REMEMBERING BY PLACE: PERSISTENT VISUAL MEMORY FOR STREAMING VIDEO AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Streaming video models promise always-on assistants that can later recall what they have seen, yet their memory is bounded by context: observations evicted from the KV cache are discarded rather than preserved as a queryable record. Existing memory agents for long video typically fill this gap by captioning incoming observations and storing text. We argue that this caption-first design is mismatched to visual-spatial experience: it commits visual evidence to language before the question is known, scales writer-LLM call with the observation stream, and does not link revisits to the same place across time. We introduce PLACEMEM, a query-blind, training-free persistent memory agent that serves as the backend of a streaming video model. Taking the surprise-selected observations of Cambrian-S as its front end, PLACEMEM organizes them into persistent place records using visual place and semantic features, and derives language from accumulated place evidence rather than from each observation. A tool-using reader retrieves across both the textual and visual records. On VSI-Super Recall, PLACEMEM improves average accuracy on 10–240-minute streams by 31.7 points over Cambrian-S alone, even with a lightweight 4B reader, and writing language per place rather than per observation cuts writer calls by 38% at competitive accuracy. On VSI-Super Count, it improves Cambrian-S's MRA from 40.7 to 48.2. These results suggest that caption-first memory need not be the default for visual-spatial memory: captions remain strong for semantic recall, but visual features and place organization supply grounding and cross-observation association that text alone does not encode.

1 INTRODUCTION

Streaming video language models process a continuous camera feed in real time, but their memory is bounded by context: visual tokens quickly exhaust the window, so any finite-context model must decide what to retain, compress, or evict. Recent systems manage this *working memory* with sophisticated design. Cambrian-S, for example, uses next-frame prediction error to keep surprising events in a bounded KV cache (Yang et al., 2025a). Yet even a well-managed context window keeps no lasting record: evicted observations are discarded, and the loss grows as streams stretch from minutes to hours. We therefore argue that working memory should be complemented by a *persistent memory* that is written as the stream unfolds and can be queried afterward. Such a memory is especially valuable for *visual-spatial memory*, where an agent must later recall where objects were, what places looked like, or how many of something it encountered.

Building persistent memory for a language model is a problem of *memory representation*: what should be stored, and in what form should it be organized? Because these memory systems are ultimately built for LLMs to query and use, most follow what we term a *caption-first* approach—understandably so, given the expressive power of language. mem0 established an extract-then-update loop for conversational memory in text (Chhikara et al., 2025); WorldMM constructs episodic and semantic graphs by exhaustively captioning temporal segments (Yeo et al., 2026); and MemDreamer and VAM build hierarchical textual stores navigated by retrieval tools (Chen et al., 2026a; Li et al., 2026). Caption-first memory is compact, searchable, and immediately compatible with an LLM reader. For general conversational history, it is often sufficient. But for visual-spatial memory, this design leaves critical gaps. A caption commits visual evidence to language before the downstream question is known, so details the writer did not anticipate cannot later be recovered from text.

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.

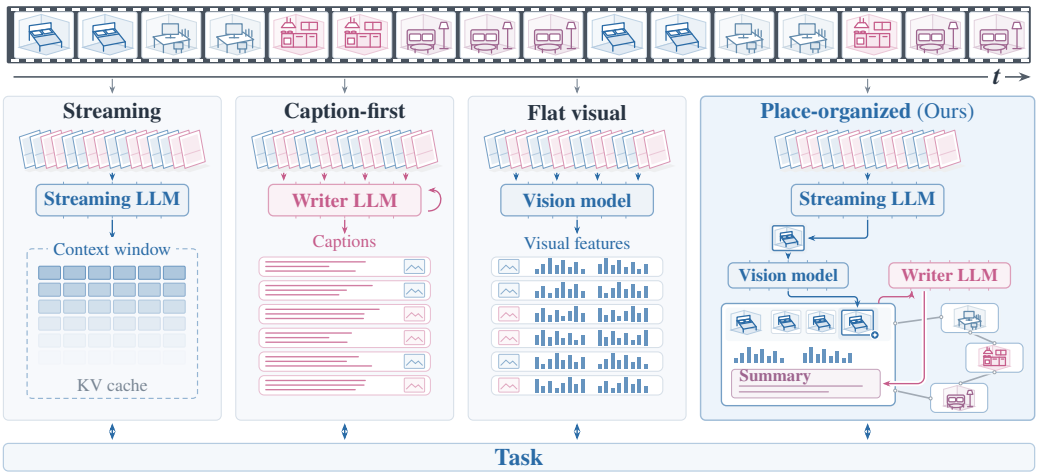


Figure 1: **What should a streaming video agent remember?** A streaming LLM keeps a bounded KV cache that gradually forgets; caption-first memory repeatedly captions the incoming visual feed and stores caption entries; flat visual memory stores per-observation features. PLACEMEM routes each observation to its place record, which accumulates images, features, and a summary that the writer updates from the place's evidence.

Captions do not naturally encode that two temporally distant observations depict the same physical location. And in a streaming system, captioning every retained keyframe ties a writer-LLM call to every observation, with no way to amortize the cost as the stream grows.

These limitations point toward a different organizing principle. Neuroscience has long recognized that humans organize spatial experience around locations: the hippocampal cognitive map anchors memories to where they occurred rather than when (O'Keefe & Nadel, 1978). The robotics and embodied-AI communities have operationalized this insight in structured spatial representations built around the same notion of *place*, including open-vocabulary 3D scene graphs (Gu et al., 2024; Werby et al., 2024) and topological maps (Zhang et al., 2024b; Hughes et al., 2022; Gorlo et al., 2026). These systems demonstrate that place is a powerful organizing principle for spatial experience. We ask whether this inspiration can be transferred to the streaming-video setting: can place organization serve as the structural representation of a persistent memory agent for video language models, even when the input is passive egocentric video without pose or depth signals?

We introduce PLACEMEM, a query-blind, training-free persistent memory agent that serves as the backend of a streaming video model, using Cambrian-S as its front end. Surprise-based keyframe selection determines *when* evidence enters memory; visual place recognition and semantic features then organize retained observations into persistent *place records*—without invoking a language model. Language is derived from accumulated place evidence, as place-level summaries and object inventories, rather than generated for every incoming observation. At query time, a tool-using reader retrieves across both the textual and visual records.

We evaluate PLACEMEM on both tasks of the VSI-Super benchmark (Yang et al., 2025a), which pairs long-horizon spatial recall with continual counting on streams of up to four hours, beyond any fixed context window, and is thus a natural test for the persistent memory of an always-on agent. On VSI-Super Recall, PLACEMEM improves average accuracy on 10–240-minute streams by 31.7 points over Cambrian-S alone (41.3% to 73.0%) with a Qwen3-VL-4B reader, and outperforms caption-first baselines fed the same surprise-selected observations on average, with the largest margins on the longest streams. Deferring language to one place-level pass reduces writer-LLM calls by 38% while remaining competitive with caption-first baselines, and loses nothing with a stronger reader. On VSI-Super Count, where captions and frame subsampling cannot distinguish new objects from repeated views of the same scene, the place-based memory helps Cambrian-S by localizing revisited segments to places, avoiding the duplicate counting that arises when observations are treated independently; this raises Cambrian-S's MRA from 40.7 to 48.2. Together, these results suggest that for visual-spatial memory, organizing observations by place before committing them to language

offers structural advantages—in both retrieval accuracy and writer efficiency—that a caption-first record does not provide.

In summary, we contribute: (1) a query-blind, training-free persistent memory backend for streaming video that separates event selection from spatial organization and readout; (2) a visual-first write path in which place records organize retained observations before language is produced, reducing writer-LLM invocations while preserving competitive retrieval accuracy; and (3) a controlled study of memory representations on VSI-Super Recall and Count, showing that captions remain strong for semantic recall, but visual features and place organization supply grounding and cross-observation association that a caption-first record does not encode.

