RESIGHT: RECONSTRUCTING FINE-GRAINED VISUAL EVIDENCE FOR STREAMING VIDEO UNDERSTANDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Understanding a changing world requires reasoning about evidence that is no longer visible. Streaming video understanding demands this ability under bounded memory, yet observations arrive before their relevance to future questions is known. We introduce **ReSight**, which reconstructs fine-grained visual evidence from a compact persistent state. Our key insight is to give memory a stable visual reconstruction target while learning separately how recovered evidence should influence an answer. ReSight maintains a fixed-size visual field through recursive least-squares updates in fixed temporal and appearance coordinates. At query time, a lightweight reader reconstructs spatially resolved features and pools them into a bounded visual context, deferring question-dependent interpretation to the decoder. To account for imperfect reconstruction, asymmetric attention preserves a perception reference within a shared decoder, and a learned signed gate regulates the deviation proposed by history-informed reasoning. This separates preserving observations from deciding their influence, without replaying past frames or optimizing parameters at query time. Trained on **only 10% of StreamTTT's training data**, ReSight surpasses StreamTTT on seven of eleven task metrics across OVO-Bench and StreamingBench and matches it on one. It improves action sequence identification by 10.1 percentage points and sequential question answering by 7.20 points, while raising real-time visual understanding accuracy by 2.76 points. Moreover, ReSight reduces per-frame ingestion time by 19% and persistent state storage by 57% relative to StreamTTT. These results highlight the effectiveness of reconstructive memory with higher data efficiency and inference efficiency.

1 INTRODUCTION

Understanding a changing world requires connecting what is visible with what has been seen before. For embodied robots and interactive assistants, observations provide only a partial view: objects disappear, viewpoints shift, and actions unfold across time. This motivates *streaming video understanding*, where a large multimodal model (LMM) receives frames causally and answers questions about the present scene and observed past (Niu et al., 2025; Lin et al., 2026). The challenge is to sustain this ability with bounded memory when observations arrive before their future relevance is known. In Figure 1(a), identifying the yellow object on the couch requires a local visual detail encountered well before the question arrives. By then, the object is no longer visible. Storing every visual token makes memory and attention costs grow with the stream, while committing early to a compact description risks discarding distinctions that future questions require. Effective memory must therefore preserve evidence for later interpretation without replaying discarded frames.

Existing approaches offer four complementary storage strategies (Figure 1(b)). *Cache-based methods* retain directly accessible KV states but lose older entries through eviction (Xu et al., 2026). *Compressive methods* consolidate features or events, trading detail for coverage (Zeng et al., 2025; Xie et al., 2026). *Retrieval-based methods* select evidence from an archive, with recall constrained by retention and retrieval success (Di et al., 2025; Fan et al., 2026). *Recurrent and fast-weight methods* accumulate history in evolving states accessed through learned readouts (Chen et al., 2026; Liu et al., 2026). Beyond storage, recent agent-memory research emphasizes reconstructive use: MRAgent assembles memory through reasoning over an associative graph (Ji et al., 2026), while MemHarness adapts retrieved textual experiences into guidance for the current state (Wu et al.,

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.

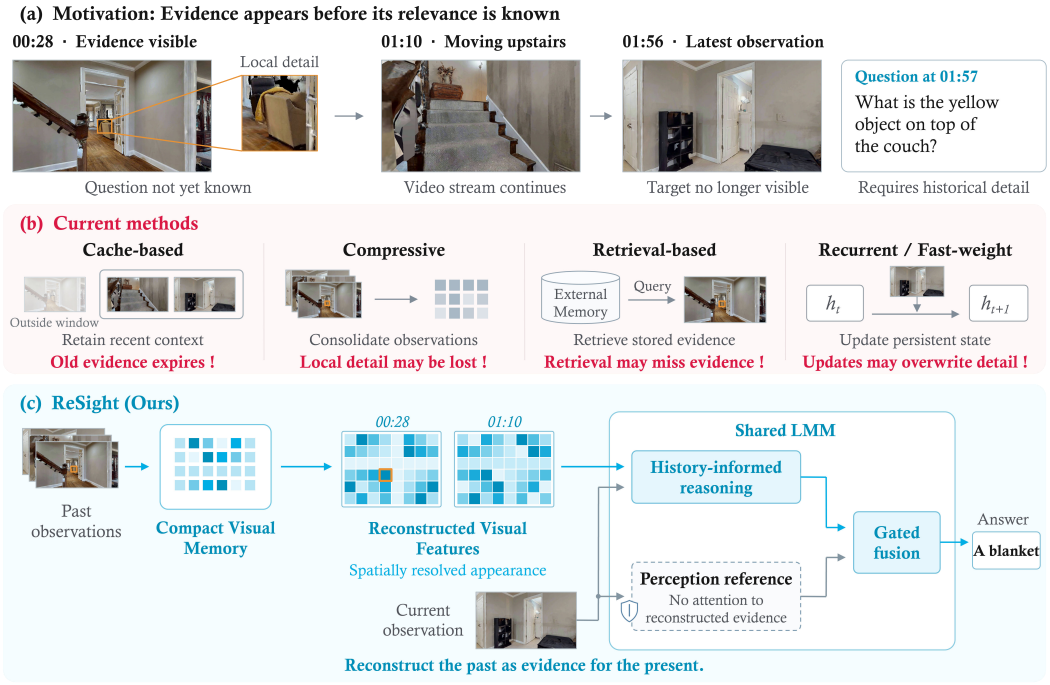


Figure 1: **Reconstructing the past as evidence for the present.** (a) A later question requires a visual detail no longer visible. (b) Existing memory strategies face different retention and access limits. (c) ReSight reconstructs visual features for historical reasoning, with gated fusion regulating their influence relative to a protected perception reference.

2026). Streaming vision poses a distinct reconstruction problem: the required evidence may lie in local appearance and spatial structure that a textual account never expressed. Reorganizing recorded descriptions cannot recover such omitted distinctions. For compact visual memory, the crucial question is therefore what its state should reconstruct so that observations remain useful across unanticipated questions.

We introduce **ReSight**, built on the principle of *reconstructing past observations as evidence for present reasoning* (Figure 1(c)). Its key design is to give persistent memory a fixed visual reconstruction target while leaving question-dependent interpretation to the reader. Storage approximates spatially resolved appearance in stable coordinates, and the decoder learns how the recovered evidence should inform each answer. Concretely, ReSight maintains a fixed-size visual field through recursive least-squares updates in fixed temporal and appearance coordinates. At query time, evenly spaced probes reconstruct historical feature grids, which a lightweight reader refines and average-pools into a bounded visual prefix. Both probing and pooling are question-independent, while the decoder interprets the resulting evidence alongside the question and available observation context. This explicit reconstruction interface allows the reader to learn without redefining the meaning of stored observations. Memory size and reconstructed context remain bounded as the stream grows, and reconstruction requires no access to discarded frames.

