Linguistic Trajectory Encoding for Efficient Long-Horizon Spatial Memory in Embodied Agents

Xie Tianyidan[1] **Shenyi Wang**[1] **Qiang Tang**[2] **Mingjie Wang**[3] **Zhicheng Qiu**[4]
Xuanfu Li[4] **Zhan Xu**[4] **Jian Yang**[1] **Lanjun Wang**[5*] **Zili Yi**[1*]

[1]Nanjing University [2]University of British Columbia
[3]Zhejiang Sci-Tech University [4]Huawei Technologies Ltd. [5]Tianjin University

Abstract

Embodied agents performing long-horizon tasks require a memory representation in which the state transitions of dynamic objects remain queryable in natural language across hours-to-days observation horizons. Existing systems either drop fine-grained motion (clip-level video-language embeddings), keep it only as raw coordinates (geometric SLAM), or organise it around immediate task context (agent working memories). None of them gives the agent a per-object timeline whose state transitions are themselves queryable in language. Our key contribution is **Linguistic Trajectory Encoding** (LTE), which compresses dynamic object motion histories via a hybrid representation combining natural language descriptions, sparse spatial anchors, and visual anchors. LTE adapts compression to motion complexity by anchoring periods without reliable observations to the last seen location, while representing motion with geometric waypoints and linguistic descriptions to preserve accuracy. To evaluate these capabilities across extended time horizons, we construct the **Spatial Memory Benchmark** (SMB) from EgoLife multi-day recordings, targeting capabilities absent in existing benchmarks: semantic trajectory retrieval and long-horizon object retrieval. On SMB, the LTE-based system achieves 45.3% success in semantic trajectory retrieval and 48.7% in long-horizon object retrieval, outperforming structured-memory and VLM baselines (best prior: 31.9% and 34.4%). LTE achieves trajectory compression by factors of $8.7\times$ to $26.1\times$ with sub-second query latency on 24 h video. On Ego4D natural-language queries, the system reaches 28.75% / 55.10% R@1/R@5, $+15.80$ / $+31.30$ pts over EgoVLPv2.

1 Introduction

Consider an embodied agent that has continuously observed a kitchen for six hours when the user asks: *"Where did I last put the washed apple?"* To answer, the agent must recover, from a long observation history, an object named in language (*"the apple"*), a state change described semantically (*"washed"*), and a location at a specific moment in time (*"where it ended up"*). A single query thus forces three retrieval modes to operate together over a memory that may span hours to days.

Existing memory systems each handle some of these modes but are limited on others. Geometric SLAM [Rosinol et al., 2021, Schmid et al., 2024] maintains persistent 3D state over time but exposes no interface for natural-language predicates such as *"washed"*. The system can tell us where an object ended up but not whether it was washed there. Video-language models [Lin et al., 2022, Pramanick et al., 2023, Pei et al., 2024] match free-form text against clip-level features but lack a 3D index, *"where did I last put it"* degenerates into an exhaustive temporal scan with no spatial pruning. Memory-augmented video agents [Fan et al., 2024, Goletto et al., 2024] organise observations into temporal events (e.g., a single *"kitchen activity"* span) but do not maintain a per-object trajectory,

*Corresponding authors: Zili Yi (yi@nju.edu.cn) and Lanjun Wang (wanglanjun@tju.edu.cn).

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

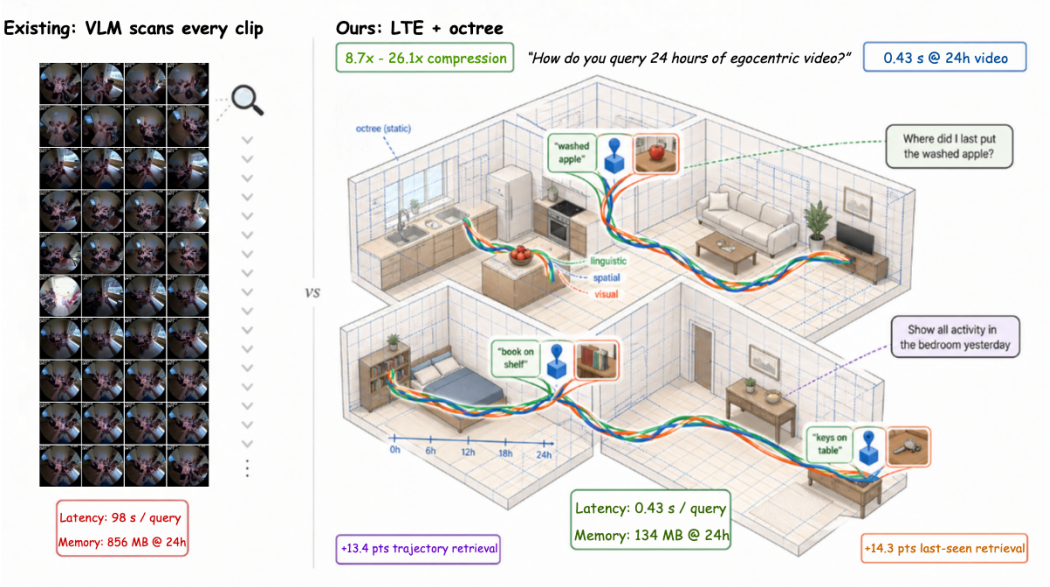

Figure 1: **Our memory system at a glance.** A 24 h egocentric day is encoded into per-object Linguistic Trajectory Encoding (LTE) records, each braided from three channels: linguistic captions ($\mathcal{C}$, green), sparse spatial anchors ($\mathcal{A}_{\text{spatial}}$, blue), and visual anchors ($\mathcal{A}_{\text{visual}}$, terracotta). An octree provides room-level spatial pruning. Natural-language queries traverse only the relevant channel(s) and recover both the location and the moment, in 0.43 s, instead of 98 s on 24 h video.