2 Related Work

Memory systems for long video. Two complementary lines of work extend video understanding beyond the context window. *Streaming* approaches compress visual state inside the model: MovieChat and Flash-VStream consolidate frame tokens into fixed-size memories (Song et al., 2024; Zhang et al., 2024a), and Cambrian-S converts latent prediction error into a drop-and-merge policy over the KV cache under a fixed token budget, a selective but lossy working memory (Yang et al., 2025a). Others learn spatial memory in the weights: LongSpace injects layer-aware spatial memory into the decoder and trains the model to use it (Lang et al., 2026). *Agentic* approaches construct a persistent external store alongside the model. mem0 established the extract-then-update loop for conversational text memory (Chhikara et al., 2025); VideoAgent pairs temporal event descriptions with object-centric tracking states (Fan et al., 2024); M3-Agent builds an entity-centric multimodal memory graph (Long et al., 2025); EM-LLM segments token streams into episodic events by surprise (Fountas et al., 2025); WorldMM combines caption-derived episodic and semantic graphs with a parallel visual store and adaptive multimodal retrieval (Yeo et al., 2026); MemDreamer builds a hierarchical text graph navigated by retrieval tools (Chen et al., 2026a); and VAM adds online selective indexing over a hierarchical store (Li et al., 2026). WorldMM is closest to ours in retaining visual evidence alongside text, but constructs its textual memories by exhaustively captioning predetermined temporal segments. We instead use visual place organization to determine when and at what granularity language should be written.

Structured spatial memory in robotics and embodied AI. Robotics and embodied AI have long organized spatial memory around *place*. Embodied agents with access to geometry build explicit scene memories, such as 3D-Mem's snapshot memory for active exploration with poses (Yang et al., 2025b); open-vocabulary 3D scene graphs such as ConceptGraphs (Gu et al., 2024) and HOV-SG (Werby et al., 2024) anchor semantic labels to spatial locations; multi-view scene graphs associate observations across viewpoints (Zhang et al., 2024b); and topological place graphs, dating back to the Spatial Semantic Hierarchy (Kuipers, 2000), abstract continuous space into discrete nodes linked by traversability. The underlying computational mechanism—visual place recognition (Lowry et al., 2016)—has matured from NetVLAD (Arandjelović et al., 2016) to recent aggregators such as SALAD (Izquierdo & Civera, 2024) and MegaLoc (Berton & Masone, 2025). Organizing memory by place is also the oldest mnemonic technique on record: the method of loci stores items at imagined locations and retrieves them by mental walkthrough (Yates, 1966), a strategy grounded in the hippocampal cognitive map (O'Keefe & Nadel, 1978). We import this place-recognition machinery into a different setting: passive egocentric video without pose or depth, where place must be recovered from appearance alone.

What multimodal memory should store. A recent training-free line of work examines the division of labor between stored language and stored pixels. CFD observes that language memories carry long-range temporal structure while pixels decide attribute-level questions, and routes frame access accordingly at query time (Cai et al., 2026). CVMA audits visual KV eviction and finds that facts already verbalized are safe to forget while unstated ones are not (Chen et al., 2026b). SMA stores distilled textual lessons to improve frozen-VLM spatial reasoning (Zhang et al., 2026). At the extremes, MemDreamer commits to a text-only store (Chen et al., 2026a), while the MemPalace toolkit organizes conversational text into spatial metaphors of "wings" and "rooms" (MemPalace contributors, 2026)—place as label rather than as a computational primitive. We extend this question along the spatial axis and to a third carrier: visual features that organize memory by place

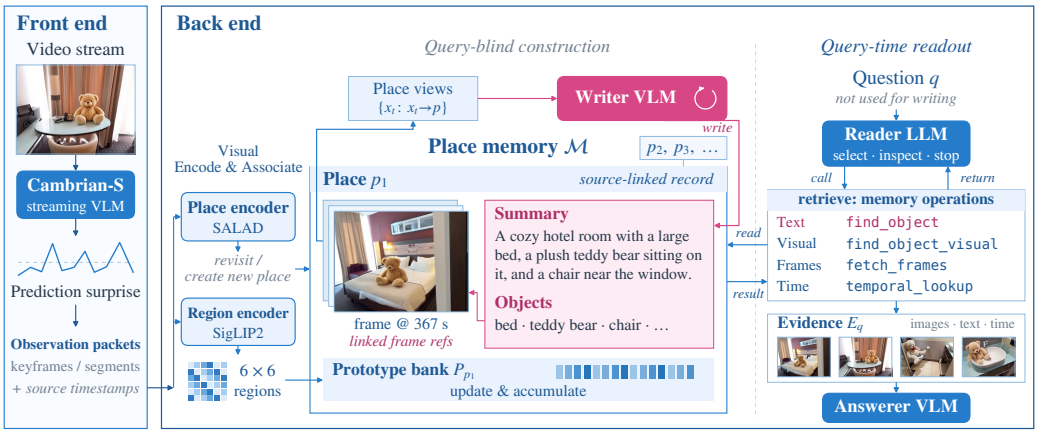


Figure 2: **Overview of PLACEMEM.** The Cambrian-S front end feeds in observations selected by prediction surprise. During query-blind construction, SALAD routes each to a new or revisited place, SigLIP2 region features update the place's region prototypes, and a writer VLM maintains a summary and object inventory from the accumulated views. At query time, a reader LLM calls memory operations and assembles evidence E_q for the answerer. Blue: visual; pink: language.

before language is produced. Our experiments reach a different division of labor than these prior studies: captions are strong for semantic recall, but a caption-only record does not directly encode the cross-view identity and overlap needed for tasks such as counting under repeated observation. Visual place structure supplies the association that text omits.

3 QUERY-BLIND VISUAL-SPATIAL MEMORY

3.1 PROBLEM SETTING AND OVERVIEW

A memory agent that works in realistic settings must observe and write memory before it knows what will be asked. Given an open-ended video stream $x_{1:t}$, it must therefore build its state $\mathcal{M}_t$ from the stream alone; when a question q arrives, a reader returns a compact evidence set $E_q = R(q, \mathcal{M}_t)$ instead of replaying the video. PLACEMEM splits this problem between a front end and a back end (Figure 2). The front end, a streaming VLM, decides *when* an observation enters memory. The back end decides *how* memory is organized, during query-blind construction, and *which* evidence reaches the answerer, during query-time readout. No component is trained or fine-tuned in the current version of PLACEMEM.

3.2 FRONT END: SURPRISE-DRIVEN OBSERVATION PACKETS

We use Cambrian-S (Yang et al., 2025a) as the front end. Cambrian-S predicts the latent features of the next frame and treats the prediction error as *surprise*, to manage its streaming context. It builds two kinds of streaming context: for recall, the surprising keyframes it retains by keeping their KV cache in its runtime window; for counting, segments of the stream split where surprise exceeds a threshold. PLACEMEM wraps these contexts into *observation packets*: each packet carries the keyframe or segment frames together with their source timestamps and surprise values.

The surprise mechanism makes Cambrian-S a good front end for two reasons. First, surprise is a native streaming signal: it is computed from the model's own predictions as frames arrive, requires no additional pass over the video, and is query-blind by construction. Second, it decouples the front end from the back end by acting as a trigger: the streaming model keeps handling the present in its own context, and the back end is invoked only when something new is observed, so evidence evicted from the context is no longer lost. This decoupling lets us keep one front end fixed and focus on the back end, while conceptually any streaming video model could be designed to emit similar packets and connect to it.

3.3 Query-Blind Construction: Visual Encoding and Place Association

When a packet reaches the back end, each of its images is registered visually, without any language model, by two encoders: a place encoder decides which place the image belongs to, and a region encoder updates what that place contains. The *place encoder*, SALAD (Izquierdo & Civera, 2024), produces a global descriptor of the surrounding place. If the descriptor matches an existing place in $\mathcal{M}$ above a similarity threshold, the image is a *revisit* and joins that place; otherwise a new place is created. This is loop closure from visual place recognition (Lowry et al., 2016), repurposed as a memory primitive: returning to a place updates its record instead of duplicating it.

The *region encoder*, SigLIP2 (Tschannen et al., 2025), preserves local semantic evidence that a single global embedding could average away, so we extract its features on a 6×6 grid of image regions. Each region feature is assigned to its most similar entry in a per-place bank P_p and folded into that entry's running mean, or starts a new entry when no similarity exceeds a threshold. Because each entry averages similar region features across views, we refer to these entries as the *region prototypes* of the place, following prototypical networks (Snell et al., 2017). The bank grows with the place's visual content rather than with the number of frames.

