which visual evidence should survive compression and how it should be recovered when needed. We introduce **EviStream**, a training-free evidence-preserving memory framework based on MLLMs. EviStream follows the principle of preserving *residual information rather than repeated observations* through complementary semantic memory and visual evidence. *Incremental Semantic Memory* (ISM) organizes the incoming stream into adaptive windows, removes temporal redundancy, and incrementally consolidates each window into semantic deltas, yielding a compact evolving textual history. To complement this semantic abstraction, *Memory-Aligned Evidence Keyframe Selection* (MEKS) preserves sparse visual evidence according to temporal novelty, the information gap left by semantic consolidation, and query relevance when available. Once a query arrives, *Continual Evidence-Polling Reasoning* (CEPR) retrieves relevant semantic memory and complementary visual evidence together with live observations, while continuously consolidating and re-retrieving new observations when current evidence is insufficient. EviStream achieves state-of-the-art real-time understanding performance on StreamingBench and OVO-Bench, reaching 84.1% and 76.8% accuracy, respectively, without additional training. Code will be released.

1 INTRODUCTION

Streaming video assistants aim to understand continuously arriving visual streams and interact with users in real time. Unlike the conventional offline setting, success in streaming video understanding (SVU) Qian et al. (2024); Xiong et al. (2025); Lin et al. (2026) hinges on both *what to answer* and *when to answer*. A capable assistant must produce accurate responses by reasoning over relevant evidence accumulated from the stream, while remaining sufficiently responsive to answer as soon as the required evidence becomes available. Answering too early may lead to incomplete or incorrect predictions, whereas answering too late reduces the usefulness of real-time interaction. Streaming video understanding therefore poses the dual challenge of maintaining sufficient information for reliable reasoning while continuously determining the appropriate moment to respond.

Addressing this dual challenge requires controlling the ever-growing video history while preserving information needed for future reasoning. Existing studies mainly follow two directions. Visual-centric approaches Yao et al. (2025); Zhang et al. (2026b); Wu et al. (2024b); Chen et al. (2026); Wang et al. (2026a); Xie et al. (2026); Ning et al. (2025) reduce visual processing through compression or selective retention, improving efficiency but limiting long-horizon context. Memory-centric approaches Ning et al. (2025); Xiong et al. (2025); Jiang et al. (2026); Liang et al. (2026); Di et al. (2025); Xiao et al. (2026) instead organize history into compact retrievable memories, extending accessible context through KV management or semantic summarization. Recent works Qian et al. (2025); Wang et al. (2026b); Azad et al. (2026); Xueyun et al. (2026) further introduce active response mechanisms, allowing models to respond only when sufficient evidence is available.

Despite this progress, three key limitations remain. First, existing systems still lack an explicit mechanism for determining which visual evidence to retain after historical information has been

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

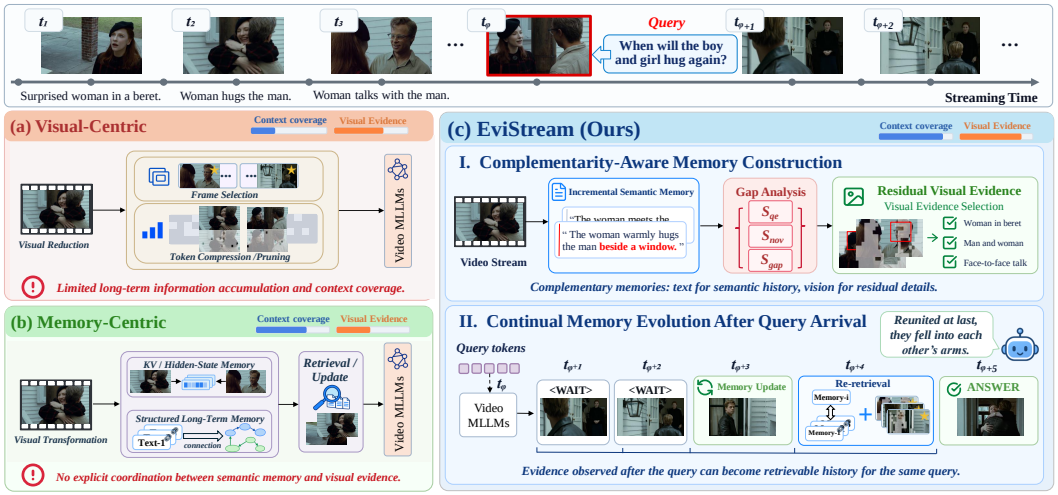

Figure 1: **Comparison of existing paradigms and EviStream.** (a) Visual-centric methods preserve visual fidelity but provide limited long-term information accumulation and context coverage. (b) Memory-centric methods extend historical context through compact memory, yet semantic memory and visual evidence are not explicitly coordinated, which may lead to the loss of fine-grained visual evidence. (c) EviStream jointly maintains evolving semantic memory and complementary visual evidence, while continually incorporating new observations for subsequent response decisions.

compressed into more compact representations. Without such coordination, visually stored evidence may either duplicate information preserved elsewhere or discard fine-grained details that later become important. Second, redundancy in streaming video arises at multiple levels, including temporally adjacent observations and repeatedly accumulated history, yet these forms of redundancy are typically handled separately rather than through a unified residual-preserving principle. Third, maintaining dynamic memory after query arrival remains less explored in existing streaming systems. When the currently available evidence is insufficient, newly observed content may become critical before the query can be answered. Consolidating such evidence into retrievable memory allows subsequent response decisions to use observations that have already left the live buffer.

To address these issues, we introduce **EviStream**, a training-free evidence-preserving memory framework for streaming video understanding. EviStream is built on a *complementarity-aware memory principle*: semantic and visual memories are constructed jointly, such that *semantic memory captures what can be compactly abstracted*, while *visual memory preserves what semantic consolidation leaves behind*. To realize this principle efficiently, EviStream performs residual memory consolidation through two complementary memory components. **Incremental Semantic Memory (ISM)** removes temporally redundant observations through recency-preserving visual compression and incrementally records only newly introduced semantic information relative to the accumulated history. **Memory-Aligned Evidence Keyframe Selection (MEKS)** preserves visual content according to temporal novelty and the information gap left by semantic abstraction, while further incorporating query relevance once user intent becomes available. Finally, EviStream maintains continual post-query memory evolution through **Continual Evidence-Polling Reasoning (CEPR)**. When the available evidence is insufficient, memory construction continues after query arrival. Newly observed content is consolidated into long-term memory, retrieval is refreshed, and the updated evidence is reused in subsequent response decisions. This allows EviStream to accumulate evidence while waiting and respond once sufficient information becomes available.

We evaluate EviStream on the standard StreamingBench and OVO-Bench benchmarks, where it achieves state-of-the-art performance across retrospective understanding, real-time perception, and proactive response. Without additional training or parameter updates, EviStream reaches 84.1% real-time visual understanding accuracy and 63.4% overall accuracy, outperforming Qwen3.5-9B by 5.8% and 4.8%, respectively, while also requiring lower peak memory and achieving lower time-to-first-token (TTFT). Beyond overall accuracy, our analyses show that each design component contributes to a distinct aspect of streaming reasoning: residual memory consolidation improves long-horizon efficiency, complementarity-aware evidence preservation retains fine-grained information that would otherwise be lost during abstraction, and continual post-query memory evolution be-

comes increasingly beneficial as the interval between query arrival and answerable evidence grows. Our contributions are summarized as follows:

- We introduce **EviStream**, a training-free framework with complementarity-aware evidence memory, where semantic memory captures compact high-level history and visual evidence preserves information beyond semantic consolidation.
- We propose *residual memory consolidation* through Incremental Semantic Memory (ISM) and Memory-Aligned Evidence Keyframe Selection (MEKS). ISM preserves semantic deltas over temporally compressed streams, while MEKS retains residual visual evidence based on temporal novelty and the information gap left by semantic abstraction.
- We perform *post-query memory evolution* with Continual Evidence-Polling Reasoning (CEPR). When evidence is insufficient, new observations are consolidated, re-retrieved, and reused in subsequent decisions, allowing evidence to accumulate while waiting.
- Extensive experiments on standard benchmarks demonstrate state-of-the-art performance, while ablations validate the contribution of each component of our unified system.