so the apple's individual history is folded into the broader event and lost. Task-oriented agent memories [Sarch et al., 2023, Wang et al., 2025] connect language with short-horizon spatial state but organise records around the current task. The apple, incidental to whatever task was active, leaves no queryable record at all. The shared shortfall lies in how dynamic objects are recorded: their motion is either dropped (clip embeddings), kept as raw 3D coordinates without semantic content (SLAM), or summarised only when task-relevant. **None of these systems gives the agent a per-object timeline whose state transitions are themselves queryable in language.**

We address this gap with **Linguistic Trajectory Encoding** (LTE), a per-object trajectory representation designed for the dynamic-object memory problem above. For each tracked object, LTE records its history through three channels: (i) a sequence of natural-language captions describing motion phases (e.g., *"apple moved from sink to countertop"*), (ii) sparse 3D positions retained at trajectory inflections via Douglas-Peucker simplification, and (iii) visual crops at the same anchors for identity verification. Each channel covers a query type that the others cannot: captions match state predicates such as *"washed"*, spatial anchors answer *"where"* in 3D, and visual crops verify *"which apple"* across motion phases. Compression is adaptive: tracking-gap intervals collapse to a single anchor, simple motions retain few waypoints, and complex ones retain more, yielding compression by factors of $8.7\times$ to $26.1\times$ relative to dense per-frame storage. Answering the apple query then comes down to a single caption match against *"washed"*, after which the linked 3D location and visual crop are returned (Figure 1). Furthermore, to deploy LTE on hours of egocentric video, we wrap it in a memory architecture (Figure 2) with two supporting elements: an octree spatial index providing region pruning for queries scoped to a room or area, and five complementary views (Object, Scene, Text, Event, Image) that reuse the same per-object LTE records under different access patterns.

To evaluate the LTE-based system in the long-horizon regime that motivated it, we construct the **Spatial Memory Benchmark** (SMB) from EgoLife [Yang et al., 2025b]: 600 queries over multi-day recordings, with individual sessions reaching 50 h. SMB introduces two new tasks, Semantic Trajectory Retrieval (STR) and Long-Horizon Object Retrieval (LOR), complementing the established Ego4D Natural Language Queries (NLQ) and Visual Queries 2D (VQ2D) benchmarks [Grauman et al., 2022]. On SMB, the LTE-based system achieves 45.3% (STR) and 48.7% (LOR), versus 31.9% and 34.4% for the strongest VLM baseline and 24.7% and 33.8% for structured-memory baselines. On Ego4D NLQ (IoU$=0.3$), it reaches 28.75% / 55.10% R@1/R@5, the strongest zero-shot result.

Contributions.

- **Linguistic Trajectory Encoding (LTE):** a per-object hybrid representation that records dynamic motion as language-described phases anchored to sparse 3D positions and visual crops, achieving trajectory compression by factors of $8.7\times$ to $26.1\times$ with sub-second query latency on $24\,\text{h}$ video.
- **Spatial Memory Benchmark (SMB):** 600 queries built on EgoLife multi-day recordings, targeting two long-horizon capabilities (semantic trajectory retrieval and long-horizon last-occurrence retrieval) that are absent from existing benchmarks.
- **Long-horizon retrieval gains:** the integrated system surpasses the strongest VLM baseline on both new SMB tasks ($+13.4$ on Semantic Trajectory Retrieval, $+14.3$ on Long-Horizon Object Retrieval), exceeds all structured-memory baselines, ranks first among zero-shot methods on the established Ego4D NLQ and VQ2D benchmarks, and approaches task-specific supervised systems on both.

2 Related Work

LTE relates to three lines of work: spatial memory representations for embodied AI, object trajectory encoding, and long-horizon video understanding. None gives a per-object timeline indexed in language. We summarise each line below and locate LTE relative to it.

Spatial memory and scene representation. Static-dynamic decomposition in SLAM [Schmid et al., 2022, 2024] separates static backgrounds from moving objects for robust localisation, and open-vocabulary 3D scene graphs [Gu et al., 2024, Werby et al., 2024] attach VLM-generated semantic attributes to per-object nodes. Both maintain rich spatial structure but treat each object's semantics as a static snapshot at observation time, not as a temporal sequence of state changes. Keyframe-based memories such as 3D-Mem [Yang et al., 2025c] and KARMA [Wang et al., 2025] subsample observations at fixed or co-visibility-driven intervals, breaking temporal continuity. We adopt octree indexing for static geometry as a supporting element and rely on LTE for the temporal dimension that these works do not address.