The result is a place-organized memory $\mathcal{M} = \{p_1, p_2, \ldots\}$ in which each place is a *source-linked record*: references to its frames and their timestamps, its region prototypes, and, once written, a summary and an object inventory (Section 3.4). Revisits separated in time update the same record, while timestamps preserve within-place order and support lookup by time. Because a reader can retrieve frames or region prototypes from a text query, and the counting path can use place identity directly, the memory is usable from vision alone, before any text is written.

3.4 Language as a Derived Place View

We compare two ways of introducing language into a query-blind memory. A *caption-first* writer invokes a VLM on every retained observation and stores the resulting caption, linked with the image. This is a strong recall baseline, but it has two costs: the writer is called whenever an observation arrives, so the write rate follows the event stream, and the captions accumulate as a flat list with no structure linking observations of the same place.

A *visual-first* writer, our design, treats language as a higher-level view derived from a place rather than as the primary record of each visit. For each place p, the back end buffers its newly assigned images, the *place views* $\{x_t : x_t \to p\}$, and invokes the writer once K new views have accumulated. The writer reads the place's current text together with these views and revises two records: a semantic summary of the place and the events observed there, and an inventory of visible objects. The frames, timestamps, and region prototypes remain available for visual verification. With $K{=}1$, the record is revised with every new view; this eager schedule approximates the write rate of a caption-first writer, but produces place-level rather than event-level text. With $K{=}\infty$, writing is deferred until the stream has been observed, giving one query-blind pass per place. In practice, updates can also happen intermittently between these extremes; a natural trigger is to re-caption a place only when its new views drift from its region prototypes, which we leave to future work.

3.5 Query-Time Readout

The question q is visible only to the read path. For recall, the reader is a tool-using LLM that iterates select, inspect, and stop over four memory operations: `find_object` searches place summaries and object inventories by text; `find_object_visual` retrieves source frames and region prototypes by text-to-image similarity; `fetch_frames` returns the source images of a place or a time span; and `temporal_lookup` resolves a timestamp to its place, frame references, and summary. The reader collects the returns into the evidence set E_q of images, text, and timestamps, and an answerer VLM produces the answer. The answerer can be the reader itself or Cambrian-S, in which case the evidence is served back to the streaming model.

For counting, Cambrian-S emits segments rather than keyframes, so place matching is relaxed to the segment level. Each segment is counted independently by Cambrian-S's pretrained counter, with long segments fitted to its visual window. Before the counts are aggregated, segments are localized to places: two segments belong to the same place when at least a fraction ρ of one segment's frames have a SALAD match above τ in the other, and segments that belong to the same place contribute

a single count. This path consumes no captions. Recall and counting therefore share the same place-organized memory while exercising different read operations: retrieval of past facts versus association and deduplication of repeated evidence.

4 EXPERIMENTS

Our experiments ask whether an agent that must remember before knowing the question should caption observations as they arrive, or organize them visually by place and derive language later. Section 4.2 tests whether place organization improves long-horizon recall over caption-first memory and lets language be written less often; Section 4.3 tests whether place identity helps where captions cannot, associating repeated observations before counts are aggregated; and Section 4.4 asks what a spatial memory should store by ablating text, images, features, and place structure.

4.1 EVALUATION SETUP

We evaluate on VSI-Super (Yang et al., 2025a), designed to test continual visual-spatial perception on streams longer than any fixed context window. Its two tasks stress complementary abilities of a persistent memory. Recall inserts an unusual object at four locations in a long indoor walkthrough and asks, as a multiple-choice question, where it appeared and in what order, testing whether sparse details survive a long horizon. Count asks for the total number of target objects across rooms seen from changing viewpoints, testing whether memory accumulates state without forgetting or double counting. Streams of 10–240 minutes emulate an always-on camera at growing time scales. We report Recall accuracy and, for Count, the official mean relative accuracy (MRA), which credits a prediction at each of ten relative-error tolerances it meets and so changes only when an error crosses a tolerance. We complement it with mean absolute relative error (MARE), which averages the relative error directly and therefore also registers smaller improvements (Appendix C).

All systems share the Cambrian-S-7B front end. Unless stated otherwise, the writer is Qwen3-VL-4B (Bai et al., 2025), the place and region encoders are SALAD and SigLIP2, and the reader also answers. We deliberately use a small, fast writer because it runs throughout the stream, so its cost grows with stream length. Our main readers are the lightweight open Qwen3-VL-4B and the strong proprietary GPT-5.5 (OpenAI, 2026); the writer–reader study (Table 3c) adds Qwen3-VL-8B, GPT-5.5, and GPT-5.6 Luna as writers and Qwen3-VL-8B as a reader. All models are frozen.

4.2 RECALL: PLACE ORGANIZATION AND LANGUAGE-WRITE FREQUENCY

Caption-first baselines. Caption-only and Caption + Image receive the same surprise-selected observations as PLACEMEM and write once per observation, as does PLACEMEM ($K{=}1$). They store one caption per observation, whereas PLACEMEM stores a summary and an object inventory per place, so the comparison isolates how memory is organized. Writer cost counts external generation calls per nominal video minute, not tokens or wall-clock time (Appendix B).

Does place organization improve recall at the same writer cost? Table 1 compares streaming models without an external backend, caption-first backends, and PLACEMEM. With the Qwen3-VL-4B reader, PLACEMEM ($K{=}1$) averages 73.0% across 10–240 minutes, against 67.7% for Caption-only and 70.0% for Caption + Image, with the same surprise-selected inputs and the same 9.86 writer calls per minute. The largest margins come on the longest streams: at 120 and 240 minutes, place-organized readout is 10.0 and 5.0 points above Caption + Image, where the flat caption record is longest. All external backends far exceed the streaming models that keep everything in context (41.3% for Cambrian-S). Place organization thus improves what a reader can recover for the same number of writes, with the largest benefit on the longest streams.

Can language be written less often? Deferring place writes ($K{=}\infty$) reduces writer calls by 38.2%, with per-duration savings of 30.4–43.9% (Appendix B). With the Qwen3-VL-4B reader, the deferred store reaches 68.3%, 4.7 points below $K{=}1$, the gap growing with duration. But with GPT-5.5, the same deferred store reaches 86.7% against 85.7% for $K{=}1$ and is the best configuration at 240 minutes with a score of 90.0%. The deferred store is therefore not intrinsically poorer; the lightweight reader is less able to exploit its coarser text. Language can thus be written in one pass

Table 1: **VSI-Super Recall across video durations.** All backends use a Qwen3-VL-4B writer. †Reported by Yang et al. (2025a). Caption-only and Caption + Image receive the same Cambrian-S surprise-selected frames as PLACEMEM, so they write only on those frames, not on every frame.

Method	Reader	Video duration (min) 10	30	60	120	240	Avg.	Writes /min
Streaming models without an external backend								
Cambrian-S†	—	45.0	41.7	40.0	40.0	40.0	41.3	—
MovieChat†	—	18.3	21.7	16.7	26.7	25.6	21.8	—
Flash-VStream†	—	28.3	33.3	23.3	28.3	31.7	29.0	—
Caption-first backends								
WorldMM	Qwen3-VL-4B	23.3	18.3	25.0	20.0	23.3	22.0	42.93
WorldMM	GPT-5.5	95.0	93.3	90.0	91.7	93.3	**92.7**	42.93
Cambrian-S + Caption-only	Qwen3-VL-4B	71.7	73.3	66.7	65.0	61.7	67.7	9.86
Cambrian-S + Caption + Image	Qwen3-VL-4B	75.0	73.3	73.3	63.3	65.0	70.0	9.86
Our place-organized backend (Cambrian-S frontend)								
PLACEMEM ($K = 1$)	Qwen3-VL-4B	78.3	73.3	70.0	73.3	70.0	**73.0**	9.86
PLACEMEM ($K = \infty$)	Qwen3-VL-4B	78.3	71.7	68.3	63.3	60.0	68.3	**6.09**
PLACEMEM ($K = 1$)	GPT-5.5	91.7	90.0	86.7	85.0	75.0	85.7	9.86
PLACEMEM ($K = \infty$)	GPT-5.5	86.7	85.0	88.3	83.3	90.0	86.7	**6.09**

per place rather than once per observation at roughly 60% of the cost, and a capable reader loses nothing. This matches how the two roles scale: the writer runs throughout the stream and must stay cheap, whereas the reader is invoked only when a question arrives and can afford a stronger model, the regime in which deferred writing matches eager writing.