2 Related Work

Streaming Video Understanding. Streaming video understanding extends offline video reasoning to continuous, causal interaction, requiring models to maintain historical context and answer queries as new observations arrive Chen et al. (2024); Lin et al. (2026); Niu et al. (2025). Recent work further moves toward *proactive* interaction, where models must decide not only what to answer but also when to respond Qian et al. (2025); Ding et al. (2025); Yu et al. (2026); Wang et al. (2026b); Azad et al. (2026); Yang et al. (2026); Zhang et al. (2026c); Shi et al. (2026). This shift makes response timing a central challenge in streaming video assistants.

Visual-Centric Approaches. A major line of work improves streaming or long-video scalability by reducing the amount of visual inputs processed at each step. Existing approaches Wu et al. (2024b); Yao et al. (2025); Zhang et al. (2026b); Xie et al. (2026); Wang et al. (2026a) exploit the substantial temporal and spatial redundancy in videos through frame selection, visual-token compression, pruning, or sparse computation, substantially reducing the per-step cost of processing long visual streams. These methods are primarily *visual-centric*, focusing on *which visual evidence to process or preserve at the representation level*. While effective in reducing per-step computation, their accessible historical context is still largely determined by the amount of visual information that can be retained or processed within a bounded budget. This limits long-horizon context coverage and motivates approaches that explicitly maintain persistent memory over the stream.

Memory-Centric Approaches. Recent work increasingly treats streaming video understanding as a persistent memory problem, aiming to extend historical context beyond the directly accessible visual window. A first line of methods maintains past information through KV caches or compact hidden states, using compression, eviction, retrieval, or selective reuse to control the cost of an ever-growing stream Chen et al. (2026); Di et al. (2025); Yang et al. (2025); Xiao et al. (2026); Xu et al. (2026). Beyond KV-based memory, more recent methods construct structured long-term memories, including event-level Zeng et al. (2026), prototype-based Minh et al. (2026), and semantic memories Zhang et al. (2025a); Liu et al. (2026b;a), enabling continual consolidation and selective retrieval over longer horizons. To conclude, memory-centric methods substantially expand accessible historical context, but long-range coverage does not by itself ensure that fine-grained visual evidence remains available for later queries.

3 Method

3.1 Framework Overview

We consider a video stream $\mathcal{V} = (x_t)_{t\geq 1}$ sampled at a fixed rate. At step t, the system accesses only the observed prefix $\mathcal{V}_{\leq t}$. A query q arrives at t_q and may concern past, present, or future events. Streaming video understanding combines historical context with sequential observations Zhang et al. (2026b); Xie et al. (2026). In proactive interaction, the system also determines when the available

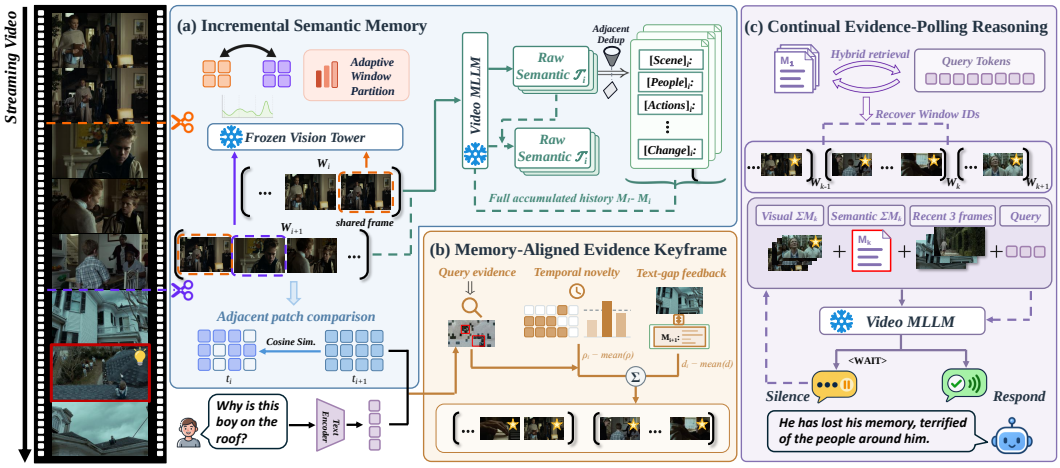

Figure 2: **EviStream:** (a) Incremental Semantic Memory (ISM) dynamically consolidates the video stream into evolving semantic memory, while (b) Memory-Aligned Evidence Keyframe Selection (MEKS) preserves complementary visual evidence not captured by semantic abstraction. After query arrival, (c) Continual Evidence-Polling Reasoning (CEPR) combines retrieved memory, keyframes, and live frames for response decisions, while memory continues evolving during polling.

evidence supports a response

$$o_t = \mathcal{F}(q, \mathcal{V}_{\le t}) \in \mathcal{Y} \cup \{\varnothing\}, \qquad t \ge t_q, \tag{1}$$

where $\mathcal{F}$ is the online system, $\mathcal{Y}$ the textual response space, and $\varnothing$ silence. After the query q arrives, the system continuously observes the incoming video stream and reassesses whether to respond as new evidence becomes available.

We propose **EviStream**, a training-free framework that equips MLLM with an external dual memory that evolves continuously over time, as shown in Figure 2. Before query arrival, EviStream continuously performs evidence-preserving memory construction by converting the incoming stream into two complementary memories. **ISM** first partitions incoming frames into adaptive windows and removes temporally redundant visual tokens. Each completed window is then summarized as a textual delta conditioned on the accumulated history, producing a compact and progressively evolving textual memory (Section 3.2). In this way, ISM preserves temporal and semantic residuals by compressing redundant observations and recording only newly introduced semantics. Since textual abstraction inevitably omits part of the original visual evidence, **MEKS** further uses each raw semantic delta to retain sparse visual observations that remain insufficiently represented in textual memory, together with temporal novelty and query relevance when available (Section 3.3). This explicitly links semantic consolidation to cross-modal evidence preservation. The resulting dual memory is therefore complementary by design. Textual memory captures the evolving semantic history, while visual memory preserves residual evidence that cannot be reliably represented using only language. After query q arrival, **CEPR** retrieves relevant textual memories and associated visual evidence from long-term memory, combines them with the most recent live observations, and performs response polling. Memory updates continue while the model waits, making new evidence retrievable for subsequent decisions (Section 3.4).

3.2 Incremental Semantic Memory

Incremental Semantic Memory removes temporal redundancy before recording semantic updates in an ordered history. Fixed-length windows in existing streaming methods Chen et al. (2026); Kim et al. (2026) can split changing events or repeatedly process stable content. We use adaptive windows with bounds $L_{\min}$ and $L_{\max}$ and one-frame overlap. After $L_{\min}$ frames, a cosine similarity below τ_{w} to the window anchor triggers a boundary; reaching $L_{\max}$ forces closure. Appendix A.2 specifies the boundary and overlap rules. For a window $W_k = (x_{k,1}, \ldots, x_{k,n_k})$, let $h_{k,i,p} \in \mathbb{R}^d$ denote the visual token at patch $p \in \{1, \ldots, J_k\}$ in frame i. Following adjacent-frame comparison in TimeChat-Online Yao et al. (2025), we remove the earlier token from each redundant pair. With

threshold τ_{p}, the retained indices are

$$R_{k,i} = \begin{cases} \{1, \ldots, J_k\}, & i \in \{1, n_k\}, \\ \{p \,|\, \cos(h_{k,i,p}, h_{k,i+1,p}) \le \tau_{\mathrm{p}}\}, & 1 < i < n_k. \end{cases} \tag{2}$$

Comparison waits for the next frame. Boundary frames remain complete, and retained tokens keep their original spatial and temporal positions. These tokens form the compressed sequence $\widetilde{H}_k$, while their retention statistics supply the temporal-novelty signal for MEKS. The frozen model converts $\widetilde{H}_k$ into a raw textual delta conditioned on the accumulated history $\mathcal{M}_{k-1}$. Adjacent exact deduplication removes lines matching the preceding raw delta, and the remaining increment is appended

$$\begin{aligned} T_k &= F_\theta\left(p_\Delta, \mathcal{M}_{k-1}, \widetilde{H}_k\right), \\ \Delta_k &= \mathrm{Dedup}_{\mathrm{adj}}(T_k, T_{k-1}), \qquad \mathcal{M}_k = \mathcal{M}_{k-1} \oplus \Delta_k, \end{aligned} \tag{3}$$

where p_Δ requests newly observed events and state changes, $\oplus$ denotes ordered concatenation, $\mathcal{M}_0$ is empty, and $\Delta_1 = T_1$. Earlier entries remain unchanged, and each entry retains its source-window identifier and temporal span. Generation budgets and retrieval caching are detailed in Appendices A.4. The raw T_k supplies text-gap feedback to MEKS.