Trajectory representation. Geometric trajectory compression [Douglas and Peucker, 1973] and semantic trajectory mining [Yan et al., 2013, Zheng, 2015] reduce coordinate sequences to waypoints or stop-move segments, but they operate purely on numerical or categorical labels without language-grounded abstraction over motion phases. Memory-augmented video agents such as AMEGO [Goletto et al., 2024] and VideoAgent [Fan et al., 2024] organize observations into temporal events or hand-object interaction tracklets, providing event-level rather than per-object linguistic structure. LTE occupies the position that none of these works fills: a per-object timeline whose successive motion phases are themselves indexed in language, anchored to sparse 3D positions and visual crops for grounding.

Long-horizon video. Ego4D [Grauman et al., 2022] supplies short ($\sim 8\,\text{min}$) natural-language query tasks. EgoLife [Yang et al., 2025b] provides continuous multi-day recordings ($300\,\text{h}$) for long-horizon evaluation. Video-language models [Pramanick et al., 2023] and zero-shot VLMs [Bai et al., 2025] lack explicit spatial structures or trajectory abstractions and lose global context across sequential clips. We address both gaps through spatial indexing plus per-object linguistic trajectories.

3 Method

Section 3.1 formalizes the problem and presents the system overview. Section 3.2 describes perception components. Section 3.3 details Linguistic Trajectory Encoding (the dynamic-object representation) and the octree spatial index. Section 3.4 explains multi-view organization. Section 3.5 presents query-processing pipelines.

3.1 Problem Setup and Architecture Overview

Problem setup. An embodied agent observes an environment continuously through a video stream $V = \{I_t\}_{t=1}^{T}$, where $I_t \in \mathbb{R}^{H\times W\times 3}$ is the RGB frame captured at time t, potentially accumulating $T > 10^5$ frames over hours to days. The agent constructs a queryable memory structure $\mathcal{M}$ supporting

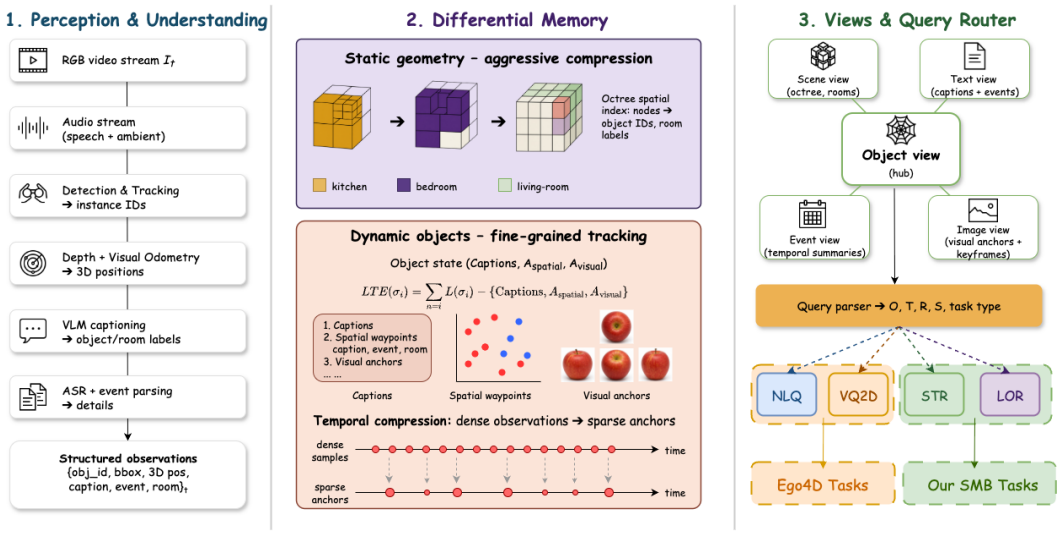

Figure 2: **LTE-centric memory architecture.** (1) Perception (left) converts egocentric video into structured per-frame observations. (2) The memory layer (centre) records dynamic objects with Linguistic Trajectory Encoding (captions $\mathcal{C}$, sparse spatial anchors $\mathcal{A}_{\text{spatial}}$, and visual anchors $\mathcal{A}_{\text{visual}}$) and uses an octree as a static spatial index for region pruning. (3) Five complementary views feed a query router that dispatches to four tasks: **NLQ** and **VQ2D** from the Ego4D benchmark suite, and **STR** and **LOR** introduced in this paper as part of SMB.

spatial queries specifying 3D regions, temporal queries over intervals, and semantic queries using natural language.

System overview. The LTE-based system exploits temporal heterogeneity in spatial data. Figure 2 illustrates the overall architecture: perception components process video streams into structured observations, memory views organize information through complementary indexing strategies, and query processing routes requests to appropriate views based on parsed constraints.

3.2 Perception and Scene Understanding

The system processes video through detection, tracking, and depth-estimation pipelines. Object detection and tracking (SAM3 [Carion et al., 2025]) maintain instance identities across frames. Monocular depth estimation combined with visual odometry (ViPE [Huang et al., 2025]) reconstructs 3D positions by back-projecting 2D detections into world coordinates. A vision-language model (Qwen3-VL [Bai et al., 2025]) provides room-level semantic labels (kitchen, bedroom, etc.) and object-centric motion descriptions. Speech recognition (Whisper [Radford et al., 2023]) transcribes audio, which a language model (Qwen3-8B [Yang et al., 2025a]) parses into temporal events. These perception outputs feed into memory construction.