Making historical evidence available also requires controlling its influence: reconstruction is approximate, and past observations may describe a state that has since changed. Additional history can consequently improve recall while weakening current-scene perception (Shen et al., 2026). ReSight therefore retains a perception reference that reconstructed evidence cannot directly alter. One shared decoder computes perception and history-informed representations under the same question and answer prefix, separated by an asymmetric attention mask. A learned signed gate regulates their difference at each prediction, allowing history to strengthen or weaken the reference distribution's preferences. Auxiliary supervision trains historical reasoning and penalizes increases in answer loss relative to the reference on real-time examples. Together, reconstruction and protected perception distinguish preserving an observation from deciding how it should influence an answer. At de-

ployment, observations update persistent memory, while queries read from it without optimizing parameters or writing answers back into memory.

We evaluate ReSight on OVO-Bench and StreamingBench using **only 10% of StreamTTT's training data**. Despite this smaller training set, ReSight outperforms StreamTTT on seven of eleven individual task metrics. It achieves 69.74% on OVO-Bench, 84.08% on real-time visual understanding, and 56.80% on sequential question answering, exceeding StreamTTT by 0.74, 2.76, and 7.20 percentage points, respectively. Historical reasoning also benefits, with a 10.1-point improvement on action sequence identification. Complementary efficiency measurements show 19% lower per-frame ingestion time and 57% less persistent state storage than StreamTTT. Together, these results support reconstructive memory as an effective approach to streaming perception and historical reasoning, combining strong performance under limited training data with efficient memory maintenance. Our contributions are threefold:

- We introduce ReSight to reconstruct the past as evidence for present reasoning. It separates observation preservation from historical influence through visual reconstruction.
- A fixed-coordinate visual field is paired with a protected perception reference. Analytic updates preserve reconstructible visual features, while asymmetric attention and signed fusion control their influence on answers.
- Experiments demonstrate high data efficiency and efficient memory maintenance. With only 10% of StreamTTT's training data, ReSight improves seven of eleven task metric with lower ingestion time and persistent storage costs.

2 METHOD

ReSight reconstructs the past as evidence for the present. Its design separates two decisions: what observations to preserve before a question arrives, and how history should influence the answer afterward. A fixed-size visual field approximates observations in stable coordinates, and a trained reader converts its reconstructions into a bounded visual context. A shared decoder then interprets this context relative to a perception reference that reconstructed evidence cannot alter directly. Figure 2 illustrates this separation between *visual preservation* and *historical influence*.

2.1 FIXED-COORDINATE VISUAL MEMORY

Give memory a stable reconstruction target. Let n count the completed temporal patches observed before a question q arrives, and let τ_i be the mean frame timestamp of patch i. The frozen visual encoder produces merger features $v_{i,p} \in \mathbb{R}^d$, where p indexes spatial locations and d is the language decoder width. We apply RMS normalization N, project onto a fixed orthonormal basis $U \in \mathbb{R}^{d \times d_v}$ fitted on training inputs, and append each feature's log-RMS magnitude. Bilinear resampling to a $g \times g$ grid and flattening yield $y_i \in \mathbb{R}^D$, with $D = g^2(d_v + 1)$: d_v appearance channels and one scale channel per cell. A fixed Fourier map $\phi : \mathbb{R} \to \mathbb{R}^{d_k}$ assigns each timestamp a unit-norm address $k_i = \phi(\tau_i)$ of width d_k. The visual field $M_n \in \mathbb{R}^{d_k \times D}$ fits these associations:

$$M_n = \arg\min_M \left\{ \frac{1}{2} \sum_{i=1}^{n} \|M^\top k_i - y_i\|_2^2 + \frac{\lambda}{2} \|M\|_F^2 \right\}, \tag{1}$$

where $\lambda > 0$ is the ridge strength, and $\|\cdot\|_2$ and $\|\cdot\|_F$ denote Euclidean and Frobenius norms. Both addresses and targets remain fixed as the reader learns, so storage continues to approximate the same visual quantities.

Update from reconstruction residuals. Maintain the inverse Gram matrix $P_n = (\lambda I_{d_k} + \sum_{i=1}^{n} k_i k_i^\top)^{-1}$, where I_r denotes the size-r identity. For b new patches, stack transposed addresses and targets as $K \in \mathbb{R}^{b \times d_k}$ and $Y \in \mathbb{R}^{b \times D}$. From the pre-update state (M, P), recursive least squares gives

$$\begin{aligned} A &= PK^\top (I_b + KPK^\top)^{-1}, \\ M^+ &= M + A(Y - KM), \qquad P^+ = P - AKP, \end{aligned} \tag{2}$$

where $A \in \mathbb{R}^{d_k \times b}$ is the regression gain and $+$ marks the updated state. Initialized with $M_0 = 0$ and $P_0 = \lambda^{-1} I_{d_k}$, this update solves Equation 1 in exact arithmetic without retaining old targets.

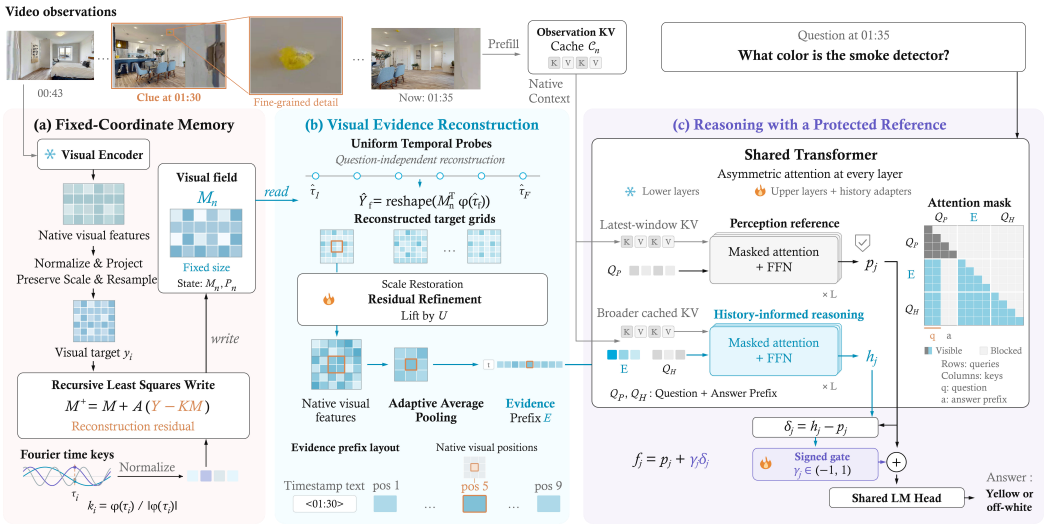


Figure 2: **Overview of ReSight.** (a) A fixed-size visual field stores observations in stable coordinates. (b) Historical features are reconstructed and average-pooled into evidence prefix. (c) Asymmetric attention protects a perception reference, while signed fusion regulates historical influence.