Comparison with WorldMM. WorldMM (Yeo et al., 2026) is a strong caption-first system that captions clips at multiple temporal scales and adds VLM2Vec-V2 Meng et al. (2025) visual embeddings. With the same Qwen3-VL-4B writer, it requires 42.93 writer calls per minute, nearly seven times our deferred rate, and reaches 22.0% with the Qwen3-VL-4B reader and 92.7% with GPT-5.5. PLACEMEM is 3.3× more accurate with the lightweight reader and within six points with GPT-5.5, at one seventh of the writer cost. Exhaustive captioning can be made to work with a strong reader; place organization makes the memory usable by a small one.

4.3 COUNTING: ASSOCIATING REPEATED VISUAL EVIDENCE

Counting exposes what a caption-only record lacks. Independently counting two segments does not reveal whether they contain new objects or repeated views of the same ones, and no caption states that two moments depict the same place. Starting from Cambrian-S surprise segments, Segment Sum fits long segments to a 128-frame window, the native context size of Cambrian-S, and sums per-segment counts. PLACEMEM keeps these inputs and counts and localizes segments to places before aggregation (Section 3.5), with no additional language-model call. We evaluate Cambrian-S, Qwen3-VL-4B, and Qwen3-VL-8B as counters; GPT-5.5 with 50 uniform frames is a fixed-budget reference. Place-matching thresholds were selected on the 10-minute split and fixed for all other durations and counters.

Does place identity help where captions cannot? Table 2 shows that both baselines without place organization degrade on long streams: the streaming counter averages 40.7 MRA, and uniform sampling collapses beyond 10 minutes (8.5 average) because 50 frames cannot cover a two-hour video. With the Cambrian-S counter, place matching raises average MRA from 46.4 to 48.2 and lowers MARE from 47.4 to 43.0, changing 95 of 200 predictions. The largest gain is at 120 minutes, where revisits are most frequent. MARE moves more than MRA: at 60 minutes MARE falls from 81.8 to 72.5 while MRA rises by only 0.4, because MRA credits a prediction only when its relative error crosses one of ten tolerance levels. Place matching improves the average of every counter: the gain is largest for Cambrian-S. Place identity thus supplies an additional association that neither

Table 2: **VSI-Super Count across video durations.** MRA and MARE averaged over durations; parentheses show changes from the same counter's Segment Sum, and ΔQA counts changed predictions out of totaling 200 questions.

		MRA ↑ / duration (min)				Average		
Method	Answerer	10	30	60	120	MRA (Δ) ↑	MARE (Δ) ↓	ΔQA
Baselines without place organization								
Streaming	Cambrian-S	40.8	41.2	35.0	45.8	40.7	56.1	
Uniform 50 frames	GPT-5.5	31.8	2.0	0.2	0.0	8.5	67.1	
Segment Sum	Cambrian-S	44.6	49.0	33.6	58.2	46.4	47.4	
Segment Sum	Qwen3-VL-4B	40.4	48.6	36.6	66.4	48.0	31.9	
Segment Sum	Qwen3-VL-8B	40.4	49.8	41.2	61.4	48.2	31.1	
Our place-organized backend (Cambrian-S frontend)								
PLACEMEM	Cambrian-S	47.0	48.8	34.0	62.8	**48.2** (+1.8)	**43.0** (−4.4)	95/200
PLACEMEM	Qwen3-VL-4B	40.6	48.2	37.8	66.0	**48.2** (+0.2)	**31.8** (−0.1)	28/200
PLACEMEM	Qwen3-VL-8B	41.6	50.8	42.4	61.0	**49.0** (+0.8)	**30.8** (−0.4)	37/200

captions nor frame sampling provides, and its headroom is bounded by how many duplicate errors the counter makes.

4.4 WHAT SHOULD SPATIAL MEMORY STORE?

Table 3: **Recall ablations at 10 minutes** (accuracy, %). (a) Caption-based readouts, with images and place tools added cumulatively, versus PLACEMEM; (b) vision-only retrieval and reranking of caption–image candidates; Prototypes are the region prototypes of each place (Section 3.3). All variants use the same surprise-selected observations; columns in (a, b) are readers. (c) Writers × readers with full memory. Qwen 4B/8B denotes Qwen3-VL-4B/8B. Bold marks column-wise best results within each panel, including ties.

(a) Memory readout

Memory Readout	Qwen 4B	Cambrian-S
Caption-only	71.7	71.7
+ Image	75.0	75.0
+ Place	71.7	71.7
PLACEMEM	**78.3**	**78.3**

(b) Visual evidence

Memory Readout	Qwen 4B	Cambrian-S
Global	71.7	63.3
Prototypes	61.7	43.3
Rerank, global	73.3	**71.7**
Rerank, 6×6	**78.3**	70.0

(c) Writer × reader

Memory Writer	Qwen 4B	Qwen 8B	GPT-5.5
Qwen 4B	78.3	**81.7**	**91.7**
Qwen 8B	**80.0**	80.0	90.0
GPT-5.5	**80.0**	75.0	86.7
GPT-5.6 Luna	68.3	71.7	**91.7**

Memory readout. Table 3(a) compares the recall readouts at 10 minutes built from the same surprise-selected observations. Per-observation captions alone reach 71.7%; adding source images raises this to 75.0%; adding place tools to the caption readout does not help by itself (71.7%), whereas PLACEMEM, which reads place summaries, object inventories, images, and region prototypes, reaches 78.3%. Table 3(b) shows that vision-only memory is viable, but both the representation and how it is used matter: SigLIP2 global retrieval reaches 71.7% with Qwen3-VL-4B and 63.3% with Cambrian-S, and region prototypes alone reach 61.7% and 43.3%. Reranking caption–image candidates with 6×6 region features lifts Qwen3-VL-4B to 78.3%, matching the full PLACEMEM, but is less effective for Cambrian-S. No single form of evidence dominates: text is the strongest single access path to semantic evidence, images add grounding, and place structure adds association, so the memory should preserve complementary access to all three.

Writers and readers. Table 3(c) crosses four writers with three readers on full memory. GPT-5.5 readout reaches 86.7–91.7% across these stores. The GPT-5.6 Luna-written store reaches 91.7% with GPT-5.5 but only 68.3% and 71.7% with the Qwen readers, below the stores written by Qwen3-VL-4B and Qwen3-VL-8B (78.3–81.7%). A writer's utility therefore depends on the downstream reader, not only on the writer in isolation.



Figure 3: **VSI-Super counting ablations (10 minutes).** (a) MRA versus similarity threshold τ at coverage threshold ρ=0.90. (b) MRA by strategy and answerer. Segment Sum denotes segment-wise summation.

Place matching features. Figure 3(a) sweeps the similarity threshold τ for place and semantic features on the 10-minute Count split, with the coverage threshold fixed at ρ=0.90. Both peak near 47 MRA, but semantic matching spans 33.8–47.2 over the sweep while place matching spans 44.6–47.0: place descriptors are the more robust matching signal, and similar peaks mask very different sensitivity. Figure 3(b) confirms that segmenting alone is not the source of the gain: for GPT-5.5, segment-wise summation raises MRA only from 31.8 to 32.6 despite six times more frames, whereas adding place matching to the same fitted segments raises Cambrian-S from 44.6 to 47.0, isolating the contribution of place organization.

4.5 Qualitative Analysis

Qualitative examples of PlaceMem on recall and counting appear in Appendix D. In recall, four retrieved places and their verbatim memory excerpts recover a teddy bear's location sequence (Figure 4). In counting, a return to a kitchen after an intervening room is associated with the earlier visit, avoiding counting the same object twice (Figure 11). Further examples and the complete stored Recall text appear in Appendices D and E.

5 Conclusion

Limitations and future work. PlaceMem is training-free, and its thresholds were selected on the 10-minute split. Learned place assignment and re-captioning triggers, or a reader trained on the memory operations, could make it stronger and more stable, and stronger future models may shift the best design choices. Finally, we evaluate one front end and one evaluation suite. Although Cambrian-S and VSI-Super are representative of streaming spatial memory, and fixing them keeps the comparison controlled, other front ends and tasks such as metric reasoning or navigation can be explored in the future, as can combining appearance-based place recognition with geometry. Our large gains over Cambrian-S stem from adding a persistent memory, the very point this paper argues and they do not diminish VSI-Super as a test of streaming models on their own, nor the value of improving them, since a stronger streaming model also makes a better front end for a memory backend.