3.3 Linguistic Trajectory Encoding and Spatial Indexing

Linguistic Trajectory Encoding for dynamic objects. Dynamic objects are the central challenge: maintaining dense, per-timestep coordinate sequences over hours of multi-object tracking is unwieldy for indexing and semantic state matching. We introduce Linguistic Trajectory Encoding (LTE) to summarise motion histories in a query-oriented hybrid form while preserving semantic accessibility and spatial grounding (Figure 2, lower-centre panel).

Core insight. Many object motions admit concise semantic descriptions: *"apple moved from sink to countertop"* captures essential semantics while preserving queryability. However, pure text loses the spatial precision needed for geometric queries. LTE resolves this through a hybrid representation: motion phases described linguistically, critical spatial positions retained as anchors, and visual snapshots at anchors for identity verification.

Formal definition. For object o_i with trajectory $\tau_i = \{(\mathbf{p}_t, t)\}_{t=1}^{T_i}$, LTE produces:

$$\text{LTE}(o_i) = (\mathcal{C}, \mathcal{A}_{\text{spatial}}, \mathcal{A}_{\text{visual}}) \tag{1}$$

where $\mathcal{C} = \{(c_j, [t_j^{\text{start}}, t_j^{\text{end}}])\}$ are interval captions with temporal spans, $\mathcal{A}_{\text{spatial}} = \{\mathbf{p}_k\}_{k \in K}$ are 3D spatial anchor positions, and $\mathcal{A}_{\text{visual}} = \{(I_{t_k}, b_{t_k})\}_{k \in K}$ are visual anchors storing frame crops and bounding boxes.

Construction process. For each detected object o_i, we maintain a numeric trajectory τ_i by back-projecting bounding-box centres to 3D world coordinates. Motion state is determined by tracking continuity. Let $m_t \in \{0, 1\}$ indicate whether o_i is detected at timestamp t, and let $t^-(t) = \max\{t' \leq t \mid m_{t'} = 1\}$. We define **tracking-gap intervals** as maximal spans where the track is missing for at least θ_{static} seconds, i.e., $m_t = 0$ and $t - t^-(t) \geq \theta_{\text{static}}$; during such spans we anchor the object to the last observed position $\mathbf{p}_{t^-(t)}$. All remaining spans are treated as motion intervals. For each interval $[t_j^{\text{start}}, t_j^{\text{end}}]$, the VLM generates caption c_j describing the object's motion within scene context through visual tagging [Yang et al., 2023].

Spatial anchors $\mathcal{A}_{\text{spatial}}$ are placed adaptively based on 3D trajectory geometry. For tracking-gap intervals, we store a single 3D position. For motion intervals, we apply Douglas-Peucker simplification [Douglas and Peucker, 1973], retaining waypoints where the 3D trajectory deviates significantly from linear interpolation. Visual anchors $\mathcal{A}_{\text{visual}}$ store frame crops at informative timestamps: for tracking-gap intervals, we place anchors only at observable boundaries (the last detected frame before the gap and the first re-detected frame after the gap if available); for motion intervals we sample more densely to capture trajectory dynamics. This provides visual evidence for identity verification and enables visual-query matching through embedding similarity.

This hybrid representation supports multiple query types: semantic queries match caption text; spatial queries test anchor positions against query regions; visual queries compute embedding similarity against visual anchors; and temporal queries interpolate between adjacent anchors, with linguistic context providing motion semantics.

Octree spatial indexing for static geometry. As a supporting element to LTE, we treat the octree as a static spatial partition over the reconstructed scene; nodes index object identifiers (and their anchors) to enable region-based pruning. The observed spatial extent $\Omega \subset \mathbb{R}^3$ is recursively subdivided to a maximum depth $d_{\max}$. Each node n at depth d covers region Ω_n and stores identifiers of objects whose 3D centres fall within its bounds. Room-level semantic labels are associated with octree regions through VLM inference on representative frames. Top-down traversal answers a query region Q by pruning when $Q \cap \Omega_n = \emptyset$, collecting all descendants when $Q \supseteq \Omega_n$, and recursing otherwise.

3.4 Memory Organization

We organize information into five complementary views with bidirectional links. The **Scene view** provides octree spatial indexing with VLM-inferred room-level labels at each node. The **Object view** maintains entity-centric records (category, identity, status) and links to LTE representations. The **Text view** aggregates LTE motion captions, room labels, and parsed speech transcripts behind a vector index. The **Event view** indexes temporal intervals with activity summaries derived from object-centric action captions and speech events. The **Image view** stores object-centric visual anchors $\mathcal{A}_{\text{visual}}$ plus scene-level adaptive keyframes sampled by optical-flow magnitude. Object records link to containing octree nodes, captions, and events; octree nodes link back to contained objects; events link to participants and text snippets. This supports cascaded filtering across views. Incremental updates reuse the reconstructed 3D coordinate frame, bypassing depth re-estimation (Appendix G.1).