The residual $Y - KM$ supplies unexplained content. P accounts for previously observed addresses (Appendix A.2). Alongside the field, a key-value cache $\mathcal{C}_n$ retains instruction tokens and recent observation states. Its token budget L and the field dimensions are independent of video length.

2.2 Visual Evidence Reconstruction

Recover spatial features across the observed interval. After at least one observation window, probe $F_n = \min(F, n)$ evenly spaced times $\widehat{\boldsymbol{\tau}} = (\widehat{\tau}_1, \ldots, \widehat{\tau}_{F_n})$ in $[\tau_1, \tau_n]$, where F is the temporal probe budget. Each probe reconstructs a grid:

$$\widehat{Y}_f = \text{reshape}_{g\times g\times(d_v+1)}\big(M_n^\top \phi(\widehat{\tau}_f)\big), \qquad f = 1, \ldots, F_n. \tag{3}$$

Write cell p as appearance coordinates $\widehat{z}_{f,p} \in \mathbb{R}^{d_v}$ and log scale $\widehat{s}_{f,p} \in \mathbb{R}$. Scale restoration divides $\widehat{z}_{f,p}$ by $\|\widehat{z}_{f,p}\|_2/\sqrt{d}$ and multiplies by $\exp(\widehat{s}_{f,p})$, with numerical safeguards detailed in Appendix A.3. Let $x \in \mathbb{R}^{F_n\times g\times g\times d_v}$ collect the restored coordinates. A residual decoder $\mathcal{D}_\eta$, with learned parameters η, produces native-width features $r_{f,p} = U\mathcal{D}_\eta(x, \widehat{\boldsymbol{\tau}})_{f,p} \in \mathbb{R}^d$. Initialized to the identity, it refines spatial neighborhoods and channels and mixes differences between temporally nearby probes. This reconstructs visual features directly from memory without replaying frames.

Bound context without predicting future relevance. Adaptive average pooling maps these features to B visual tokens in a fixed spatio-temporal layout. For bin $\mathcal{B}_\ell$,

$$e_\ell = \frac{1}{|\mathcal{B}_\ell|} \sum_{(f,p)\in\mathcal{B}_\ell} r_{f,p} \in \mathbb{R}^d, \qquad \ell = 1, \ldots, B, \tag{4}$$

where $|\mathcal{B}_\ell|$ is its number of input cells. Pooling is parameter-free and question-independent in training and inference. Bins may overlap, so B counts slots rather than distinct observations. The tokens form a temporary prefix E, with temporal-bin means rendered as timestamp text and spatial layout encoded by native visual positions. Thus B controls visual context independently of memory width d_k and cache budget L, while the decoder determines relevance to the question.

2.3 Reasoning with a Protected Perception Reference

Separate evidence interpretation from the reference. Reconstructed evidence is approximate, so it should not directly overwrite the perception used to assess it. One shared Transformer processes

$[Q_P; E; Q_H]$, where $[\,;\,]$ denotes concatenation and Q_P, Q_H copy the same question prompt and answer prefix. An asymmetric mask lets perception positions read instruction tokens, the still-visible suffix of the latest observation window, and their own causal text, but no historical positions. Historical positions in E and Q_H access the broader observation context, recognized question tokens in Q_P and their own causal sequence, but cannot read answer tokens from Q_P. Low-rank adapters on selected query, value, and output projections act only at historical positions. Separate positional cursors prevent E from shifting perception positions.

For answer token a_j at step j, let $a_{<j}$ denote preceding answer tokens and $p_j, h_j \in \mathbb{R}^d$ the two final normalized hidden states predicting a_j from the same $q, a_{<j}$. At fixed parameters, cache, positions, and text, the mask makes p_j invariant to reconstructed evidence (Appendix A.4). This protection holds within the query: cached states can encode earlier context, and training updates shared parameters.

Learn history's influence relative to perception. The historical deviation $\delta_j = h_j - p_j$ defines a direction for revising the reference. The fused representation $f_j \in \mathbb{R}^d$ is

$$f_j = p_j + \gamma_j \delta_j, \qquad \gamma_j = \tanh\big(w_g^\top [\mathsf{N}(p_j); \mathsf{N}(\delta_j)] + b_g\big)\,, \tag{5}$$

where $w_g \in \mathbb{R}^{2d}$ and $b_g \in \mathbb{R}$ are learned gate parameters. Their zero initialization gives $f_j = p_j$; the signed scalar $\gamma_j \in (-1, 1)$ subsequently controls the strength and direction of the deviation. With shared linear output head $W_{\text{out}} \in \mathbb{R}^{|\mathcal{V}|\times d}$ over vocabulary $\mathcal{V}$, let $\pi_j^P, \pi_j^H, \pi_j^F$ be the softmax distributions of $W_{\text{out}} p_j$, $W_{\text{out}} h_j$, and $W_{\text{out}} f_j$, respectively. For token $v \in \mathcal{V}$,

$$\pi_j^F(v) \;\propto\; \pi_j^P(v) \left(\frac{\pi_j^H(v)}{\pi_j^P(v)} \right)^{\gamma_j}. \tag{6}$$

Fusion therefore modulates perception by the relative support supplied by historical reasoning. This identity does not assume calibrated confidence. Each query uses a disposable cache copy: E is inserted once, and each decoding step feeds the selected token to both paths in one packed call. Neither reconstructed tokens nor answers enter persistent memory.

2.4 Training Objective

Zero-gated initialization preserves perception but initially blocks the fused loss from training history-specific modules. We therefore supervise the historical path directly. For an answer of J tokens and branch $R \in \{F, P, H\}$, define $\mathcal{L}_R = -J^{-1} \sum_{j=1}^{J} \log \pi_j^R(a_j)$. The turn loss is

$$\mathcal{L} = \mathcal{L}_F + \lambda_H \mathcal{L}_H + c\,(\lambda_P \mathcal{L}_P + \lambda_\Delta [\mathcal{L}_F - \mathrm{sg}(\mathcal{L}_P)]_+)\,, \tag{7}$$

where $\lambda_H, \lambda_P, \lambda_\Delta \geq 0$ are auxiliary weights, $c \in \{0, 1\}$ indicates a real-time annotation, $[u]_+ = \max(u, 0)$, and sg stops gradients. Historical supervision trains the reader while the gate is closed. On real-time examples, perception supervision maintains the reference, and the hinge penalizes fusion when its mean answer loss exceeds that reference. Detaching $\mathcal{L}_P$ prevents the hinge from rewarding a worse reference. No evidence-location or branch-selection labels are required.