3.3 Memory-Aligned Evidence Keyframe Selection

The raw delta T_k may omit counts, brief actions, or other fine-grained evidence. MEKS selects complementary frames after formal delta generation, reusing the projected tokens from Section 3.2 without additional visual encoding. The tokens are projected into the LLM input embedding space. MEKS caches the pre-pruning mean $\bar{h}_{k,i} = J_k^{-1} \sum_{p=1}^{J_k} h_{k,i,p}$ and represents tokenized text $s = (w_1, \ldots, w_{L_s})$ as $z(s) = L_s^{-1} \sum_{\ell=1}^{L_s} e_\theta(w_\ell)$, where e_θ is the base LLM's input embedding lookup. These representations share dimensionality but can retain modality offsets.

Query evidence. For an available query, let $A_{k,i}(q)$ contain the $\ell_{k,i} = \min(K_e, |R_{k,i}|)$ retained patches most similar to $z(q)$. For $\ell_{k,i} > 0$, the score is

$$S^{\mathrm{qe}}_{k,i} = \frac{1}{\ell_{k,i}} \sum_{p \in A_{k,i}(q)} \cos\left(h_{k,i,p}, z(q)\right). \tag{4}$$

Averaging the most relevant patches emphasizes localized evidence and limits dilution by surrounding content.

Temporal novelty. Let $\mathcal{I}_k = \{2, \ldots, n_k - 1\}$ and $\rho_{k,i} = |R_{k,i}|/J_k$. MEKS estimates relative temporal variation from patch retention

$$S^{\mathrm{nov}}_{k,i} = \begin{cases} \rho_{k,i} - \dfrac{1}{|\mathcal{I}_k|} \displaystyle\sum_{j \in \mathcal{I}_k} \rho_{k,j}, & i \in \mathcal{I}_k,\ |R_{k,i}| > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

A positive score indicates an above-average fraction of patches differing from the following frame. The mean excludes fully retained boundary frames and adjusts for the window's overall retention level. This query-independent signal favors local changes and transient content.

Text-gap feedback. MEKS compares each pre-pruning frame representation with the current raw delta T_k, using the centered discrepancy as a proxy for visual evidence insufficiently represented in the delta, where subtracting the window mean removes the shared discrepancy offset and all frames contribute equally, including boundary frames and frames without retained patches:

$$d_{k,i} = 1 - \cos\left(\bar{h}_{k,i}, z(T_k)\right), \; S^{\mathrm{gap}}_{k,i} = d_{k,i} - \frac{1}{n_k} \sum_{j=1}^{n_k} d_{k,j}. \tag{6}$$

The combined score determines the retained keyframes

$$\begin{aligned} S_{k,i} &= \alpha S^{\mathrm{qe}}_{k,i} + \beta S^{\mathrm{nov}}_{k,i} + \gamma S^{\mathrm{gap}}_{k,i}, \\ \mathcal{K}_k &= \{x_{k,i} \,|\, S_{k,i} > \tau_{\mathrm{e}}\}, \end{aligned} \tag{7}$$

where α, β, and γ weight the signals and τ_{e} controls retention. Selection uses novelty and text-gap feedback before query arrival, with query evidence added for windows summarized during an active query. Keyframes retain timestamps and window identifiers for joint retrieval with text.

3.4 Continual Evidence-Polling Reasoning

Memory construction continues after query arrival, allowing new observations to become retrievable evidence while the model waits. A temporary retrieval view combines accumulated deltas with any provisional entry from the ongoing window. Dense and lexical rankings identify the top-K_{r} memory lines and their source windows; Appendix A.5 gives the retrieval formulation.

Anchor Neighborhood Expansion. Anchor Neighborhood Expansion (ANE) recovers event context by including all deduplicated entries from matched windows and their temporal neighbors. For matched window indices $\mathcal{H}_t(q)$ and temporal-neighborhood radius P, the selected windows are

$$\mathcal{J}_t(q) = \left(\bigcup_{k \in \mathcal{H}_t(q)} \{k - P, \ldots, k + P\} \right) \cup \mathrm{Recent}_{N_{\mathrm{r}}}(t), \tag{8}$$

restricted to available indices. The latest N_{r} windows provide mandatory recent context, including the ongoing window when it has a provisional entry. Shared window identifiers recover the associated MEKS keyframes alongside the selected text. Before each poll, a coverage check determines whether to summarize the ongoing window. A provisional summary covers all arrived frames and supplies polling context without updating committed history or running MEKS. Formal summarization later replaces this retrieval record under the same window identifier and appends the deduplicated formal delta to history. Appendix A.6 gives the trigger and update rules.

Response polling. Each poll uses the ordered context

$$\mathcal{C}_t(q) = \mathcal{V}_t^{\mathrm{hist}} \oplus \mathcal{M}_t^{\mathrm{hist}} \oplus \widetilde{T}_t \oplus \mathcal{V}_t^{\mathrm{live}} \oplus q, \tag{9}$$

where the blocks contain historical keyframes, text from completed windows, the available provisional summary, and live frames. Historical images and text are chronological within their respective blocks. The frozen model decides

$$\begin{aligned} (a_t, y_t) &= F_\theta \left(p_{\mathrm{dec}}, \mathcal{C}_t(q) \right), \\ o_t &= \begin{cases} \varnothing, & a_t = \text{silence}, \\ y_t, & a_t = \text{respond}, \end{cases} \end{aligned} \tag{10}$$

where p_{dec} is the decision instruction, a_t the decision, and $y_t \in \mathcal{Y}$ the candidate response. Memory construction and response inference run concurrently, with polling performed once per second. The model remains silent until sufficient evidence is available, while completed-window and provisional updates refresh retrieval and context for subsequent decisions.

4 Experiments

4.1 Experimental setup

Datasets and Metrics. To evaluate the effectiveness of EviStream, we conduct experiments on StreamingBench Lin et al. (2026), which includes multiple real-time visual understanding tasks and Proactive Output (PO), and OVO-Bench Niu et al. (2025), which evaluates historical information retrieval, real-time content analysis, and Forward Active Responding, for streaming video understanding. We further evaluate on Video-MME Fu et al. (2025) and LongVideoBench Wu et al. (2024a) for offline long-video understanding. Offline evaluation uses the lmms-eval Zhang et al. (2024) framework to ensure standardized benchmarking. We compare trained and training-free streaming methods and report benchmark accuracy, with the Proactive Output (PO) timing criterion specified in Section 4.2. Efficiency is measured by peak GPU memory usage and time to first token (TTFT).

Implementation Details. We use Qwen3.5-9B Bai et al. (2025); Qwen Team (2026) without additional training and sample videos at 1 FPS. Adaptive windows contain 20–30 frames, and each

Table 1: Performance on StreamingBench for real-time visual understanding and Proactive Output (PO).Among open-source streaming Video-LLMs, including both training-based and training-free methods, the **best** and second-best results are highlighted. † denotes reproduced backbone results.