3.5 Query Processing

The system supports four query tasks from two provenances. **Natural Language Queries (NLQ)** and **Visual Queries 2D (VQ2D)** are established Ego4D benchmarks: NLQ requires localising a temporal interval given a free-form question, VQ2D requires retrieving the most recent frame containing an object specified by a visual crop. **Semantic Trajectory Retrieval (STR)** and **Long-Horizon Object Retrieval (LOR)** are introduced as part of SMB to evaluate state-conditioned retrieval and last-occurrence retrieval, respectively, over multi-day horizons absent from Ego4D. A natural-language query q is parsed by Qwen3-8B into structured constraints: objects $\mathcal{O}$, time $\mathcal{T}$, spatial regions $\mathcal{R}$, semantics $\mathcal{S}$, plus a query-type label that routes to specialised pipelines (Figure 2, right-bottom).

All four tasks share a unified cascaded-filtering pattern (parse $\rightarrow$ route $\rightarrow$ filter $\rightarrow$ aggregate); full execution details appear in Appendix G.3.

NLQ. The primary path targets the Object view: given mention X we retrieve the entity record and its LTE; timeline and spatial history come from $\mathcal{C}$ and $\mathcal{A}_{\text{spatial}}$. Event constraints Y trigger Event-view lookup with cross-checking against Text-view captions. Intervals are filtered by spatial (octree) and temporal (LTE timestamps) constraints, and then merged.

VQ2D. A DINOv2 ViT-L/14 [Oquab et al., 2023] embedding of the query crop is matched (cosine ≥ 0.7) against visual anchors $\mathcal{A}_{\text{visual}}$ across all objects, temporally filtered to the most recent occurrence before T.

STR. The state description S is embedded with Qwen3-Embedding-8B [Zhang et al., 2025] and matched against LTE captions $\mathcal{C}$ in the Text view (cosine > 0.8), after octree pruning by $\mathcal{R}$ and $\mathcal{T}$. Matched captions resolve to object ID, temporal span, and spatial evidence.

LOR. Object-view entities are filtered by activity in $[T - \Delta t, T]$ and (optionally) by $\mathcal{R}$ via octree traversal; the most recent LTE anchor timestamp $t^* \leq T$ and its bounding box are returned.

Query complexity depends on constraint types rather than video duration, avoiding exhaustive frame scans. Each response carries an explicit confidence label (HIGH/MEDIUM/LOW) distinguishing in-window observations from extrapolated last-known positions and historical-only data, surfacing off-camera uncertainty to downstream agents (Appendix F).

4 Experiments

4.1 Experimental Setup

Datasets. Ego4D [Grauman et al., 2022] provides established temporal-semantic reasoning tasks. The Natural Language Queries (NLQ) task contains $5{,}462$ queries over 303 validation videos (avg. 8.2 min). We report Recall at rank k at IoU thresholds 0.3 and 0.5. The Visual Queries 2D (VQ2D) task provides $4{,}500$ queries across $1{,}200$ validation clips, evaluated via spatiotemporal AP (stAP), temporal AP (tAP), Success rate (IoU ≥ 0.5), and Recovery rate. To evaluate long-horizon spatiotemporal memory, we construct the **Spatial Memory Benchmark** (SMB) from EgoLife [Yang et al., 2025b], which provides continuous multi-day recordings (7 days, 300 total hours across 6 participants, individual sessions reaching 50 h) in persistent home environments.

SMB comprises two tasks with 300 queries each. **Semantic Trajectory Retrieval (STR)** requires locating objects based on their motion state and trajectory within specified temporal windows and semantic spatial regions; we construct 300 queries on 232 object instances with verified ground-truth state labels and spatiotemporal evidence. **Long-Horizon Object Retrieval (LOR)** requires retrieving the last occurrence of described objects within temporal lookback windows ranging from 2 to 24 hours; we construct 300 queries with verified ground-truth evidence frames. Both tasks evaluate success: whether the system retrieves frames within the object's ground-truth temporal span with bounding-box IoU ≥ 0.3. Construction details, IoU ≥ 0.5 results, and per-horizon breakdowns appear in Appendix A and D.

Baselines. For Ego4D NLQ we compare against supervised methods EgoVLPv2 [Pramanick et al., 2023], GroundNLQ [Hou et al., 2023], EgoVideo [Pei et al., 2024], and OSGNet [Feng et al., 2025], plus the zero-shot method VideoAgent [Fan et al., 2024]. For VQ2D we compare supervised VQLoC [Xu et al., 2023] and PRVQL [Fan et al., 2025], and zero-shot RELOCATE [Khosla et al., 2025]. For SMB we compare against two families of baselines, each probing a distinct alternative to LTE. The first tests whether a strong general-purpose VLM augmented with an open-vocabulary detector is sufficient for long-horizon retrieval: *Qwen3-VL-8B-Instruct* and *Qwen3-VL-235B-A22B-Instruct* [Bai et al., 2025], both paired with Grounding-DINO [Liu et al., 2024] under a 2 s sliding-clip protocol that scans the video at query time. The second tests whether existing structured-memory paradigms suffice without per-object linguistic trajectories: *Keyframe Memory* (KFMem) follows the 3D-Mem [Yang et al., 2025c] snapshot design, using DINOv2 diversity-based keyframe selection with 3D position storage and VLM description matching, and *VideoAgent* [Fan et al., 2024] uses event-segmented memory with VLM querying. Implementation details (models, hyperparameters, compute) and baseline protocols appear in Appendix B and G.4. We also provide an Ascend NPU implementation, with deployment details in Appendix B.1.