We first train reconstruction, historical adapters, and fusion with the backbone fixed, then jointly tune them with the upper language-model layers. Losses are averaged over turns within each annotation and then over annotations. Joint training retains gradients through the observation cache. The visual encoder, basis, temporal addresses, analytic writer, and average pooling remain fixed. At deployment, observations update persistent state, while answering requires no optimization of model parameters or evidence tokens.

3 Experiments

Our experiments test whether reconstructive memory preserves useful history, improves answers without compromising perception, and supports continuous observation with bounded storage.

3.1 Experimental Setup

Benchmarks and protocols. We evaluate real-time perception (RP) and backward tracing on OVO-Bench (OVO) (Niu et al., 2025), and real-time visual understanding (RTVU) and sequential QA

Table 1: **Streaming benchmark results.** Bold/underline indicate the best/second-best model scores, excluding humans. StreamingBench Avg. averages RTVU and SQA when both are available. "–": unreported. Input settings follow Appendix B.

		StreamingBench			OVO-Bench											
					Real-Time Perception							Backward Tracing				Overall
Model	Input	RTVU	SQA	Avg.	OCR	ACR	ATR	STU	FPD	OJR	Avg.	EPM	ASI	HLD	Avg.	Avg.
Human	–	91.46	95.00	93.23	94.0	92.6	94.8	92.7	91.1	94.0	93.2	92.6	93.0	91.4	92.3	92.77
Offline Video LMMs																
Qwen2.5-VL-7B	1 fps	73.31	–	–	67.8	55.1	67.2	42.1	66.3	60.9	59.9	51.5	58.8	23.7	44.7	52.28
LLaVA-OneVision-7B	32	71.12	30.00	50.56	66.4	57.8	73.3	53.4	71.3	62.0	64.0	54.2	55.4	21.5	43.7	53.85
InternVL2-8B	16	63.72	42.80	53.26	67.1	60.6	63.8	46.1	68.3	56.5	60.4	48.2	57.4	24.7	43.4	51.90
LLaVA-Video-7B	64	–	–	–	69.1	58.7	68.8	49.4	74.3	59.8	63.4	56.2	57.4	7.5	40.4	51.86
Qwen2-VL-7B	64	69.04	44.40	56.72	69.1	53.2	63.8	50.6	66.3	60.9	60.7	44.4	66.9	34.4	48.6	54.62
LongVU-7B	1 fps	–	–	–	55.7	49.5	59.5	48.3	68.3	63.0	57.4	43.1	66.2	9.1	39.5	48.45
Online / Streaming LMMs																
VideoLLM-online-8B	2 fps	35.99	30.80	33.40	8.1	23.9	12.1	14.0	45.5	21.2	20.8	22.2	18.8	12.2	17.7	19.26
Flash-VStream-7B	1 fps	23.23	26.80	25.02	24.2	29.4	28.5	33.7	25.7	28.8	28.4	39.1	37.2	5.9	27.4	27.90
Dispider-7B	1 fps	67.63	34.80	51.22	57.7	49.5	62.1	44.9	61.4	51.6	54.6	48.5	55.4	4.3	36.1	45.35
TimeChat-Online-7B	1 fps	75.28	–	–	75.2	46.8	70.7	47.8	69.3	61.4	61.9	55.9	59.5	9.7	41.7	51.80
StreamForest-7B	1 fps	77.26	–	–	68.5	53.2	71.6	47.8	65.4	60.9	61.2	58.9	64.9	32.3	52.0	56.60
Streamo-7B	1 fps	–	–	–	79.2	57.8	75.0	49.4	64.4	70.1	66.0	54.6	52.0	31.7	46.1	56.05
HERMES-7B†	1 fps	79.44	–	–	85.2	64.2	71.6	53.4	74.3	65.2	69.0	48.5	62.2	37.6	49.4	59.20
SimpleStream-4B	16	–	–	–			–				77.5		–		54.6	66.06
StreamTTT-4B	2/4K	81.32	49.60	65.46	91.3	**80.7**	**80.2**	56.2	**82.2**	77.7	78.1	60.9	58.1	**60.8**	59.9	69.00
ReSight-4B (ours)	2/4K	**84.08**	**56.80**	**70.44**	**93.3**	79.8	**80.2**	**62.9**	78.2	**82.1**	**79.4**	**62.0**	**68.2**	50.0	**60.1**	**69.74**

(SQA) on StreamingBench (Lin et al., 2026). OVO reports task-macro category means and their average. StreamingBench reports question accuracy and the RTVU/SQA average. Episodic recall is $\mathrm{ER} = (\mathrm{EPM} + \mathrm{ASI})/2$, excluding HLD. Following StreamTTT (Chen et al., 2026), we use 2 fps, a 262,144-pixel cap per frame, 32-frame ingestion blocks, and native-cache budget $L = 4096$. Official pre-cuts, query cutoffs, prompts, and SQA dialogue are preserved. Only observations update persistent memory, which resets between streams.

Implementation details. We initialize from Qwen3-VL-4B-Instruct and train on 10% of StreamTTT's training data: 22,925 annotations and 53,479 answer turns. We set $d = 2560$, $d_v = 512$, $d_k = 2048$, $g = 12$, and $\lambda = 0.01$. The two-block residual decoder $\mathcal{D}_\eta$ refines $F = 16$ probes; average pooling produces $B = 16 \times 8 \times 8 = 1024$ evidence tokens. Historical Q/V/O adapters have rank 16, with loss weights $(\lambda_H, \lambda_P, \lambda_\Delta) = (0.25, 0.5, 0.5)$. Training uses 86 reader-only and 2,784 joint updates, tuning the top eight language-model layers in the latter stage. Peak learning rates are 2×10^{-6} for these layers, 3×10^{-5} for the reader, and independently 10^{-4} for fusion. Appendix B provides further implementation settings.

3.2 Main Results

Using **only 10% of StreamTTT's training data**, our 4B model surpasses StreamTTT-4B on both benchmarks (Table 1). It reaches 69.74 OVO, 84.08 RTVU, and 56.80 SQA, improving by 0.74, 2.76, and 7.20 points, respectively. The StreamingBench average increases from 65.46 to 70.44. Across the 11 individual task metrics, ReSight improves seven and matches one, demonstrating competitive performance with substantially less adaptation data. The gains span perception and historical recall. OVO perception rises from 78.1 to 79.4, while EPM/ASI improve from 60.9/58.1 to 62.0/68.2. The pronounced gains on ASI and SQA highlight the potential of reconstructive memory for reasoning about earlier actions and answering successive questions over a video stream. Improvements across EPM, ASI, and SQA further suggest that the recovered evidence supports multiple forms of history-dependent reasoning, extending beyond a single recall task. Together, these results are consistent with improved historical reasoning alongside real-time perception.