Method	# Frames	PO	Real-Time Visual Understanding										
			OP	CR	CS	ATP	EU	TR	PR	SU	ACP	CT	Avg.
Human	-	100	89.5	92.0	93.6	91.5	95.7	92.5	88.0	88.8	89.7	91.3	91.5
Proprietary Models													
Gemini 1.5 Pro	1 fps	45.1	79.0	80.5	83.5	79.7	80.0	84.7	77.8	64.2	72.0	48.7	75.7
GPT-4o	64	**56.9**	77.1	80.5	83.9	76.5	70.2	83.8	66.7	62.2	69.1	49.2	73.3
Online Video MLLMs (Training)													
VideoLLM-online [CVPR 2024]	2 fps	–	39.1	40.1	34.5	31.1	46.0	32.4	31.5	34.2	42.5	27.9	36.0
Flash-VStream [ICCV 2025]	1 fps	–	25.9	43.6	24.9	23.9	27.3	13.1	18.5	25.2	23.9	48.7	23.2
Dispider [CVPR 2025]	1 fps	–	74.9	75.5	74.1	73.1	74.4	59.9	76.1	62.9	62.2	45.8	67.6
TimeChat-Online [ACM MM 2025]	1 fps	–	80.2	82.0	79.5	83.3	76.1	78.5	78.7	64.6	69.6	**58.0**	75.4
ReKV [ICLR 2025]	0.5 fps	–	74.4	78.9	78.6	77.1	68.3	67.9	67.6	62.6	64.3	44.6	69.1
StreamForest [NeurIPS 2025]	1 fps	–	83.1	82.8	82.7	84.3	77.5	78.2	76.9	69.1	75.6	54.4	77.3
ThinkStream [ECCV 2026]	2 fps	–	83.7	70.3	76.0	82.7	76.7	78.2	76.9	70.7	74.4	45.2	75.0
Em-Garde [CVPR 2026]	1 fps	38.0	80.4	74.2	89.9	84.0	78.9	82.2	87.0	71.1	68.8	42.5	76.7
Response-G1 [ACL 2026]	0.2-1 fps	44.0	84.0	78.1	88.0	84.6	74.8	83.5	83.3	63.0	69.3	**58.0**	77.5
Online Video MLLMs (Training-Free)													
FreshMem [arXiv 2026]	–	–	84.5	83.6	77.6	83.3	78.1	80.4	70.4	74.4	66.9	30.0	74.2
FluxMem [CVPR 2026]	1 fps	–	80.2	81.1	81.4	85.3	78.0	83.8	80.6	65.9	69.6	52.1	76.4
HERMES [ACL 2026]	1 fps	–	83.1	76.6	87.4	85.5	78.0	86.9	85.2	76.8	72.2	49.2	79.1
QueryStream [ICLR 2026]	1 fps	-	82.4	84.4	79.2	82.4	78.0	81.3	78.7	65.0	69.3	47.3	75.3
STRIDE [ECCV 2026]	1 fps	42.8	77.1	75.0	77.3	72.8	76.4	77.9	76.9	69.9	**84.3**	46.1	74.2
Qwen3.5-9B†	1 fps	21.6	82.1	**87.5**	87.1	83.7	**83.0**	78.2	82.4	69.5	73.0	55.9	78.3
+ EviStream (Ours)	1 fps	47.6	**89.7**	86.7	**91.8**	**86.7**	82.4	**90.7**	**89.8**	**81.3**	79.6	52.1	**84.1** (↑5.8%)

inference call uses at most 128 historical keyframes. Qwen3-Embedding-0.6B Zhang et al. (2025c) serves as the retrieval encoder for semantic memory search. Experiments run on eight NVIDIA A800 GPUs with 80 GB of memory each. Full implementation details are provided in Appendix A.2.

4.2 Main Results on Streaming Video Benchmarks

Results on StreamingBench Table 1 presents the results on StreamingBench. Compared with existing trained models and training-free online methods, EviStream achieves state-of-the-art overall performance in real-time visual understanding, improving Qwen3.5-9B from 78.3 to 84.1. It outperforms the proprietary models GPT-4o Hurst et al. (2024) and Gemini 1.5 Pro Team et al. (2024), and exceeds the strongest trained method, Response-G1 Ma et al. (2026), by 6.6 points. These results show that EviStream uses continuously accumulated historical evidence more effectively. Residual memory consolidation and adaptive window segmentation summarize events with stable visual dynamics and provide the model with more relevant textual and visual context during retrieval. In particular, EviStream substantially outperforms comparable models on clip summarization (CS) and text-rich understanding (TR), approaching human performance.

Results on OVO-Bench Table 2 compares the methods on OVO-Bench. EviStream leads five of six Real-Time Visual Perception tasks, which require continuous analysis with historical information, and achieves a state-of-the-art score of 76.8 among online models. This performance comes with reduced GPU memory usage without additional resource consumption. Within-window visual redundancy compression and incremental memory reduce context occupied by visual placeholders, retaining only complementary keyframe tokens to tightly control context changes. On Backward Tracing, EviStream improves Qwen3.5-9B from 46.2 to 58.4, surpassing QueryStream Zhang et al. (2026b) and HERMES Zhang et al. (2026a). We attribute this improvement to MEKS, which supplements historical event reasoning with visual information absent from semantic memory.

Results on Proactive Response tasks We evaluate proactive streaming response on Proactive Output (PO) in StreamingBench, with results reported in Table 1. For PO, the model is polled at the query time and once per second within the $[-4, +4]$-second interval centered on the ground-truth

Table 2: Evaluation results on OVO-Bench across Real-Time Visual Perception and Backward Tracing. EviStream achieves leading results among online models across diverse temporal reasoning tasks, while its memory usage remains independent of the frame rate.

Method	# Frames	Real-Time Visual Perception							Backward Tracing				All
		OCR	ACR	ATR	STU	FPD	OJR	Avg.	EPM	ASI	HLD	Avg.	Avg.
Human	-	94.0	92.6	94.8	92.7	91.1	94.0	93.2	92.6	93.0	91.4	92.3	92.8
Proprietary Models													
Gemini 1.5 Pro	1 fps	87.3	67.0	80.2	54.5	68.3	67.4	70.8	68.6	75.7	52.7	62.3	66.5
GPT-4o	64	69.1	65.1	65.5	50.0	68.3	63.7	63.6	49.8	71.0	55.4	58.7	61.2
Online Video MLLMs (Training)													
Flash-VStream	1 fps	25.5	32.1	29.3	33.7	29.7	28.8	29.9	36.4	33.8	5.9	25.4	27.6
Dispider	1 fps	57.7	49.5	62.1	44.9	61.4	51.6	54.6	48.5	55.4	4.3	36.1	45.4
TimeChat-Online	1 fps	74.5	48.6	68.1	48.3	69.3	59.8	61.4	56.9	64.9	11.8	44.5	53.0
StreamForest	1 fps	68.5	53.2	71.6	47.8	65.4	60.9	61.2	58.9	64.9	32.3	52.0	56.6
Em-Garde	1 fps	76.5	61.5	65.5	47.2	70.3	57.1	63.0	48.2	63.5	45.2	52.2	57.6
Response-G1	0.2–1 fps	90.1	74.3	75.9	59.6	69.3	71.7	73.6	55.6	66.9	33.9	52.1	62.8
Online Video MLLMs (Training-Free)													
FreshMem	–	77.2	60.6	70.7	56.7	63.4	70.7	66.7	57.2	56.8	46.8	54.0	60.4
FluxMem	1 fps	81.2	59.6	70.7	53.4	75.2	63.0	67.2	–	–	–	–	–
HERMES	1 fps	85.3	65.2	74.4	54.0	68.3	66.8	69.0	44.8	64.2	39.8	49.6	59.3
QueryStream	1 fps	74.5	47.7	70.7	46.6	71.3	57.6	61.4	54.2	63.5	8.6	42.1	51.8
STRIDE	1 fps	76.5	64.2	79.3	61.2	73.3	63.6	69.7	57.2	**72.3**	14.0	47.8	58.8
Qwen3.5-9B†	1 fps	74.5	64.2	75.0	62.4	**79.2**	63.6	69.8	**59.6**	70.3	8.6	46.2	58.0
+ EviStream	1 fps	**91.3**	**76.2**	**81.9**	**64.6**	73.3	**73.4**	**76.8** (↑7.0%)	54.9	62.8	**57.5**	**58.4** (↑12.2%)	**67.6** (↑9.6%)

response time. Accuracy is computed as

$$\mathrm{Acc} = \frac{1}{N_{\mathrm{PO}}} \sum_{i=1}^{N_{\mathrm{PO}}} \mathbb{I}\left(\left|\hat{t}_i - t_i^{\mathrm{gt}}\right| \leq 2\right), \tag{11}$$

where N_{PO} is the number of PO samples, $\hat{t}_i$ is the time of the first "Yes" response for sample i, and t_i^{gt} is the ground-truth response time. On PO, EviStream improves the backbone score from 21.6 to 47.6 and exceeds Response-G1 by 3.6 points. This result supports the effectiveness of the complete evidence-polling system under the stated protocol.