Table 1: **Ego4D NLQ validation set.** *Take-away:* the LTE-based system reaches $28.75\%/55.10\%$ R@1/R@5 at IoU=0.3, improving substantially over video-language baselines (EgoVLPv2: $12.95/23.80$) and approaching task-specific supervised models. Structured object timelines and event indexing close most of the gap to supervised methods on temporal localisation.

Method	IoU=0.3		IoU=0.5	
	R@1	R@5	R@1	R@5
Supervised				
EgoVLPv2 [Pramanick et al., 2023]	12.95	23.80	7.91	16.11
GroundNLQ [Hou et al., 2023]	27.20	54.42	18.91	39.98
EgoVideo [Pei et al., 2024]	28.65	53.30	19.73	40.42
OSGNet [Feng et al., 2025]	**32.56**	**59.82**	**22.74**	**46.35**
Zero-shot				
VideoAgent [Fan et al., 2024]	17.39	33.05	7.47	15.73
Ours	28.75	55.10	20.31	42.63

Table 2: **Ego4D VQ2D.** *Take-away:* object-centric visual anchors enable direct identity matching, yielding the strongest zero-shot performance on all four metrics.

Method	stAP	tAP	Succ.	Rec.
Supervised				
VQLoC	0.22	0.31	55.9	47.1
PRVQL	0.27	0.35	57.9	47.9
Zero-shot				
RELOCATE	0.33	0.41	58.0	50.5
Ours	**0.36**	**0.43**	**59.5**	**51.2**

Table 3: **Spatial Memory Benchmark.** STR/LOR success rate (%) on 300 queries each. Q3VL-8B/235B+GD denote Qwen3-VL augmented with Grounding-DINO; KFMem is a 3D-Mem-style structured-memory baseline. *Take-away:* the LTE-based system delivers $+13.4$ STR / $+14.3$ LOR over the strongest VLM baseline.

Method	STR (%)	LOR (%)
VLM baselines (clip-scanning)		
Q3VL-8B+GD	21.5	25.1
Q3VL-235B+GD	31.9	34.4
Structured-memory baselines		
KFMem (3D-Mem-style)	19.8	33.8
VideoAgent [Fan et al., 2024]	24.7	30.5
Ours	**45.3**	**48.7**

4.2 Results on Ego4D Benchmarks

Table 1 presents NLQ results. At IoU=0.3 the LTE-based system reaches $28.75\%/55.10\%$ R@1/R@5, improving substantially over video-language baselines (EgoVLPv2: $12.95\%/23.80\%$) and approaching task-specific supervised models (OSGNet: $32.56\%/59.82\%$). This suggests that explicitly structured object timelines and event-indexed retrieval recover much of the benefit needed for temporal localization, while the remaining gap is consistent with supervised boundary refinement in specialized systems.

Table 2 shows VQ2D results. We obtain the strongest zero-shot performance (0.36 stAP and 59.5% success), slightly exceeding the prior zero-shot baseline RELOCATE (0.33 stAP and 58.0% success). This aligns with LTE storing object-centric visual anchors that enable direct identity matching followed by temporal filtering, avoiding exhaustive frame-level scans.

4.3 Spatial Memory Benchmark

Table 3 presents SMB results. On STR, the LTE-based system achieves 45.3% success, outperforming VLM and structured-memory baselines (Q3VL-235B+GD: 31.9%, VideoAgent: 24.7%, KFMem: 19.8%). KFMem achieves reasonable LOR (33.8%) through spatial keyframe lookup but fails on STR (19.8%) because it cannot represent motion histories. VideoAgent's event-based memory improves over raw VLM ($+3.2$ STR / $+5.4$ LOR) but lacks trajectory-level abstraction. Our advantage on STR ($+20.6$ over KFMem) confirms that trajectory encoding is essential for semantic state queries.

On LOR, the LTE-based system achieves 48.7%, outperforming all baselines. Per-horizon analysis (Appendix D) shows that our performance degrades mildly as the lookback window grows (51.2%