Conclusion. We introduce PlaceMem, a persistent memory agent that serves as the backend of a streaming video model. It organizes surprise-selected observations into place records using visual features and derives language per place rather than per observation. On VSI-Super, it improves long-horizon recall over the streaming model and over caption-first baselines fed the same observations, cuts writer calls by 38% at competitive accuracy, and localizes revisited segments to places to avoid double counting. We explore the choices of memory representations and find that vision-first place organization offers particular value for visual-spatial streaming video: grounding, association across revisits, and cheaper writing.

AI USE STATEMENT

In this work, we used generative AI tools for method implementation: AI coding assistants helped implement parts of the memory system and the experiment code. We have not used generative AI tools for research ideation, hypothesis formulation, experiment design, theoretical or mathematical claims and proofs; Additionally, we used generative AI tools for drafting and polishing the text of the manuscript and for drafting and refining figures, under specific human instructions. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

PLACEMEM is built entirely on open-source code, model weights, and data. The front end is the released Cambrian-S-7B model and code (Yang et al., 2025a); place and region features come from the public SALAD (Izquierdo & Civera, 2024) and SigLIP2 (Tschannen et al., 2025) checkpoints; and all experiments use the publicly released VSI-Super benchmark. The open-weight Qwen3-VL models (Bai et al., 2025) are served locally with vLLM (Kwon et al., 2023), while GPT models are accessed through their provider's cloud API; since hosted models can change over time, we record the exact model identifiers used. We will release our code, prompts, and configurations to reproduce all reported results.

REFERENCES

Relja Arandjelović, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

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

Gabriele Berton and Carlo Masone. MegaLoc: One retrieval to place them all. *arXiv preprint arXiv:2502.17237*, 2025.

Weitong Cai, Hang Zhang, Yukai Huang, Yiqiao Xie, Shan Gao, Jiankang Deng, Songcen Xu, Jifei Song, and Zhensong Zhang. Caption-once, frames-on-demand: Visual-need routing for budget-aware agentic long video understanding. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2026.

Cong Chen, Guo Gan, Kaixiang Ji, ZhaoYang Zhang, Zhen Yang, Guangming Yao, Hao Chen, Jingdong Chen, Yi Yuan, and Chunhua Shen. MemDreamer: Decoupling perception and reasoning for long video understanding via hierarchical graph memory and agentic retrieval mechanism. *arXiv preprint arXiv:2606.07512*, 2026a.

Hong Chen, Kang Chen, Yuxuan Fan, Bo Wang, Yubo Gao, Yuanlin Chu, and Xuming Hu. Seen, said, or forgotten? A causal audit of visual KV memory across dialog turns. *arXiv preprint arXiv:2607.25467*, 2026b.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building production-ready AI agents with scalable long-term memory. *arXiv preprint arXiv:2504.19413*, 2025.

Yue Fan, Xiaojian Ma, Rujie Wu, Yuntao Du, Jiaqi Li, Zhi Gao, and Qing Li. VideoAgent: A memory-augmented multimodal agent for video understanding. In *European Conference on Computer Vision (ECCV)*, 2024.

Zafeirios Fountas, Martin A Benfeghoul, Adnan Oomerjee, Fenia Christopoulou, Gerasimos Lampouras, Haitham Bou-Ammar, and Jun Wang. Human-like episodic memory for infinite context LLMs. In *International Conference on Learning Representations (ICLR)*, 2025.

Nicolas Gorlo, Lukas Schmid, and Luca Carlone. Describe anything anywhere at any moment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 35002–35013, 2026.

Qiao Gu, Alihusein Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, Chuang Gan, Celso Miguel de Melo, Joshua B Tenenbaum, Antonio Torralba, Florian Shkurti, and Liam Paull. ConceptGraphs: Open-vocabulary 3D scene graphs for perception and planning. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

N. Hughes, Y. Chang, and L. Carlone. Hydra: A real-time spatial perception system for 3D scene graph construction and optimization. 2022.

Sergio Izquierdo and Javier Civera. Optimal transport aggregation for visual place recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Benjamin Kuipers. The spatial semantic hierarchy. *Artificial Intelligence*, 119(1-2):191–233, 2000.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Shiqiang Lang, Jing Liu, Haoyang He, Peiwen Sun, Yuanteng Chen, Tao Liu, Lan Yang, Longteng Guo, and Honggang Zhang. LongSpace: Exploring long-horizon spatial memory from perception to recall in video. *arXiv preprint arXiv:2606.05677*, 2026.

Aiden Yiliu Li, Nels Numan, and Anthony Steed. Visual agentic memory: Enabling online long video understanding via online indexing, hierarchical memory, and agentic retrieval. *arXiv preprint arXiv:2605.16481*, 2026.

Lin Long, Yichen He, Wentao Ye, Yiyuan Pan, Yuan Lin, Hang Li, Junbo Zhao, and Wei Li. Seeing, listening, remembering, and reasoning: A multimodal agent with long-term memory. *arXiv preprint arXiv:2508.09736*, 2025.

Stephanie Lowry, Niko Sünderhauf, Paul Newman, John J Leonard, David Cox, Peter Corke, and Michael J Milford. Visual place recognition: A survey. *IEEE Transactions on Robotics*, 32(1): 1–19, 2016.

MemPalace contributors. MemPalace: Local-first AI memory. `https://github.com/MemPalace/mempalace`, 2026.

Rui Meng, Ziyan Jiang, Ye Liu, Mingyi Su, Xinyi Yang, Yuepeng Fu, Can Qin, Zeyuan Chen, Ran Xu, Caiming Xiong, et al. Vlm2vec-v2: Advancing multimodal embedding for videos, images, and visual documents. *arXiv preprint arXiv:2507.04590*, 2025.

John O'Keefe and Lynn Nadel. *The Hippocampus as a Cognitive Map*. Oxford University Press, 1978.

OpenAI. Introducing GPT-5.5. `https://openai.com/index/introducing-gpt-5-5/`, 2026.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, Yan Lu, Jenq-Neng Hwang, and Gaoang Wang. MovieChat: From dense token to sparse memory for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

Abdelrhman Werby, Chenguang Huang, Martin Büchner, Abhinav Valada, and Wolfram Burgard. Hierarchical open-vocabulary 3D scene graphs for language-grounded robot navigation. In *First Workshop on Vision-Language Models for Navigation and Manipulation at ICRA 2024*, 2024.

Shusheng Yang, Jihan Yang, Pinzhi Huang, Ellis Brown, Zihao Yang, Yue Yu, Shengbang Tong, Zihan Zheng, Yifan Xu, Muhan Wang, Daohan Lu, Rob Fergus, Yann LeCun, Li Fei-Fei, and Saining Xie. Cambrian-S: Towards spatial supersensing in video. *arXiv preprint arXiv:2511.04670*, 2025a.

Yuncong Yang, Han Yang, Jiachen Zhou, Peihao Chen, Hongxin Zhang, Yilun Du, and Chuang Gan. 3D-Mem: 3D scene memory for embodied exploration and reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025b.

Frances A Yates. *The Art of Memory*. University of Chicago Press, 1966.

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

Haoji Zhang, Yiqin Wang, Yansong Tang, Yong Liu, Jiashi Feng, Jifeng Dai, and Xiaojie Jin. Flash-VStream: Memory-based real-time understanding for long video streams. *arXiv preprint arXiv:2406.08085*, 2024a.

Haokai Zhang, Yuhang Ding, Yunshu Zhou, Xinze Du, Shengtao Zhang, Zhiyue Zhao, Yuling Xi, and Hao Chen. Spatial memory agent: Experience-grounded procedure memory for spatial intelligence. *arXiv preprint arXiv:2608.12743*, 2026.

Juexiao Zhang, Gao Zhu, Sihang Li, Xinhao Liu, Haorui Song, Xinran Tang, and Chen Feng. Multi-view scene graph. *Advances in Neural Information Processing Systems*, 37:17761–17788, 2024b.

A Implementation and Evaluation Details

Baselines. The streaming Recall baselines are from Cambrian-S's Appendix Table 17 (Yang et al., 2025a), using its LFP-finetuned Cambrian-S-7B backbone. WorldMM uses a Qwen3-VL-4B for its memory, consistent with our method. Its Qwen3-VL-4B and GPT-5.5 readers share the same memory.