4.3 Results on Offline Video Benchmarks

We further test on offline long-video understanding benchmarks. In Table 3, EviStream achieves 63.6, exceeding all training-based and training-free methods. This result demonstrates that EviStream generalizes to offline QA by retaining key information from long videos and supporting queries across windows. On Video-MME Fu et al. (2025), it achieves 54.6 on long videos lasting 30–60 minutes and 66.4 overall across videos lasting 1–60 minutes. These results show that organizing event context through complementarity-aware evidence memory also provides an effective long-term memory representation for offline long-video understanding.

4.4 Ablation Studies

Efficiency of Event-Adaptive Stream Encoding. Table 4 examines the accuracy and resource costs of window segmentation within the memory construction pipeline. We measure TTFT from query input to the first output token. Since the latency reported for FluxMem Xie et al. (2026) excludes visual encoding, we use its reproduction in ObjectStream Dong et al. (2026) as a separate reference. With Qwen3.5-9B, all three window variants improve accuracy and reduce peak GPU memory usage and TTFT relative to the bare backbone. The histogram variant compares normalized HSV histograms of the current frame and window anchor, whereas the ViT variant uses their visual-feature similarity. The ViT variant achieves the highest score of 63.4 and the lowest measured peak memory usage of 19.24 GB, compared with 58.6 and 21.73 GB for the backbone. Its TTFT is 5.76 s, below the backbone's 6.31 s but above the fixed-window and histogram variants at 5.68 s and 5.43 s. The more semantically informative visual representations of the ViT-based strategy provide a better balance between accuracy and efficiency. The results further indicate that encoding delta memory in advance and caching historical retrieval embeddings reduces repeated computation at query time.

Table 3. Accuracy on offline long-video benchmarks, LongVideoBench and VideoMME (w/o subtitles).

Method	# Frames	LongVideo Bench	VideoMME	
			long	overall
Video Length	–	*8 sec–60 min*	*30–60 min*	*1–60 min*
Offline Video MLLMs				
LLaVA-OneVision	32	56.5	48.8	58.4
Kangaroo	64	54.2	46.6	56.0
MovieChat	2048	–	33.4	38.2
LongVU	1 fps	–	–	60.6
Q-Frame	4+8+32	58.4	48.3	58.3
FLoC	128	58.0	50.9	61.0
Nar-KFC	8	57.7	52.0	61.6
FOCUS	64	63.5	**56.1**	65.4
Online Video MLLMs				
Flash-VStream	1 fps	–	50.3	61.2
Dispider	1 fps	–	49.7	57.2
TimeChat-Online	1 fps	57.7	52.4	63.3
ViSpeak	64	–	–	60.0
ThinkStream	2 fps	56.4	–	61.9
StreamBridge	1 fps	59.1	–	64.4
QueryStream	1 fps	58.0	52.9	63.8
FluxMem	1 fps	61.1	54.0	65.3
HERMES	1 fps	–	54.4	62.0
EviStream	1 fps	**63.6**	54.6	**66.4**

Table 4. Efficiency analysis of different window segmentation strategies on OVO-Bench with Qwen3.5-9B. Adaptive strategies use their best-performing thresholds.

Variant	Peak Mem.↓	TTFT↓	Perf.↑
Qwen2.5-VL-7B (baseline)	35.8GB	13.43s	49.8
+ FluxMem	25.8GB	6.64s	53.3
Qwen3.5-9B (baseline)	21.73GB	6.31s	58.6
+ Fixed Window	21.12GB	5.68s	62.7
+ Histogram(τ_{hist}=0.4)	19.28GB	**5.43s**	62.1
+ ViT-based(τ_{vit}=0.75)	**19.24GB**	5.76s	**63.4**

Table 5. Ablation of post-query memory evolution for PO, including Query-Time Retrieval (QR), Window Refresh (WR), and Provisional Update (PU). All methods perform retrieval once when the query arrives.

Memory Variant	Mechanism			Acc. (%)
	QR	WR	PU	
(B) Frozen at Query	✓	×	×	45.9
(C) w/o Provisional	✓	✓	×	47.0
(A) Co-Evolution	✓	✓	✓	**47.6**

Sensitivity to Adaptive Window Bounds. We vary the window bounds over $[10, 40]$, $[10, 30]$, $[20, 40]$, and $[20, 30]$ to examine temporal context for each semantic consolidation step. As shown in Figure 3, increasing $L_{\min}$ from 10 to 20 improves the score under both upper bounds. The numbers of windows decrease from 59.3K to 32.7K and from 60.6K to 33.7K, a reduction of 44%. This reduces model calls required for incremental memory updates and provides a longer minimum context for generating each delta. Raising $L_{\max}$ from 30 to 40 does not improve performance. The adopted $[20, 30]$ configuration achieves the highest score of 84.05, suggesting that these bounds preserve sufficient local context while limiting the temporal span of each consolidation step.

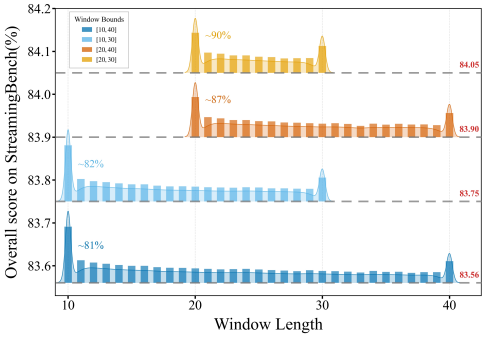

Figure 3. Adaptive window bounds on StreamingBench. Dashed levels mark scores; colored profiles show length distributions. Percentages give the combined share at both bounds.

Ablation on Post-Query Memory Evolution. To examine the effect of memory evolution after query arrival on proactive responding, we compare different forms of state evolution. As shown in Table 5, freezing memory after the query yields 45.9% accuracy. Enabling window-level memory refresh improves accuracy to 47.0%, showing that newly consolidated evidence supports subsequent response decisions. Adding provisional updates further increases accuracy to 47.6% by incorporating recent observations before formal window closure. Overall, these results demonstrate that continual memory evolution after query arrival improves proactive responding, with both window refresh and provisional updates contributing to the gain.

5 Conclusion

We presented **EviStream**, a training-free framework for streaming video understanding with continually evolving semantic memory and complementary visual evidence. By preserving residual information rather than repeated observations, EviStream combines visual compression, incremental semantic memory, and memory-aligned keyframe selection to retain temporal, semantic, and cross-modal evidence. At query time, retrieved semantic memory and visual evidence support response polling as memory continues to evolve with new observations. Extensive experiments demonstrate state-of-the-art performance, while ablations validate each component of our proposed approach.

STATEMENT ON THE USE OF ARTIFICIAL INTELLIGENCE

We used generative AI tools to assist with language polishing, manuscript organization, literature search, and the interpretation of experimental results provided by the authors. These tools also assisted with LaTeX formatting and figure annotations. Claude was used to modify parts of the evaluation code. The method and experimental design were developed by the authors without generative AI assistance. All AI-assisted content and code were manually reviewed and verified by the authors. We take full responsibility for the final content of this paper, including its claims, results, and associated artifacts.

ETHICS STATEMENT

We have read and adhere to the ICLR Code of Ethics. Our evaluation uses publicly available video understanding benchmarks, including StreamingBench, OVO-Bench, Video-MME, and LongVideoBench. All datasets and pretrained models are used in accordance with their licenses. EviStream uses Qwen3.5-9B as its backbone and Qwen3-Embedding-0.6B for retrieval, and may inherit biases and factual errors from these pretrained components. The framework aims to improve memory efficiency and preserve evidence for interactive video understanding. EviStream operates without additional training and reduces peak GPU memory usage and time to first token relative to the backbone in the evaluated efficiency setting. We declare no competing interests.

REPRODUCIBILITY STATEMENT

We will release the source code for EviStream, together with the evaluation scripts and configuration files, upon publication to support reproducibility. The release will include implementations of Incremental Semantic Memory, Memory-Aligned Evidence Keyframe Selection, and Continual Evidence-Polling Reasoning. Our framework uses the publicly available Qwen3.5-9B Qwen Team (2026) backbone and Qwen3-Embedding-0.6B Zhang et al. (2025b) retrieval encoder without additional training. All experiments use publicly available benchmarks. Offline evaluation uses the lmms-eval Zhang et al. (2024) framework.