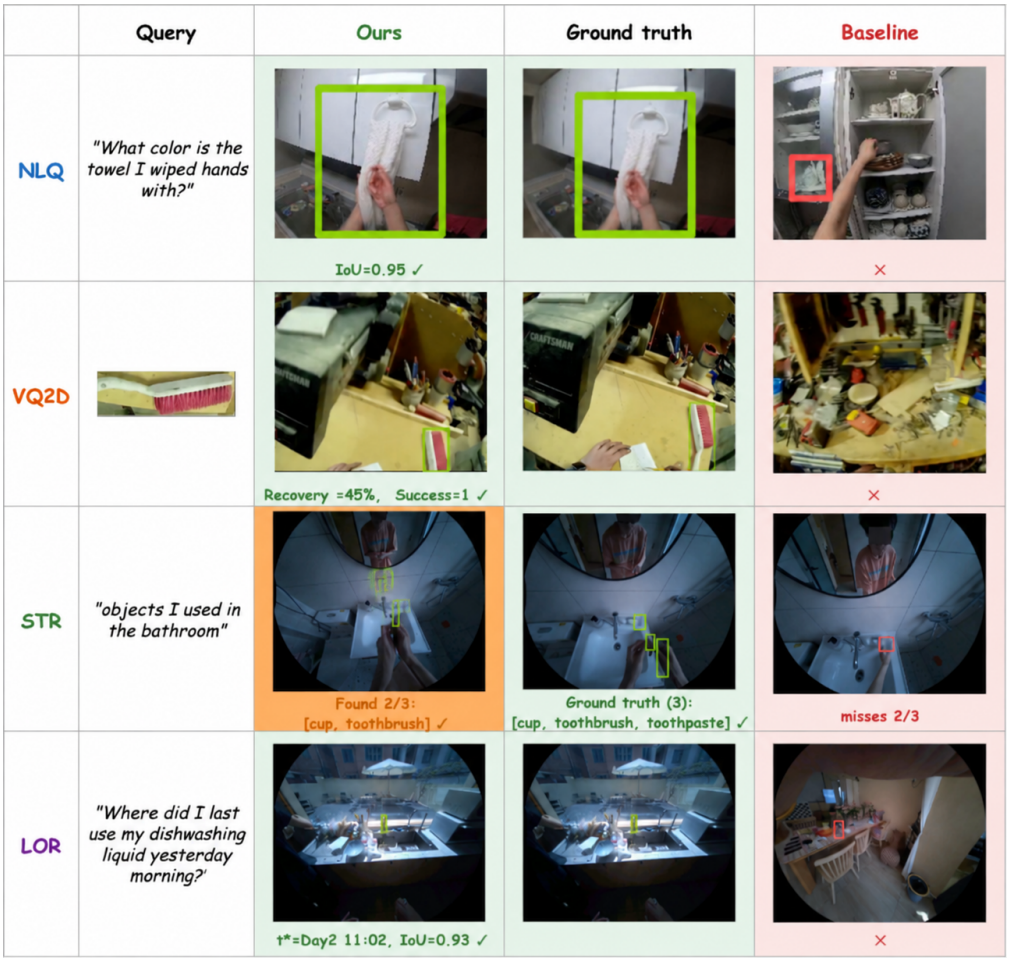

Figure 3: **Qualitative examples across the four tasks.** Each row compares our system against the strongest competing method with one success and one diagnostic failure/partial case. Green=success, amber=partial, red=failure. *Take-away:* most failures originate upstream of LTE rather than from the memory representation; tracking ID-switches alone account for 34% of STR errors (Appendix E).

at 2 h to 46.8% at 24 h), whereas VLM baselines drop more sharply (Q3VL-235B+GD: 39.1% to 29.3%). At the stricter IoU ≥ 0.5 threshold the gap is preserved (+12.4 STR, +13.5 LOR; full table in Appendix D), confirming the result is not an artefact of a lenient threshold.

Figure 3 presents qualitative examples across the four tasks, including success and partial/failure cases. Common failure modes are dominated by tracking errors (ID switches and lost tracks); a full categorisation appears in Appendix E.

Efficiency analysis. Table 4 reports memory footprint and per-query latency on a single A800 GPU. Memory grows sub-linearly (45 $\rightarrow$ 134 MB for 12$\times$ more video) because LTE compression becomes more effective as tracking gaps lengthen. Query latency remains sub-second (0.43 s at 24 h) via octree pruning, while the VLM baseline scales linearly (98.3 s at 24 h). For 300 SMB queries on 24 h video, our total query time is 2.2 minutes vs. 8.2 hours for the VLM. End-to-end timing breakdown is in Appendix C.

Ablation studies. Table 5 reports a fine-grained ablation. Removing all of LTE causes the largest STR drop (-16.5); removing only text captions accounts for -11.8 STR, isolating linguistic abstraction as the dominant contributor. Removing only visual anchors drives the largest VQ2D loss (-6.1); removing the octree alone reduces STR/LOR by $-3.2/-3.5$. LOR is approximately invariant to LTE (relies solely on Object view + octree), validating the modular design. The three LTE channels

Table 4: **Efficiency on a single A800 GPU.** Memory footprint and per-query latency vs. video duration. Compression ratio compares LTE-only storage against a dense per-frame trajectory store. *Take-away:* memory grows sub-linearly and query latency stays sub-second up to 24 h, while the VLM baseline scales linearly and crosses 1.5 min/query.

Dur.	#Obj.	Dense (MB)	Total (MB)	LTE (MB)	Compr.	Ours (s/q)	Q3VL-235B+GD (s/q)
2 h	∼18	71	45	8.2	8.7×	0.15	8.2
6 h	∼35	214	73	16.1	13.3×	0.24	24.5
12 h	∼52	428	98	23.5	18.2×	0.32	49.1
24 h	∼78	856	134	32.8	26.1×	0.43	98.3

Table 5: **Fine-grained ablation.** Removing one component at a time across NLQ R@5 (%), VQ2D Success (%), SMB-STR Success (%), and SMB-LOR Success (%). Δ is the change relative to the full system. *Take-away:* text captions drive STR (−11.8), visual anchors drive VQ2D (−6.1), the octree drives LOR (−3.5); LOR is invariant to LTE.