Recall. Memory construction is query-blind. For multi choices question answering, Qwen readers tend to bias towards the first option, so we use simple chain of thoughts (CoT) to let it reason first and then answer, and also apply a schema to ensure getting a structured answer.

Counting. Cambrian-S uses surprise thresholds of 0.39 for 10/30/60 minutes and 0.41 for 120 minutes, which we followed. We followed all the hyperparameters same as the release and we got higher numbers on the 120mins counting split, which we reported. Long segments are fitted to 128 frames. Place matching uses $(\tau, \rho) = (0.75, 0.90)$, selected on the 10-minute split and fixed elsewhere. Each counter's segment inputs and counts are held fixed when adding place matching. Our API provider of GPT-5.5 only accepts a maximum of 50 frames in one request, and that is why we set it in the experiments. A longer context should be better if possible. Due to budget constraint, we did not roll out the Segment Sum baseline of GPT 5.5 on all durations after seeing it yielded similar results with subsampling 50 frames. It could still get better performance on other durations if budget allows for future testing. Qwen3-VL-4B and Qwen3-VL-8B use the same integer-count prompt, greedy decoding, and a 16-token output limit; their native visual preprocessing differs from Cambrian-S's.

B Writer Cost

Writes/min is the total number of language-model generations required to build memory divided by the total nominal video minutes across QA/store instances. It includes captioning, summary/inventory updates, and, for WorldMM, multiscale memory and offline-indexing generations. Reader/answerer calls, embedding computation, cache hits, retries, and recovery overhead are excluded. WorldMM starts with an empty response cache for each video; repeated requests within that video count once. This measures model calls, not tokens, FLOPs, or latency.

Calls/min	10 min	30 min	60 min	120 min	240 min
$K = 1$	10.52	9.52	9.17	9.21	10.37
$K = \infty$	7.33	6.62	6.38	6.27	5.82
Reduction (%)	30.4	30.5	30.4	31.9	43.9
WorldMM	40.12	40.97	40.60	42.19	44.24

Pooled rates are 9.86 for $K = 1$, 6.09 for $K = \infty$, and 42.93 for WorldMM (1,184,823 generations over 27,600 nominal minutes). Caption baselines report the construction cost of the shared $K = 1$ store. A store's cost is independent of the reader and is not counted again when another reader reuses it.

C Evaluation Metrics

VSI-Super Recall contains 60 questions at each of five durations (10, 30, 60, 120, and 240 minutes), and Count contains 50 questions at each of four durations (10, 30, 60, and 120 minutes). Recall accuracy is the percentage of correct answers. Invalid or unparseable answers remain in the evaluation denominator. Counting segmentation and place matching are query-independent; only the counter receives the requested object category.

For a counting question i, let y_i denote the ground-truth count and $\hat{y}_i$ the predicted count. Its absolute relative error is

$$e_i = \frac{|\hat{y}_i - y_i|}{y_i}. \tag{1}$$

The official VSI-Super mean relative accuracy (MRA) averages correctness over ten confidence levels $\mathcal{C} = \{0.50, 0.55, \ldots, 0.95\}$:

$$\mathrm{MRA} = \frac{100}{N} \sum_{i=1}^{N} \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \mathbf{1}[e_i \leq 1 - c]\,. \tag{2}$$

MRA changes only at these tolerance boundaries and gives zero credit for $e_i > 0.50$. Mean absolute relative error (MARE),

$$\mathrm{MARE} = \frac{100}{N} \sum_{i=1}^{N} e_i, \tag{3}$$

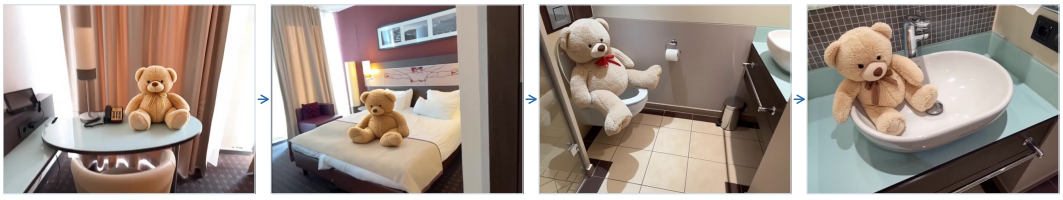
is continuous in the error and is better when lower. All evaluated ground-truth counts are nonzero. Averages weight durations equally; changes are computed before rounding. ΔQA counts predictions that differ from the same counter's Segment Sum baseline, out of 200 questions; it does not count only improvements.

D Additional Qualitative Examples

D.1 Recall from Stored Place Memory

Figure 4 is the example discussed in Section 4.5; Figures 5 and 6 add two more. All use saved deferred Qwen3-VL-4B stores. Bold words highlight the target in verbatim memory excerpts; complete records appear in Appendix E. Images keep the answerer's input order, and scores are SigLIP2 query-to-region-prototype similarities rather than probabilities or calibrated object-presence confidences.

Q. Which of the following correctly represents the order in which the teddy bear appeared in the video?



P1 · 323 s	**P2 · 367 s**	**P3 · 385 s**	**P4 · 407 s**
SigLIP2 score = 0.132	SigLIP2 score = 0.128	SigLIP2 score = 0.124	SigLIP2 score = 0.138
Stored place memory (verbatim excerpts)			
*…round glass table holding a tan **teddy bear…***	*…a large bed, a plush **teddy bear** sitting on it…*	*…a large **teddy bear** sitting on the toilet.…*	*A plush **teddy bear** sits inside a white sink…*
Objects: table; **teddy bear;** …	Objects: bed; **teddy bear;** …	Objects: large beige **teddy bear;** toilet; …	Objects: sink; **teddy bear;** …

Answer: A. Table → Bed → Toilet → Basin **GT: A**

Figure 4: **Recall through stored place memory.** Four retrieved images and verbatim memory excerpts recover the teddy bear's location sequence (10-minute split); Qwen3-VL-4B answers correctly (A). Scores are SigLIP2 query-to-region-prototype similarities. Full records are in Appendix E.

Q. Which of the following correctly represents the order in which the Statue of Liberty appeared in the video?

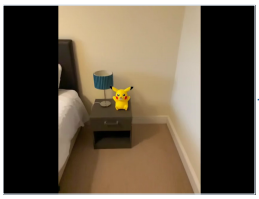
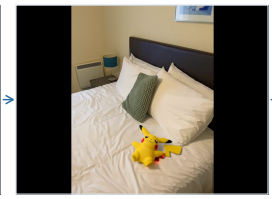
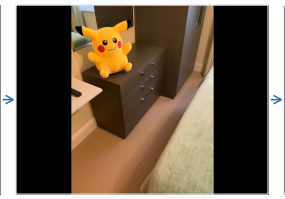
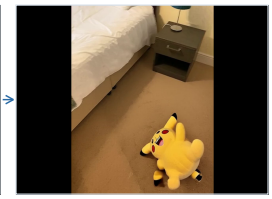


P1 · 61 s	**P2 · 90 s**	**P3 · 97 s**	**P4 · 102 s**
SigLIP2 score = 0.053	SigLIP2 score = 0.063	SigLIP2 score = 0.059	SigLIP2 score = 0.066
Stored place memory (verbatim excerpts)			
*…a green **Statue of Liberty** figurine on a desk…*	*An office chair holds a green **Statue of Liberty** figurine.…*	*A wooden cabinet holds a green **Statue of Liberty** figurine…*	***…Statue of Liberty** figurine sits inside a blue trash can…*
Objects: desk; green **Statue of Liberty;** …	Objects: office chair; **Statue of Liberty;** …	Objects: wooden cabinet; green **Statue of Liberty** figurine; …	Objects: blue trash can; **Statue of Liberty;** …

Answer: B. Table → Chair → Cabinet → Trash can **GT: B**

Figure 5: **Recall across four office locations.** The Statue of Liberty moves Table → Chair → Cabinet → Trash can (30-minute split, video 00000009); Qwen3-VL-4B answers correctly (B).