REFERENCES

Shehreen Azad, Vibhav Vineet, and Yogesh S Rawat. Streamready: Learning what to answer and when in long streaming videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 40494–40504, 2026.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 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.

Xueyi Chen, Keda Tao, Kele Shao, and Huan Wang. Streamingtom: Streaming token compression for efficient video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24675–24685, 2026.

Gordon V Cormack, Charles LA Clarke, and Stefan Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pp. 758–759, 2009.

Shangzhe Di, Zhelun Yu, Guanghao Zhang, Haoyuan Li, Hao Cheng, Bolin Li, Wanggui He, Fangxun Shu, and Hao Jiang. Streaming video question-answering with in-context video kv-cache retrieval. In *International Conference on Learning Representations*, volume 2025, pp. 42115–42127, 2025.

Xin Ding, Hao Wu, Yifan Yang, Shiqi Jiang, Qianxi Zhang, Donglin Bai, Zhibo Chen, and Ting Cao. Streammind: Unlocking full frame rate streaming video dialogue through event-gated cognition. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13448–13459. IEEE, 2025.

Mingkang Dong, Muxin Pu, Jie Li, Bohan Guo, Songruo Chen, Bin Ren, Xu Zheng, Chen Zhao, Tianwen Qian, Mohamed Elhoseiny, et al. Objectstream: Latent objects as memory anchors for streaming video understanding. *arXiv preprint arXiv:2607.28312*, 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, 2025.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Hongxiang Jiang, Zengrui Ge, Guo Chen, Qixiong Wang, Jile Jiao, Xuetao Feng, Yuan Wang, and Yan Wang. Memento: Toward an all-day proactive assistant for ultra-long streaming video. In *International Conference on Learning Representations*, volume 2026, pp. 35267–35281, 2026.

Junho Kim, Hosu Lee, James M Rehg, Minsu Kim, and Yong Man Ro. Stride: When to speak meets sequence denoising for streaming video understanding. *arXiv preprint arXiv:2603.27593*, 2026.

Zhijia Liang, Jiaming Li, Weikai Chen, Yanhao Zhang, Haonan Lu, and Guanbin Li. Oasis: On-demand hierarchical event memory for streaming video reasoning. *arXiv preprint arXiv:2604.17052*, 2026.

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.

Yuxin Liu, Peiqin Zhuang, and Yali Wang. Streamems: Streaming video understanding with self-evolving memory scheme for vision-language models. *arXiv preprint arXiv:2608.27881*, 2026a.

Zikang Liu, Longteng Guo, Handong Li, Ru Zhen, Xingjian He, Ruyi Ji, Xiaoming Ren, Yanhao Zhang, Haonan Lu, and Jing Liu. Thinking in streaming video. In *European Conference on Computer Vision*, pp. 21–38. Springer, 2026b.

Ke Ma, Jiaqi Tang, Bin Guo, Xueting Han, Ruonan Xu, Qingfeng He, Ziheng Wang, Xu Wang, Qifeng Chen, Zhiwen Yu, et al. Response-g1: Explicit scene graph modeling for proactive streaming video understanding. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 44139–44153, 2026.

Le Tu Ngoc Minh, Jinyeong Lim, and Dongsu Han. Protokv: Streaming video understanding under delayed query with summary-state memory. *arXiv preprint arXiv:2606.26762*, 2026.

Zhenyu Ning, Guangda Liu, Qihao Jin, Chengwei Li, Wenchao Ding, Minyi Guo, and Jieru Zhao. Livevlm: Efficient online video understanding via streaming-oriented kv cache and retrieval. *arXiv preprint arXiv:2505.15269*, 2025.

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, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Shuangrui Ding, Dahua Lin, and Jiaqi Wang. Streaming long video understanding with large language models. *Advances in Neural Information Processing Systems*, 37:119336–119360, 2024.

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.

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

Yansong Shi, Qingsong Zhao, Tianxiang Jiang, Xiangyu Zeng, Yi Wang, and Limin Wang. RIVER: A real-time interaction benchmark for video llms. In Carl Vondrick, Bharath Hariharan, Colin Raffel, Lerrel Pinto, Diyi Yang, and Aleksandra Faust (eds.), *The Fourteenth International Conference on Learning Representations, ICLR 2026, Rio de Janeiro, Brazil, April 23-27, 2026*. proceedings.iclr.cc, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/1022661f3f43406065641f16ce25eafa-Abstract-Conference.html.

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

Yiyu Wang, Xuyang Liu, Xiyan Gui, Xinying Lin, Boxue Yang, Chenfei Liao, Tailai Chen, and Linfeng Zhang. Accelerating streaming video large language models via hierarchical token compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18523–18533, 2026a.

Yueqian Wang, Songxiang Liu, Disong Wang, Nuo Xu, Guanglu Wan, Huishuai Zhang, and Dongyan Zhao. Mmduet2: Enhancing proactive interaction of video mllms with multi-turn reinforcement learning. In *International Conference on Learning Representations*, volume 2026, pp. 23257–23270, 2026b.

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, 2024a.

Shiwei Wu, Joya Chen, Kevin Qinghong Lin, Qimeng Wang, Yan Gao, Qianli Xu, Tong Xu, Yao Hu, Enhong Chen, and Mike Zheng Shou. Videollm-mod: Efficient video-language streaming with mixture-of-depths vision computation. *Advances in Neural Information Processing Systems*, 37:109922–109947, 2024b.

Junbin Xiao, Jiajun Chen, Tianxiang Sun, Xun Yang, and Angela Yao. Mukv: Multi-grained kv cache compression for long streaming video question-answering. *arXiv preprint arXiv:2605.22269*, 2026.

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*, 2026.

Haomiao Xiong, Zongxin Yang, Jiazuo Yu, Yunzhi Zhuge, Lu Zhang, Jiawen Zhu, and Huchuan Lu. Streaming video understanding and multi-round interaction with memory-enhanced knowledge. In *International Conference on Learning Representations*, volume 2025, pp. 69332–69351, 2025.

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.

Tian Xueyun, Wei Li, Bingbing Xu, Heng Dong, Yuanzhuo Wang, and Huawei Shen. Roma: Real-time omni-multimodal assistant with interactive streaming understanding. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 23018–23039, 2026.

Yanlai Yang, Zhuokai Zhao, Satya Narayan Shukla, Aashu Singh, Shlok Kumar Mishra, Lizhu Zhang, and Mengye Ren. Streammem: Query-agnostic kv cache memory for streaming video understanding. *arXiv preprint arXiv:2508.15717*, 2025.

Zhenyu Yang, Kairui Zhang, Yuhang Hu, Bing Wang, Shengsheng Qian, Bin Wen, Fan Yang, Tingting Gao, Weiming Dong, and Changsheng Xu. Livestar: Live streaming assistant for real-world online video understanding. *Advances in Neural Information Processing Systems*, 38:31266–31304, 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.

Xueyang Yu, Cheng Shi, Yang Wang, and Sibei Yang. Eyes wide open: Ego proactive video-llm for streaming video. *Advances in Neural Information Processing Systems*, 38:13420–13463, 2026.

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.

Gengyuan Zhang, Mingcong Ding, Tong Liu, Yao Zhang, and Volker Tresp. Memory helps, but confabulation misleads: Understanding streaming events in videos with mllms. *arXiv preprint arXiv:2502.15457*, 2025a.

Haowei Zhang, Shudong Yang, Jinlan Fu, See Kiong Ng, and Xipeng Qiu. Hermes: Kv cache as hierarchical memory for efficient streaming video understanding. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8411–8430, 2026a.

Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, and Ziwei Liu. Lmms-eval: Reality check on the evaluation of large multimodal models, 2024. URL https://arxiv.org/abs/2407.12772.

Kairui Zhang, Zhenyu Yang, Bing Wang, Shengsheng Qian, and Changsheng Xu. Querystream: Advancing streaming video understanding with query-aware pruning and proactive response. In *The Fourteenth International Conference on Learning Representations*, 2026b.

Kecheng Zhang, Zongxin Yang, Mingfei Han, Haihong Hao, Yunzhi Zhuge, Changlin Li, Zhihui Li, Xiaojun Chang, et al. Progressive online video understanding with evidence-aligned timing and transparent decisions. In *International Conference on Learning Representations*, volume 2026, pp. 5092–5121, 2026c.