Table 2: **Protecting perception and controlling history.** (a) Shared versus protected attention. (b) Learned fusion versus fixed endpoints.

(a) Attention

Mask	RP	ER	RTVU	SQA
Shared	69.84	51.03	68.75	46.00
Partial	80.22	59.31	76.88	50.00
Protected	88.19	66.03	83.13	52.00

(b) Fusion

Gate	RP	ER	RTVU	SQA
Learned	86.81	66.03	82.50	54.00
$\gamma = 0$	87.73	66.03	80.00	54.00
$\gamma = 1$	82.41	64.48	79.38	50.00

Table 3: **Memory capacity and readout density.** (a) Memory width d_k at fixed $B = 1024$. (b) Evidence budget B at fixed $d_k = 2048$.

(a) Memory capacity

d_k	RP	ER	RTVU	SQA
512	86.41	61.55	81.25	50.00
1024	87.00	64.14	81.88	52.00
2048	87.60	66.03	82.50	54.00

(b) Readout density

B	RP	ER	RTVU	SQA
512	87.60	63.28	82.50	52.00
1024	87.60	66.03	82.50	54.00
2304	87.50	66.21	82.50	54.00

3.3 Design Analysis

Attention separation (hypothetical analysis). As shown in Table 2a, performance increases from shared to partial to protected attention across all four metrics. Protected attention exceeds the Partial configuration by 7.97 points on RP, 6.72 on ER, 6.25 on RTVU, and 2.00 on SQA, suggesting that blocking direct evidence access at answer positions alone is insufficient: evidence-conditioned question representations can still transmit historical information into the perception path. Complete separation would then support a more stable reference while retaining effective historical reasoning.

Controlling historical influence. Learned fusion outperforms the history endpoint $\gamma = 1$ on all four metrics (Table 2b). Relative to the perception endpoint $\gamma = 0$, it improves RTVU by 2.50 points and matches ER/SQA, but reduces RP by 0.92 points. History therefore provides useful, task-dependent corrections. These results support retaining p_j as a reference when applying the historical correction $h_j - p_j$, while identifying selective use of history as a remaining source of error.

Memory capacity and readout density. At fixed $B = 1024$, increasing d_k from 512 to 2048 improves all four metrics, with larger gains on ER and SQA than on RP and RTVU (Table 3a). Width remains beneficial throughout this range. At fixed $d_k = 2048$, increasing B from 512 to 1024 improves ER and SQA, but increasing it more adds only 0.18 ER points, leaves SQA unchanged, and slightly lowers RP/RTVU (Table 3b). The results therefore distinguish storage capacity from readout density: wider memory continues to improve performance, whereas additional evidence beyond $B = 1024$ offers little benefit in this setting.

3.4 Memory beyond the Native Visual Cache

Recovering information after erasure. We erase native visual context after ingestion while retaining persistent memory, testing whether past observations remain accessible without replaying their frames. On the 39 EPM/ASI questions, ReSight retains 57.59 ER versus 40.52 for StreamTTT (Figure 3a). The gap widens from 1.55 points with the cache intact (69.48 versus 67.93) to 17.07 after erasure. This separation indicates that the visual field preserves historical information whose contribution becomes more apparent when native visual context is unavailable.

Testing the source of recovered evidence. To test whether recovery depends on the stored video, we substitute another video's memory and measure $\Delta_{\text{source}} = \text{ER}_{\text{correct}} - \text{ER}_{\text{donor}}$. At complete erasure, this gap is 20.52 points for ReSight and 0.00 for StreamTTT (Figure 3b). Zeroing reconstructed evidence while preserving the prefix layout and decoding structure yields 45.69 ER, between correct (57.59) and donor memory (37.07). Correct memory therefore contributes 11.90 points over zero evidence, whereas mismatched memory further reduces accuracy. The source gap thus reflects both useful recovery and interference from an incorrect history. Appendix B.2 provides absolute scores and protocol details.

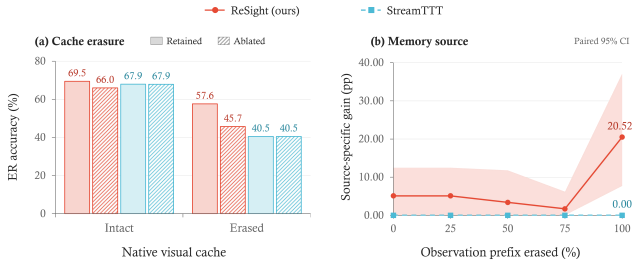


Figure 3: **Memory beyond the native visual cache.** (a) ER with memory retained (solid) or ablated (hatched). (b) Correct-minus-donor ER across erasure levels.

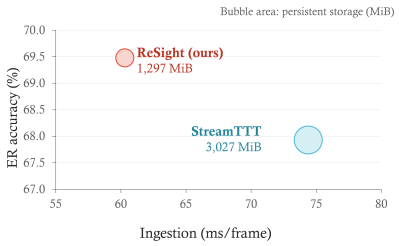


Figure 4: **Efficiency analysis.** Intact-cache ER versus ingestion cost, with measurements at 1,024 frames.

3.5 Efficiency Analysis

Figure 4 combines intact-cache ER with resource measurements at 1,024 frames, evaluated on their respective inputs. ReSight achieves higher ER than StreamTTT, while reducing ingestion cost from 74.39 to 60.30 ms/frame (19%) and persistent storage from 3,027 to 1,297 MiB (57%). At this operating point, the recall gain accompanies lower costs for maintaining history. Together with the erasure results, this supports the visual field as a compact representation that retains useful information beyond the native cache. Storage accounting includes native KV, the visual field, and inverse-Gram state. Appendix B.3 details the protocol.

3.6 Qualitative Analysis

As shown in Figure 5, each question is posed at the second frame, and positive margins favor the annotated answer. In (a), perception favors "in your hand" over the annotated pocket location. Reconstructed evidence enables the historical path to recover the correct location, shifting the fused pocket-minus-hand margin from -0.34 with zero evidence to +0.51 with full evidence. Perception remains unchanged, tying the correction to evidence supplied to the historical path. In (b), perception correctly places the green tower to the left of the walkway, but the historical path favors "in the background," with a left-minus-background margin of -0.15. Fusion retains the correct preference at +0.73, below perception's +3.79. Zeroing evidence restores the historical path's correct preference and raises the fused margin to +1.86. These cases show that historical readout can recover a missed event or introduce a misleading preference. The protected reference helps fusion withstand the latter error here, without fully preserving perception's original margin.