Q. Which of the following correctly represents the order in which the Pikachu appeared in the video?

P1 · 5073 s	**P2 · 5092 s**	**P3 · 5098 s**	**P4 · 5155 s**
SigLIP2 score = 0.065	SigLIP2 score = 0.040	SigLIP2 score = 0.028	SigLIP2 score = 0.050
			
Stored place memory (verbatim excerpts)			
*...A yellow **Pikachu** plush toy sits on the nightstand....*	*A neatly made bed ... a yellow **Pikachu** plush toy....*	*...a yellow **Pikachu** plush toy sitting on a dark wood dresser....*	*...A yellow **Pikachu** plush toy lies on the carpeted floor.*
Objects: nightstand; **Pikachu** plush toy; …	Objects: bed; **Pikachu** plush toy; …	Objects: dark wood dresser; yellow **Pikachu** plush toy; …	Objects: **Pikachu** plush toy; bed; …

Answer: D. Nightstand → Bed → Chest of drawers → Ground **GT: D**

Figure 6: **Recall from the 240-minute split.** The Pikachu example (video 00000020) is answered correctly (D). The four observations span 5073–5155 s, i.e., 82 seconds rather than four hours; frames keep their original letterboxing.

D.2 PLACE-BASED COUNT ASSOCIATION

The next four examples use the Place-only setting of Table 2, $(\tau, \rho) = (0.75, 0.90)$, and saved Cambrian-S counts. Directional place coverage is the fraction of frames in one segment whose best SALAD cosine match in the other reaches τ. A coverage of at least ρ establishes the corresponding containment relation.

Q. How many different chair(s) are there in the video?

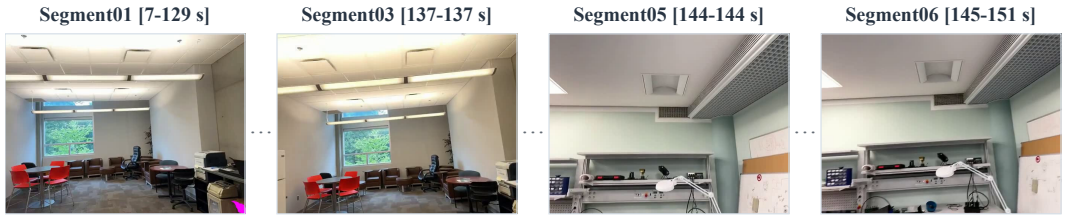


Place coverage: Segment03 → Segment01 = 1.0000
Reverse coverage = 0.0081
Frame cosine (32 s, 137 s) = 0.7772

Local dedup: 12 + 4 → 12 **Full video: 38 (before place dedup.) → 34 | GT 32**

Figure 7: **Removing a repeated chair-count contribution.** Segment01 covers Segment03 and remains the count source, replacing 12+4 with 12; Segment02 keeps its own count. The estimate changes from 38 to 34 (GT 32; 10-minute split, video 00000007). The displayed 32/137 s pair has SALAD cosine 0.777; the segments are 8 s apart.

Q. How many different ceiling light(s) are there in the video?



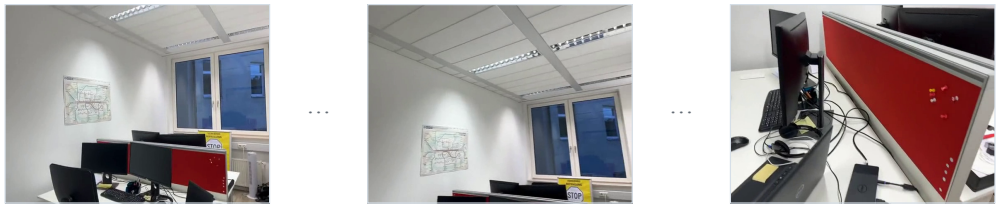
Segment01 + Segment03: 4 + 4 → 4
Place coverage (new → old) = 1.0000
Place coverage (old → new) = 0.0081
Shown pair cosine = 0.7772

Segment05 + Segment06: 2 + 2 → 2
Place coverage (new → old) = 0.1429
Place coverage (old → new) = 1.0000
Shown pair cosine = 0.8544

Local dedup: 4 + 4 → 4; 2 + 2 → 2 **Full video: 41 (before place dedup.) → 35 | GT 31**

Figure 8: **Both directions of place containment.** In video 00000005 (10-minute split), Segment01 covers Segment03 and is retained, while Segment06 covers Segment05 and replaces it, removing 4 and 2 counts (41 → 35, GT 31). Cosines refer to the displayed 32/137 s and 144/146 s frames; the office scene is shared with Figure 7.

Q. How many different monitor(s) are there in the video?



Place coverage: Segment12 → Segment11 = 0.0286 Reverse coverage = 1.0000

Frame cosine (938 s, 941 s) = 0.7552

Local dedup: 4 + 4 → 4 **Full video: 143 (before place dedup.) → 139 | GT 111**

Figure 9: **Count-source replacement under adjacent overlap.** The one-frame Segment11 is covered by Segment12, which becomes the count source (4+4 → 4; 60-minute split, video 00000001). The estimate falls from 143 to 139 (GT 111): absolute error drops from 32 to 28 while per-question MRA stays at 50. The last two images both belong to Segment12.

Q. How many different bed(s) are there in the video?



Place coverage: Segment16 → Segment15 = 0.0081 Reverse coverage = 1.0000

Adjacent overlap: 1 s gap

Local dedup: 2 + 2 → 2 **Full video: 91 (before place dedup.) → 82 | GT 73**

Figure 10: **Place overlap in a 120-minute video.** Segment15 and Segment16 begin one second apart (video 00000041); Segment16 is retained as the count source (2+2 → 2). All merges change the estimate from 91 to 82 (GT 73), of which this pair accounts for 2. The final two frames both belong to Segment16.

D.3 Semantic Revisit Association and Parameter Sensitivity

Figure 11 uses Semantic-only $(\tau, \rho) = (0.75, 0.90)$. Its directional coverage is computed from the saved semantic feature banks, not the displayed frame pairs. The selected return removes 2 of the full-video reduction of 4; the 200-second gap is measured between the compared segment boundaries (197 seconds from the latest member of the old group). At this setting, full-50-question MRA is 47.2, versus 47.0 for Place-only. For this video, the corresponding Place-only run predicts 18.

Figure 12 retains a separate parameter-ablation example: OR combines Place $(0.85, 0.90)$ with Semantic $(0.75, 0.80)$. It depicts the same visually identifiable kitchen as Figure 11. These selected local examples do not establish independent scene coverage or overall superiority of the OR configuration.

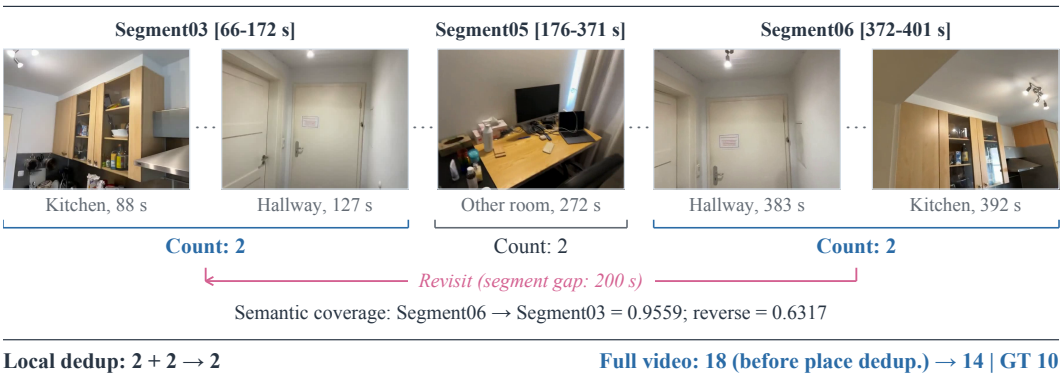


Figure 11: **Deduplicating a scene revisit.** After an intervening room, Segment06 returns to the kitchen and hallway of Segment03 (10-minute split). Association links the two and replaces 2+2 with 2, part of a video-level correction from 18 to 14 (GT 10). Per-segment counts appear below sampled frames.