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

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

Yikai Zheng, Xin Ding, Yifan Yang, Shiqi Jiang, Hao Wu, Qianxi Zhang, Weijun Wang, Ting Cao, and Yunxin Liu. Em-garde: A propose-match framework for proactive streaming video understanding. *arXiv preprint arXiv:2603.19054*, 2026.

A Appendix

A.1 Streaming Video Understanding

Streaming video understanding requires models to process an ongoing video causally, maintain useful historical context, and answer queries at arbitrary timestamps without repeatedly revisiting the full stream. Recent works such as VideoLLM-online Chen et al. (2024), StreamingBench Lin et al. (2026), and OVO-Bench Niu et al. (2025) have established this setting beyond conventional offline long-video understanding, emphasizing continuous perception, temporal reasoning, and online interaction. More recently, the task has further evolved from *reactive* question answering toward *proactive* interaction, where models must decide both what to answer and *when* to respond. Dispider Qian et al. (2025), StreamMind Ding et al. (2025), Eyes Wide Open Yu et al. (2026), MMDuet2 Wang et al. (2026b), StreamReady Azad et al. (2026), and LiveStar Yang et al. (2026) explicitly study asynchronous perception, event-triggered reasoning, or response-versus-silence decisions. This trend highlights response timing as an increasingly important component of streaming intelligence.

A.2 Implementation Details

Models and evaluation. EviStream uses Qwen3.5-9B Qwen Team (2026) as its frozen backbone. We compare against representative trained and training-free streaming methods, including Em-Garde Zheng et al. (2026), FluxMem Xie et al. (2026), and QueryStream Zhang et al. (2026b). Offline evaluation uses the lmms-eval Zhang et al. (2024) framework. All experiments use eight NVIDIA A800 GPUs with 80 GB of memory each.

Visual processing and window construction. Videos are sampled at 1 FPS. Window summarization uses the default model-processor video preprocessing, while boundary encoding and historical keyframe inputs use an image-pixel budget of 448^2. The adaptive window bounds are $L_{\min} = 20$ and $L_{\max} = 30$ frames, with a visual-change threshold $\tau_{\mathrm{w}} = 0.75$. Adjacent windows share one boundary frame, which counts toward both length bounds. The frame that triggers a scene change enters the new window and becomes its anchor, while the preceding frame is inherited as the first frame of that window. Following adjacent-frame token comparison in Yao et al. (2025), we use $\tau_{\mathrm{p}} = 0.5$ for pruning.

Keyframe selection and retrieval. The MEKS weights are $\alpha = 0.55$, $\beta = 0.10$, and $\gamma = 0.35$, with $K_e = 16$ and $\tau_{\mathrm{e}} = 0.03$. Qwen3-Embedding-0.6B Zhang et al. (2025c) serves as the retrieval encoder E_{r} for memory search. MEKS uses the backbone's input embeddings for text, as described in Section 3.3. RRF selects the top $K_{\mathrm{r}} = 30$ memory lines, ANE uses radius $P = 1$, and the latest $N_{\mathrm{r}} = 2$ windows provide recent context. After retrieval and expansion, the historical visual context is limited to 128 keyframes, retaining the most recent frames when this limit is exceeded.

A.3 Inference Algorithms

Memory construction. Algorithm 1 summarizes the update for a completed adaptive window. The window-indexed records $\mathcal{D}$ contain formal memory increments and any provisional entry, while $\mathcal{K}[k]$ stores the keyframes associated with window k. Visual pruning preserves boundary frames and removes redundant tokens from the earlier frame after its successor arrives. The frozen model generates a raw delta T_k from the compressed window and committed history, with $\mathcal{M}_0 = \varnothing$. Adjacent exact deduplication produces the increment appended to history, while MEKS uses the raw T_k to select complementary visual evidence.

A.4 Dynamic Generation Budget

The maximum output length starts at 256 tokens. Before each subsequent formal window summary, the budget increases by $\Delta_B = 512$ tokens if the previous output reached 90% of the current limit. This rule accommodates longer deltas without a large fixed initial budget. Each formal summary conditions on the full committed history. Provisional summaries neither change this budget nor enter subsequent formal-summary prompts as committed history.

Algorithm 1 Formal memory construction in EviStream

Input Completed window W_k, projected tokens H_k, committed history $\mathcal{M}_{k-1}$
Previous raw delta T_{k-1} when $k > 1$, optional query q, stores $\mathcal{D}$ and $\mathcal{K}$
Output Updated history $\mathcal{M}_k$, raw delta T_k, updated stores $\mathcal{D}$ and $\mathcal{K}$

for each frame i in W_k **do**
  $\bar{h}_{k,i} \leftarrow J_k^{-1} \sum_{p=1}^{J_k} h_{k,i,p}$
end for
$(\widetilde{H}_k, \{R_{k,i}\}) \leftarrow \textsc{PruneEarlierTokens}(H_k, \tau_{\mathrm{p}})$
$T_k \leftarrow F_\theta(p_\Delta, \mathcal{M}_{k-1}, \widetilde{H}_k)$
if $k = 1$ **then**
  $\Delta_k \leftarrow T_k$
else
  $\Delta_k \leftarrow \mathrm{Dedup}_{\mathrm{adj}}(T_k, T_{k-1})$
end if
$\mathcal{M}_k \leftarrow \mathcal{M}_{k-1} \oplus \Delta_k$
$\mathcal{K}_k \leftarrow \mathrm{MEKS}(H_k, \{R_{k,i}\}, \{\bar{h}_{k,i}\}, T_k, q)$
$\mathcal{K}[k] \leftarrow \mathcal{K}_k$ with frame timestamps and window identifier k
$\mathcal{D}[k] \leftarrow \textsc{FormalRecord}(\Delta_k, k, \mathrm{Span}(W_k))$
  This assignment replaces any provisional record for window k.
Cache $e_i = E_{\mathrm{r}}(m_i)$ for each memory line m_i in $\mathcal{D}[k]$
return $\mathcal{M}_k, T_k, \mathcal{D}, \mathcal{K}$

A.5 Hybrid Retrieval

After each memory update, the retrieval encoder E_{r} embeds each memory line m_i, and its vector e_i is cached in RAM. These cached vectors avoid repeated historical encoding at query time. We construct a temporary retrieval view from accumulated deltas and any provisional entry. Memory lines are ranked by both dense semantic retrieval and lexical retrieval. The dense branch uses E_{r} to match semantically related descriptions against cached memory vectors, while the lexical branch captures explicit term overlap. The two rankings are combined through reciprocal rank fusion (RRF) Cormack et al. (2009)

$$s_{\mathrm{sem}}(q, m_i) = E_{\mathrm{r}}(\pi \oplus q)^\top e_i,$$
$$s_{\mathrm{rrf}}(m_i) = \frac{1}{\kappa + r_{\mathrm{lex}}(i)} + \frac{1}{\kappa + r_{\mathrm{sem}}(i)}. \tag{A1}$$

where π denotes the query instruction, $r_{\mathrm{lex}}(i)$ and $r_{\mathrm{sem}}(i)$ are the ranks of m_i in the lexical and semantic retrieval branches, respectively, and κ is the rank-smoothing constant. The top-K_{r} memory lines determine the relevant windows.

Before ranking, query-time fuzzy deduplication operates on a temporary view of historical lines. Within the same paragraph, lines with similarity at least 0.92 are merged while retaining the earliest occurrence; short sentences containing numbers are protected. This operation leaves original deltas and committed history unchanged. It is distinct from adjacent exact deduplication during memory construction. ANE expands matched windows as specified in Equation 8.

A.6 Provisional Memory Updates

Observations in an ongoing window may leave the live buffer before formal consolidation. Before each poll, EviStream checks

$$n_t - c_t - \ell_t \geq \tau_{\mathrm{prov}}, \tag{A2}$$

where n_t counts all arrived frames in the current window, c_t is the coverage count of its latest provisional summary, and ℓ_t is the live-buffer length. The configurable threshold τ_{prov} expresses a duration in sampling steps. When the condition holds, the model summarizes all arrived frames, including the latest live frames, and updates the coverage to $c_t = n_t$. The provisional record shares the current window identifier and enters the next retrieval through the mandatory latest-two-window fallback. It does not run MEKS, change committed history, or alter the formal output budget.