4 Related Work

Memory for streaming LMMs. Streaming memory spans KV caches, compressed observations, retrieval, and recurrent states. StreamingVLM retains attention sinks and recent-token windows, StreamMem compresses KV states without future queries, and HERMES maintains a hierarchical cache (Xu et al., 2026; Yang et al., 2025; Zhang et al., 2026). StreamForest organizes events into memory trees, while FluxMem combines temporal selection with spatial consolidation (Zeng et al., 2025; Xie et al., 2026). ReKV retrieves historical video KV entries from external storage (Di et al., 2025), while FOLIO connects entity-centered semantic records to visual evidence (Fan et al., 2026). StreamTTT and Spatial-TTT accumulate history in fast weights (Chen et al., 2026; Liu et al., 2026). These strategies can coexist, as in LatentStream's combination of storage, retrieval, and latent memory (Qu et al., 2026). Beyond video, MRAgent and MemHarness reconstruct memories through graph-guided reasoning or adaptation of textual experiences (Ji et al., 2026; Wu et al., 2026). ReSight reconstructs spatially resolved visual features from a bounded recurrent state.

Test-time learning for memory. Test-time training adapts models through self-supervision on test inputs (Sun et al., 2020); TTT layers use online-trained models as recurrent states (Sun et al., 2024). Titans develops neural long-term memory (Behrouz et al., 2025), while LaCT uses large-chunk updates for hardware efficiency (Zhang et al., 2025). E^2-TTT parallelizes token-dependent transitions

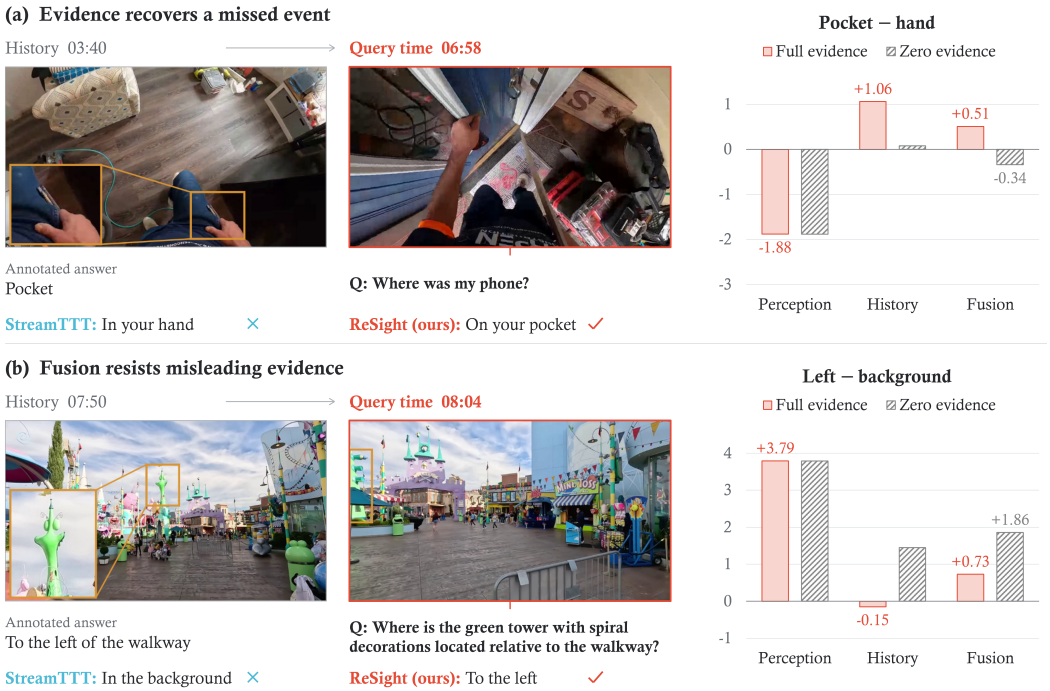


Figure 5: **Qualitative analysis.** (a,b) Reconstructed evidence corrects a perception error. (c,d) Fusion preserves the correct answer despite an erroneous historical preference. Bars show correct-minus-competing logit margins for perception, history, and fusion.

under a chunk-start gradient approximation (Zhong et al., 2026). These methods make the update objective and schedule central to what a recurrent state retains. LatentStream instead refines memory tokens and retrieved embeddings at query time through confidence-based optimization with a frozen backbone (Qu et al., 2026). ReSight uses analytic updates to preserve fixed visual targets, leaving question-dependent interpretation to an offline-trained reader.

Controlling historical influence. SimpleStream identifies a trade-off between historical recall and current-scene accuracy (Shen et al., 2026). SelectStream limits history to a query-conditioned evidence budget while keeping the current observation directly visible (Ge et al., 2026); OASIS retrieves additional context when short-context reasoning remains uncertain (Liang et al., 2026). Other methods regulate memory within the model: StreamTTT gates fast-weight outputs alongside recent-context attention (Chen et al., 2026), and δ-mem introduces low-rank associative corrections to attention (Lei et al., 2026). These approaches control evidence selection or its integration into intermediate representations. ReSight regulates historical influence at the answer representation relative to a protected perception reference.

5 Conclusion

We present ReSight for streaming video understanding under bounded memory. Its central insight is to separate preserving observations from deciding how history should influence an answer. A fixed-size visual field reconstructs spatially resolved evidence, while a protected perception reference and signed fusion regulate its use. With **only 10% of StreamTTT's training data**, ReSight improves seven of eleven task metrics. Separate diagnostics show 19% lower ingestion time and 57% less persistent storage. These results support data-efficient learning and efficient memory maintenance through a reconstructive view of visual memory: preserve evidence before its relevance is known, and learn its influence when a question arrives.

AI USE STATEMENT

Generative AI tools assisted with language polishing and figure-design suggestions. The authors take full responsibility for the final manuscript and its claims.

ETHICS STATEMENT

This work uses existing video benchmarks. Deployment should account for privacy risks from retaining visual history and biases inherited from data and pretrained models, particularly when continuously observing individuals.

REPRODUCIBILITY STATEMENT

Sections 2 and 3.1 describe the method and experimental setup. Appendices A and B provide the supporting derivations, implementation details, training configurations, and evaluation protocols.

REFERENCES

Ali Behrouz, Peilin Zhong, and Vahab Mirrokni. Titans: Learning to memorize at test time. In *NeurIPS*, pp. 113506–113543, 2025.

Joya Chen, Zeyun Zhong, and Mike Zheng Shou. Streamttt: Reconciling real-time perception and long-term memory in streaming vlms. *arXiv preprint arXiv:2608.13416*, 2026.

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 *ICLR*, 2025.

Haoyang Fan, Dhruv Parikh, Anvitha Ramachandran, Sameh Gobriel, Nilesh Jain, Rajgopal Kannan, and Viktor Prasanna. FOLIO: Focused semantic memory for streaming video understanding. *arXiv preprint arXiv:2607.13298*, 2026.