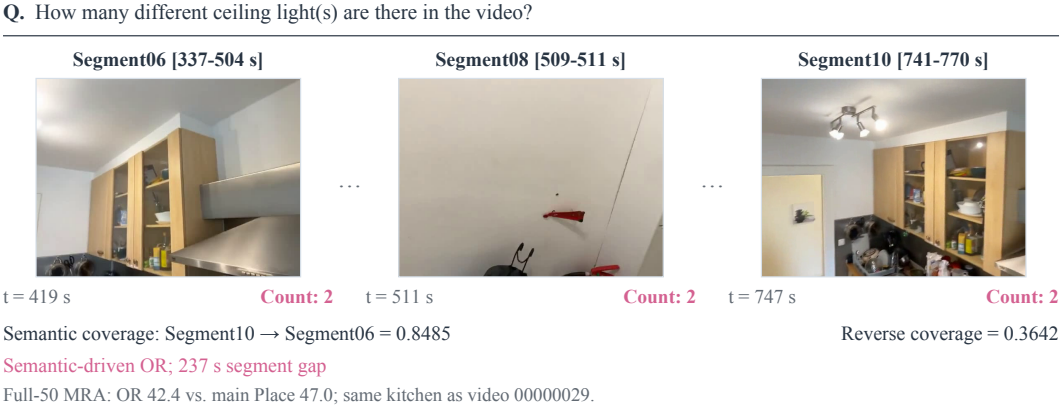


Figure 12: **OR-configuration example from the same kitchen.** In video 00000014 (10-minute split), the semantic branch links Segment10 to Segment06 (coverage 0.85) although both place coverages are zero, removing 2 of the total reduction 30 → 14 (GT 12). This configuration's full-split MRA is 42.4, below Place-only's 47.0.

E COMPLETE STORED MEMORY FOR THE RECALL EXAMPLES

The following summaries and complete object-name inventories reproduce the saved r03 place memory used by the Qwen3-VL-4B r04b Full readout. P1–P4 are display aliases for the original place IDs; they preserve the answerer input order. Summaries aggregate place observations and need not describe every individual frame. Inventory counts are stored memory fields, not physical-instance ground truth. The accompanying JSON preserves all inventory metadata.

E.1 10MIN VIDEO 00000007 TEDDY BEAR

Question. These are frames of a video. Which of the following correctly represents the order in which the teddy bear appeared in the video?

A. Table, Bed, Toilet, Basin

B. Toilet, Table, Bed, Basin

C. Bed, Basin, Table, Toilet

D. Bed, Toilet, Table, Basin

Answer: A; GT: A.

P1 (323 s). *Stored place ID:* `plc_6a9c68a08f54a`.

Place summary. A cozy room with a round glass table holding a tan teddy bear and a telephone. A modern lamp and a chair are nearby, with curtains framing a window. A desk with a tablet and power outlets is visible to the left.

Complete object list. chair (count 1); curtain (count 1); desk (count 1); lamp (count 1); table (count 1); teddy bear (count 1); telephone (count 1).

P2 (367 s). *Stored place ID:* `plc_6a9c85283560b`.

Place summary. A cozy hotel room with a large bed, a plush teddy bear sitting on it, and a chair near the window. The walls are painted a deep red, and there's a framed picture above the bed. A curtain hangs beside a window, and a bedside lamp is illuminated.

Complete object list. bed (count 1); chair (count 1); curtain (count 1); lamp (count 1); pillow (count 1); teddy bear (count 1); wall art (count 1).

P3 (385 s). *Stored place ID:* `plc_6a9c94e6eed67`.

Place summary. A bathroom with a large teddy bear sitting on the toilet. The toilet is open, and a roll of toilet paper hangs on a holder. A sink, trash can, and toilet brush holder are also visible.

Complete object list. large beige teddy bear (count 1); shower door (count 1); sink (count 1); toilet (count 1); toilet brush holder (count 1); toilet paper roll (count 1); trash can (count 1).

P4 (407 s). *Stored place ID:* `plc_6a9ca8593d6a7`.

Place summary. A plush teddy bear sits inside a white sink, with a coin resting beside it. The sink is set into a light blue countertop, and a chrome faucet is mounted above. Behind the sink, small gray tiles line the wall, and a black electrical outlet is visible nearby.

Complete object list. coin (count 1); drawer (count 1); electrical outlet (count 1); faucet (count 1); sink (count 1); teddy bear (count 1); wall tiles (count 1).

E.2 30MIN VIDEO 00000009 STATUE OF LIBERTY

Question. These are frames of a video. Which of the following correctly represents the order in which the Statue of Liberty appeared in the video?

A. Table, Cabinet, Trash can, Chair

B. Table, Chair, Cabinet, Trash can

C. Table, Cabinet, Chair, Trash can

D. Chair, Trash can, Cabinet, Table

Answer: B; GT: B.

P1 (61 s). *Stored place ID:* `plc_1632ff104d5fdb`.

Place summary. An office space with a green Statue of Liberty figurine on a desk, surrounded by office chairs, monitors, and a whiteboard. The room has blue carpeting and a visible blue door leading to another area.

Complete object list. blue door (count 1); blue office chair (count 1); computer monitor (count 1); desk (count 1); green Statue of Liberty (count 1); roll of paper (count 1); whiteboard (count 1).

P2 (90 s). *Stored place ID:* `plc_1633044cf557cc`.

Place summary. An office chair holds a green Statue of Liberty figurine. A desk with a keyboard, headphones, and water bottles is nearby. The room has a carpeted floor and a white wall.

Complete object list. Statue of Liberty (count 1); box (count 1); computer monitor (count 1); desk (count 1); headphones (count 1); keyboard (count 1); office chair (count 1); water bottles (count 1).

P3 (97 s). *Stored place ID:* `plc_16330500a8b387`.

Place summary. A wooden cabinet holds a green Statue of Liberty figurine, a computer monitor, and electronic devices. The setup is in a room with a white ceiling and exposed lighting fixture.

Complete object list. black box (count 1); computer monitor (count 1); green Statue of Liberty figurine (count 1); white router (count 1); wooden cabinet (count 1).

P4 (102 s). *Stored place ID:* `plc_163305a73271da`.

Place summary. A small green Statue of Liberty figurine sits inside a blue trash can lined with a white bag. Beside it is an empty blue bucket. The scene is set against a plain wall on a gray carpeted floor.

Complete object list. Statue of Liberty (count 1); blue bucket (count 1); blue trash can (count 1); gray carpet (count 1).

E.3 240MIN VIDEO 00000020 PIKACHU

Question. These are frames of a video. Which of the following correctly represents the order in which the Pikachu appeared in the video?

A. Chest of drawers, Nightstand, Bed, Ground

B. Bed, Ground, Chest of drawers, Nightstand

C. Chest of drawers, Bed, Nightstand, Ground

D. Nightstand, Bed, Chest of drawers, Ground

Answer: D; GT: D.

P1 (5073 s). *Stored place ID:* `plc_6ff28d2f8419f`.

Place summary. A bedroom corner with a bed, a dark gray nightstand, and a blue-striped lamp. A yellow Pikachu plush toy sits on the nightstand. The walls are painted a light beige color.

Complete object list. Pikachu plush toy (count 1); bed (count 1); lamp (count 1); nightstand (count 1); wall (count 1).

P2 (5092 s). *Stored place ID:* `plc_6ff290fe9849e`.

Place summary. A neatly made bed with white sheets and pillows, featuring a yellow Pikachu plush toy. A nightstand with a blue lamp and an air conditioner sits beside the bed. A framed picture hangs on the wall above the headboard.

Complete object list. Pikachu plush toy (count 1); air conditioner (count 1); bed (count 1); lamp (count 1); nightstand (count 1); picture frame (count 1); pillow (count 1).

P3 (5098 s). *Stored place ID:* `plc_6ff296e3f66a0`.

Place summary. A bedroom with a yellow Pikachu plush toy sitting on a dark wood dresser. A remote control rests on a small white table beside it. A bed with a light-colored bedspread is visible to the right, and a mirror hangs on the wall above the dresser.

Complete object list. bed (count 1); dark wood dresser (count 1); mirror (count 1); remote control (count 1); yellow Pikachu plush toy (count 1).

P4 (5155 s). *Stored place ID:* `plc_6ff2a6b19ce9f`.

Place summary. A bedroom with a white bed, a gray nightstand, and a teal lamp. A yellow Pikachu plush toy lies on the carpeted floor.

Complete object list. Pikachu plush toy (count 1); bed (count 1); lamp (count 1); nightstand (count 1).