When the window closes, formal summarization uses committed history to generate T_k. Its locally deduplicated increment is appended to history, and the formal retrieval record replaces the provisional record under the same identifier. Earlier committed deltas remain unchanged. After query

Table 6: Complete evaluation results on OVO-Bench across Real-Time Visual Perception, Backward Tracing, and Forward Active Responding.

Method	# Frames	Real-Time Visual Perception							Backward Tracing				Forward Act. Res.				Overall
		OCR	ACR	ATR	STU	FPD	OJR	Avg.	EPM	ASI	HLD	Avg.	REC	SSR	CRR	Avg.	Avg.
Human	-	94.0	92.6	94.8	92.7	91.1	94.0	93.2	92.6	93.0	91.4	92.3	95.5	89.7	93.6	92.9	92.8
Proprietary Models																	
Gemini 1.5 Pro	1 fps	87.3	67.0	80.2	54.5	68.3	67.4	70.8	68.6	75.7	52.7	62.3	35.5	74.2	61.7	57.2	65.3
GPT-4o	64	69.1	65.1	65.5	50.0	68.3	63.7	63.6	49.8	71.0	55.4	58.7	27.6	73.2	59.4	53.4	58.6
Online Video MLLMs (Training)																	
Flash-VStream	1 fps	25.5	32.1	29.3	33.7	29.7	28.8	29.9	36.4	33.8	5.9	25.4	5.4	67.3	60.0	44.2	33.2
Dispider	1 fps	57.7	49.5	62.1	44.9	61.4	51.6	54.6	48.5	55.4	4.3	36.1	18.1	37.4	48.8	34.7	41.8
TimeChat-Online	1 fps	74.5	48.6	68.1	48.3	69.3	59.8	61.4	56.9	64.9	11.8	44.5	31.8	38.5	40.0	36.8	47.6
StreamForest	1 fps	68.5	53.2	71.6	47.8	65.4	60.9	61.2	58.9	64.9	32.3	52.0	32.8	70.6	57.1	53.5	55.6
Em-Garde	1 fps	76.5	61.5	65.5	47.2	70.3	57.1	63.0	48.2	63.5	45.2	52.2	–	–	–	–	–
Response-G1	0.2–1 fps	90.1	74.3	75.9	59.6	69.3	71.7	73.6	55.6	66.9	33.9	52.1	41.9	71.1	61.7	58.2	61.3
Online Video MLLMs (Training-Free)																	
FreshMem	–	77.2	60.6	70.7	56.7	63.4	70.7	66.7	57.2	56.8	46.8	54.0	23.9	**72.7**	55.0	48.3	56.3
FluxMem	1 fps	81.2	59.6	70.7	53.4	75.2	63.0	67.2	–	–	–	–	–	–	–	–	53.3
HERMES	1 fps	85.3	65.2	74.4	54.0	68.3	66.8	69.0	44.8	64.2	39.8	49.6	–	–	–	–	–
QueryStream	1 fps	74.5	47.7	70.7	46.6	71.3	57.6	61.4	54.2	63.5	8.6	42.1	33.2	43.1	40.8	39.0	47.5
STRIDE	1 fps	76.5	64.2	79.3	61.2	73.3	63.6	69.7	57.2	**72.3**	14.0	47.8	46.4	63.1	**69.6**	59.7	59.1
Qwen3.5-9B†	1 fps	74.5	64.2	75.0	62.4	**79.2**	63.6	69.8	**59.6**	70.3	8.6	46.2	**50.4**	64.4	65.0	**59.9**	58.6
+ EviStream	1 fps	**91.3**	**76.2**	**81.9**	**64.6**	73.3	**73.4**	**76.8**	54.9	62.8	**57.5**	**58.4**	39.2	62.7	63.3	55.1	**63.4** (↑4.8%)

arrival, the live buffer grows from one to three frames and then retains the latest three at their original resolution.

A.7 Additional Results on OVO-Bench

Table 6 presents the complete OVO-Bench results across Real-Time Visual Perception, Backward Tracing, and Forward Active Responding (FAR). EviStream improves the overall score of Qwen3.5-9B from 58.6 to 63.4, with gains in Real-Time Visual Perception and Backward Tracing.

FAR evaluates proactive responses through offline simulation at different video timestamps. On FAR, EviStream outperforms most evaluated models but achieves 55.1 compared with 59.9 for the backbone. We attribute this result to the offline simulation used by FAR, where the model receives the full video preceding each timestamp and the query for a single decision. This setting resembles offline long-video understanding and thus provides more complete information.

A.8 MEKS Edge Conventions

MEKS runs only after formal summary generation. Before a query is available, selection uses temporal novelty and text-gap feedback. Boundary frames remain keyframe candidates and receive zero temporal novelty. Frames with no retained patches receive zero query-evidence and novelty scores, while their text-gap score still uses the cached pre-pruning full-frame mean. The novelty mean includes interior frames with zero retention. The text-gap mean includes every frame in the window with equal weight and compares against the current raw delta, before exact line deduplication. Text representations use the base LLM tokenizer and input embedding lookup, without a separate MEKS text encoder. The retrieval encoder E_{r} serves only memory search.

A.9 Effect of Complementarity-Aware Evidence Memory.

To assess the individual contributions of query evidence, temporal novelty, and text-gap feedback to evidence-aware keyframe selection, we selectively enable or remove each score, as shown in Table 7. Including S_{qe} retains a larger proportion of complementary frames within each window. On Forward Active Responding in OVO-Bench, the S_{qe}+S_{nov} configuration achieves 54.7, exceeding the configuration with only S_{nov}. This result indicates that query-guided evidence retention from the recent video stream is necessary for proactive online perception, even when historical memory frames are compressed. On Backward Tracing, which depends on fine-grained historical evidence, S_{nov}+S_{gap} achieves 58.2, outperforming either S_{nov} or S_{gap} individually. Retention based on frame–text dis-

crepancy and temporal novelty thus has complementary effects, improving the preservation of visual content not covered by semantic memory. Combining all three scores achieves the highest accuracy of 63.4 while retaining 16.5% of all historical frames. The three selection signals jointly maintain stable performance.

Table 7. Ablation of the complementary scoring signals in MEKS. Drop Ratio measures the proportion of historical frames removed by keyframe selection.

MEKS part				OVO-Bench			Score
qe	nov	gap	Drop Ratio	RTVP.	BT.	FAR.	Avg.
✗	✗	✗	0	76.6	**58.5**	46.4	60.5
✓	✗	✗	↓ 83.0%	76.4	58.4	55.0	63.3
✗	✓	✗	↓ 99.1%	76.1	57.2	46.9	60.1
✗	✗	✓	↓ 85.3%	73.8	57.5	53.0	61.4
✓	✓	✗	↓ 83.6%	76.5	58.1	54.7	63.1
✓	✗	✓	↓ 83.7%	76.6	57.7	**55.4**	63.2
✗	✓	✓	↓ 98.9%	76.6	58.2	47.2	60.7
✓	✓	✓	↓ 83.5%	**76.8**	58.4	55.1	**63.4**

A.10 Analysis of the Number of Retrieved Keyframes for EviStream

To assess how the number of keyframes retrieved for QA affects performance, we compare accuracy and truncation ratios under different visual evidence budgets on OVO-Bench. This analysis evaluates the budget after text retrieval and temporal neighborhood expansion (ANE). As shown in Figure 4, a maximum of 32 keyframes truncates most retrieval results to visual evidence from short-term memory, yielding 62.98%. As the keyframe budget increases, the proportion of candidate keyframes removed by truncation steadily decreases. Accuracy peaks at 63.42% with 128 frames and then declines, indicating that most retrieved visual evidence can already be retained in full.

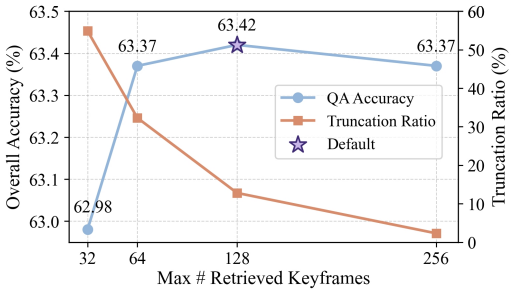

Figure 4. Historical-keyframe budget on OVO-Bench. Truncation is the fraction of candidate keyframes discarded after retrieval and expansion. The star marks the default budget.