Haonan Ge, Yiwei Wang, Hang Wu, and Yujun Cai. What should a streaming video model remember? *arXiv preprint arXiv:2606.16353*, 2026.

Shuo Ji, Yibo Li, and Bryan Hooi. Memory is reconstructed, not retrieved: Graph memory for llm agents. *arXiv preprint arXiv:2606.06036*, 2026.

Jingdi Lei, Di Zhang, Junxian Li, Weida Wang, Kaixuan Fan, Xiang Liu, Qihan Liu, Xiaoteng Ma, Baian Chen, and Soujanya Poria. $delta$-mem: Efficient online memory for large language models. *arXiv preprint arXiv:2605.12357*, 2026.

Zhijia Liang, Jiaming Li, Weikai Chen, Yanhao Zhang, Haonan Lu, and Guanbin Li. OASIS: On-demand hierarchical event memory for streaming video reasoning. In *CVPR*, pp. 2821–2831, 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*, pp. 12147–12151, 2026.

Fangfu Liu, Diankun Wu, Jiawei Chi, Yimo Cai, Yi-Hsin Hung, Xumin Yu, Hao Li, Han Hu, Yongming Rao, and Yueqi Duan. Spatial-TTT: Streaming visual-based spatial intelligence with test-time training. In *ECCV*, 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 *CVPR*, pp. 18902–18913, 2025.

Hongyu Qu, Guangming Yao, Ling Xing, Xiaobin Hu, Rongxing Ding, Guibin Zhang, Fan Zhang, Yi Yuan, Xiangbo Shu, and Shuicheng Yan. Beyond retrieval: Progressive latent memory evolution for streaming video understanding. *arXiv preprint arXiv:2609.04131*, 2026.

Yujiao Shen, Shulin Tian, Jingkang Yang, and Ziwei Liu. A simple baseline for streaming video understanding. *arXiv preprint arXiv:2604.02317*, 2026.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *ICML*, pp. 9229–9248, 2020.

Yu Sun, Xinhao Li, Karan Dalal, Jiarui Xu, Arjun Vikram, Genghan Zhang, Yann Dubois, Xinlei Chen, Xiaolong Wang, Sanmi Koyejo, et al. Learning to (learn at test time): RNNs with expressive hidden states. *arXiv preprint arXiv:2407.04620*, 2024.

Rong Wu, Daocheng Fu, Licheng Wen, Xuemeng Yang, Shu Zou, Jianbiao Mei, Yuxin Wang, Hairong Zhang, Yu Yang, Tao Hu, et al. Memharness: Memory is reconstructed, not replayed. *arXiv preprint arXiv:2607.28272*, 2026.

Yiweng Xie, Bo He, Junke Wang, Xiangyu Zheng, Ziyi Ye, and Zuxuan Wu. FluxMem: Adaptive hierarchical memory for streaming video understanding. In *CVPR*, pp. 31272–31282, 2026.

Ruyi Xu, Guangxuan Xiao, Yukang Chen, Liuning He, Kelly Peng, Yao Lu, and Song Han. StreamingVLM: Real-time understanding for infinite video streams. In *ICLR*, 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.

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. In *NeurIPS*, pp. 75804–75835, 2025.

Haowei Zhang, Shudong Yang, Jinlan Fu, See-Kiong Ng, and Xipeng Qiu. HERMES: KV cache as hierarchical memory for efficient streaming video understanding. *arXiv preprint arXiv:2601.14724*, 2026.

Tianyuan Zhang, Sai Bi, Yicong Hong, Kai Zhang, Fujun Luan, Songlin Yang, Kalyan Sunkavalli, William T Freeman, and Hao Tan. Test-time training done right. *arXiv preprint arXiv:2505.23884*, 2025.

Zeyun Zhong, Joya Chen, Manuel Martin, Frederik Diederichs, Juergen Gall, and Jürgen Beyerer. Rethinking expressivity and efficiency in test-time training. *arXiv preprint arXiv:2608.21308*, 2026. doi: 10.48550/arXiv.2608.21308. URL https://arxiv.org/abs/2608.21308.

A Memory and Attention Details

A.1 Fixed Visual Coordinates

For $u \in \mathbb{R}^r$, let $\mathsf{N}(u) = u/\sqrt{\|u\|_2^2/r + \epsilon_N}$ denote RMS normalization. For a native visual feature $v \in \mathbb{R}^d$, define $\rho(v) = \max(\|v\|_2/\sqrt{d}, 10^{-6})$. The target associated with temporal patch i is

$$y_i = \mathrm{vec}\Big[\mathcal{R}_g\Big(\big\{[U^\top \mathsf{N}(v_{i,p}); \log \rho(v_{i,p})]\big\}_p\Big)\Big], \tag{8}$$

where $\mathcal{R}_g$ bilinearly resamples the spatial grid with aligned corners disabled. Normalization uses FP32 with $\epsilon_N = 2^{-23}$. The columns of U are the leading d_v eigenvectors of the uncentered second moment of normalized features, estimated from initial windows of up to 16 frames from 128 training sources without answer supervision. Thus $D = g^2(d_v + 1) = 73{,}872$.

With $m = d_k/2$, the fixed temporal address is

$$\phi(\tau) = \frac{1}{\sqrt{m}}\big[\cos(2\pi\omega_u\tau), \sin(2\pi\omega_u\tau)\big]_{u=0}^{m-1}, \qquad \omega_u = \frac{u + \frac{1}{2} + \frac{1}{4}(\xi_u - \frac{1}{2})}{2m}, \tag{9}$$

where timestamps are in seconds and $\xi_u \sim \mathrm{Uniform}(0, 1)$ is drawn once with seed 2026091161. The implementation explicitly normalizes each address to unit length. Frequencies and visual coordinates remain fixed throughout training and inference.

Table 4: **Attention visibility.** Rows are queries; columns are keys.

Queries	Observation cache	Q_P	E	Q_H
Q_P	Instruction + latest-window suffix	Causal window	–	–
E	Instruction + recent context	Q_P^q	Causal	–
Q_H	Instruction + recent context	Q_P^q	All	Causal

A.2 Recursive Update

Let $S_n = \lambda I_{d_k} + \sum_{i=1}^{n} k_i k_i^\top$ and $R_n = \sum_{i=1}^{n} k_i y_i^\top$. The ridge normal equations give $M_n = S_n^{-1} R_n$ and $P_n = S_n^{-1}$. For a new block, $S_{n+b} = S_n + K^\top K$ and $R_{n+b} = R_n + K^\top Y$; applying the Woodbury identity yields Equation equation 2. The implementation uses a linear solve for the $b \times b$ system and symmetrizes the updated P. Writes have unit weight, with no forgetting factor.

The implementation partitions output coordinates into nine spatial groups, each with an inverse-Gram copy. Shared addresses and unit weights make these copies identical in exact arithmetic, implementing the single-P regression in the main text. Storage counts all nine.

A.3 Reconstruction and Pooling

Scale restoration uses the reconstructed appearance and log-scale channels from Section 2.2:

$$x_{f,p} = \widehat{z}_{f,p} \frac{\exp(\mathrm{clip}(\widehat{s}_{f,p}, -7, 3))}{\max(\|\widehat{z}_{f,p}\|_2 / \sqrt{d}, 10^{-6})}. \tag{10}$$

The denominator uses native width d because orthonormal U preserves norms. Each of the two blocks in $\mathcal{D}_\eta$ operates on RMS-normalized coordinates and adds three residuals: a depthwise 3×3 spatial convolution, a $512 \to 1024 \to 512$ SiLU MLP, and a linear map of consecutive probe differences. The spatial branch has a learned scalar $\tanh$ gain; the channel and temporal branches have fixed scale 0.1. Temporal mixing applies only when the probe gap lies in $(0, 3]$ seconds. Zero spatial gains and zero channel/temporal output maps initialize $\mathcal{D}_\eta$ to the identity.

Pooling uses Cartesian products of one-dimensional bins. For a input cells and b output slots, bin j contains indices $\lfloor ja/b \rfloor$ through $\lceil (j+1)a/b \rceil - 1$. Temporal-bin means supply timestamp text; native visual positions encode the layout. The budget B excludes timestamps and delimiters. The prefix and query cache are temporary.

A.4 Attention Isolation

Table 4 specifies the allowed blocks. Here Q_P^q contains the recognized question span in Q_P, excluding answer tokens and recognized option lines. Observation visibility also obeys the native sliding-window rule. Evidence neither consumes that observation budget nor shifts perception positions. Rank-16 adapters act only at historical positions, on Q/V/O projections at one-based layers $8, 12, \ldots, 36$, with scale 0.1. The fusion parameters w_g, b_g initialize to zero.

At fixed parameters, cache, positions, and text, induction over decoder layers makes p_j invariant to E: perception reads only observation and perception states. Cached states can still encode earlier context, and fusion can change subsequent generated tokens. Moreover, $h_j - p_j$ includes differences in cache visibility and adapters, so it does not isolate reconstructed content.

B Experimental Details

B.1 Training and Evaluation

Optimization. The main run uses AdamW on eight H100-80GB GPUs, batch eight, and seed 42. Peak rates are 2×10^{-6} for the upper language layers, 3×10^{-5} for reconstruction and historical adapters, and 10^{-4} for fusion. Each parameter group uses 3% warm-up and cosine decay over its active updates. We use AdamW coefficients $(0.9, 0.999)$, weight decay 0.01, and gradient clipping at 1.0. Backbone weights are BF16; reconstruction, fusion, M, and P are FP32. Training caps

inputs at 480 frames and 131,072 pixels per frame; evaluation allows 10,240 frames. The flag c follows the annotation's real-time metadata. The qualitative checkpoint instead uses fusion learning rate 3×10^{-5}.

Controlled comparisons. The development panel contains 67 RP, 50 backward-tracing, 160 RTVU, and 50 SQA questions. ER uses the 29 EPM and 10 ASI questions within backward tracing. The attention variants share native initialization, training order, batch four, and seed 42, with constant learning rates 2×10^{-6}, 3×10^{-5}, and 10^{-4} for backbone, reader, and fusion. The shared-attention control adds Q_P-to-E access while keeping Q_H blocked. Fusion endpoints substitute p_j or h_j for f_j in the frozen computation graph. This panel is reused for development.

Benchmark inputs. Full evaluation contains 837 OVO perception, 631 backward-tracing, 2,500 RTVU, and 250 SQA questions. In Table 1, numeric Input entries denote frame counts. 2/4K denotes 2 fps with a 4K native cache. SimpleStream-4B uses 16 recent frames at 1 fps. Public SQA scores for InternVL2-8B and Qwen2-VL-7B use 32 frames and 0.2–1 fps, respectively. † denotes Qwen2.5-VL-7B with HERMES and a 4K cache.

B.2 Source-Specific Recovery

Table 5: **Absolute recovery scores (ER)**. C: correct; D: donor; A: ablated memory.

	ReSight			StreamTTT		
Erased (%)	C	D	A	C	D	A
0	69.48	64.31	66.03	67.93	67.93	67.93
25	69.48	64.31	66.03	67.93	67.93	67.93
50	67.76	64.31	66.03	67.93	67.93	67.93
75	66.03	64.31	64.31	66.21	66.21	66.21
100	57.59	37.07	45.69	40.52	40.52	40.52

Table 5 reports absolute scores for all 39 recall questions from 17 source videos, including initially incorrect answers. We freeze full-stream memory and erase a prefix of observation blocks, rounding the removed count down. Native context is rebuilt from instructions and the retained suffix at the original positions; complete erasure leaves only instruction KV. StreamTTT's recurrent updates also see only the suffix during replay; the frozen memory is installed immediately before querying, preventing erased history from leaking into the rebuilt cache.

Donors come from different source videos, assigned by minimizing total absolute log-duration mismatch without answer labels. Methods share the assignment and recipient inputs. For ReSight, we replace the pooled visual evidence with donor evidence or zeros, preserving recipient timestamps, positions, prefix length, and fusion. For StreamTTT, we replace the complete recurrent state with the donor state or its instruction-only state. Native cache is fixed across memory conditions within each method. Training and total state size differ between methods.

The correct-minus-zero gain for ReSight is 11.90 points, with a 95% interval of $[0.00, 28.35]$; the source gap also includes interference from donor evidence. Intervals use 10,000 paired bootstrap resamples of recipient source videos with seed 42, recomputing the EPM/ASI macro average in each resample. Resamples missing either task are redrawn. Curve intervals are pointwise and conditional on the fixed donor assignment.

B.3 Resource Measurement

Methods share preprocessed tensors, prompt, native-cache budget, batch one, and a math-SDPA runtime. Paired comparisons run sequentially on the same exclusive H100. We use three videos at 64, 256, 512, and 1,024 frames, with one warm-up and two measured fresh-stream replays per setting, synchronizing the GPU before and after timing. Latency is the median of video medians; peak allocation is the maximum.

Ingestion includes transfer to GPU, visual encoding, observation processing, and state updates, excluding CPU decoding and preprocessing. Figure 4 combines ingestion cost and persistent storage

at 1,024 frames with intact-cache ER evaluated separately on the recall panel. The comparison measures the cost of maintaining history during continuous observation.

Persistent storage includes native KV, M, all inverse-Gram copies, and the last-patch buffer; weights and query workspace count only toward peak allocation. The default M and nine inverse-Gram copies occupy 577.125 and 144 MiB. Adding the buffer and 576 MiB native cache gives the reported 1,297.27 MiB.