	Absolute (%)				Δ vs. Ours			
Configuration	NLQ	VQ2D	STR	LOR	NLQ	VQ2D	STR	LOR
Ours (full)	55.10	59.5	45.3	48.7	–	–	–	–
w/o text captions	53.40	58.9	33.5	48.7	−1.70	−0.6	−11.8	0.0
w/o spatial anchors	54.30	57.1	41.2	48.7	−0.80	−2.4	−4.1	0.0
w/o visual anchors	54.70	53.4	43.9	48.7	−0.40	−6.1	−1.4	0.0
w/o LTE (all removed)	52.31	51.8	28.8	48.7	−2.79	−7.7	−16.5	0.0
w/o octree	54.10	58.7	42.1	45.2	−1.00	−0.8	−3.2	−3.5
w/o LTE & octree	51.80	51.2	26.3	44.9	−3.30	−8.3	−19.0	−3.8

combine super-additively on STR: keep-only-one variants give individual gains of +7.4 (text), +2.7 (spatial), and +1.3 (visual) over the no-LTE baseline (Appendix D, Table 12), summing to +11.4, while the full system delivers +16.5. The +5.1-pt gap reflects channel interaction: each channel becomes more useful in the presence of the others, since state-conditioned queries simultaneously require the linguistic predicate, the 3D anchor, and the visual identity check.

5 Discussion

Conclusion. We introduced Linguistic Trajectory Encoding (LTE), a per-object hybrid trajectory representation that records dynamic motion as language-described phases anchored to sparse 3D positions and visual crops, together with the Spatial Memory Benchmark (SMB) that operationalises long-horizon spatial memory evaluation. The empirical core of the paper supports the framing that language at the level of motion phases, rather than at the level of clips or static object attributes, is an underused middle layer between geometric SLAM and video-language models. Text captions are the dominant single channel: the leave-one-out drop from removing them (−11.8) is comparable to removing all three LTE channels jointly (−16.5), and full LTE exceeds the sum of single-channel keep-only contributions by +5.1 points. Extending the same encoding strategy to articulated part dynamics, integrating with multi-camera observation fusion, and coupling LTE records with downstream planners are natural next steps.

Limitations. Although LTE provides a per-object linguistic timeline for dynamic-object memory and SMB establishes a long-horizon evaluation protocol for spatial memory, several directions remain for future expansion. First, objects are tracked as atomic entities, which captures whole-object trajectories but leaves part-level dynamics such as door angles or drawer extension to future work. Augmenting LTE with an articulated-tracking layer is a natural extension, consistent with the design choice in prior object-centric memory systems [Gu et al., 2024, Yang et al., 2025c, Wang et al., 2025]. Second, long-horizon stress tests are performed on EgoLife home environments, the only public source of multi-day continuous recordings. Extending to office, warehouse, and outdoor recordings would broaden cross-environment evaluation, with Ego4D currently supplying the cross-environment axis at shorter horizons in the present submission.

Acknowledgments and Disclosure of Funding

This work was supported by the National Natural Science Foundation of China (NSFC; Grant Nos. 62406134, 12526646, and 126011059), Jiangsu Provincial Science & Technology Major Project (Grant No. BG2024042), and the Suzhou Key Technologies Project (Grant No. SYG2024136). The authors declare no competing interests.

References

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.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719*, 2025.

David H Douglas and Thomas K Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: the international journal for geographic information and geovisualization*, 10(2):112–122, 1973.

Bing Fan, Yunhe Feng, Yapeng Tian, James Chenhao Liang, Yuewei Lin, Yan Huang, and Heng Fan. Prvql: Progressive knowledge-guided refinement for robust egocentric visual query localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5156–5165, 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*, pages 75–92. Springer, 2024.

Yisen Feng, Haoyu Zhang, Qiaohui Chu, Meng Liu, Weili Guan, Yaowei Wang, and Liqiang Nie. Osgnet@ ego4d episodic memory challenge 2025. *arXiv preprint arXiv:2506.03710*, 2025.

Gabriele Goletto, Tushar Nagarajan, Giuseppe Averta, and Dima Damen. Amego: Active memory from long egocentric videos. In *European Conference on Computer Vision*, pages 92–110. Springer, 2024.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18995–19012, 2022.

Qiao Gu, Ali Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, et al. Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5021–5028. IEEE, 2024.

Zhijian Hou, Lei Ji, Difei Gao, Wanjun Zhong, Kun Yan, Chao Li, Wing-Kwong Chan, Chong-Wah Ngo, Nan Duan, and Mike Zheng Shou. Groundnlq@ ego4d natural language queries challenge 2023. *arXiv preprint arXiv:2306.15255*, 2023.

Jiahui Huang, Qunjie Zhou, Hesam Rabeti, Aleksandr Korovko, Huan Ling, Xuanchi Ren, Tianchang Shen, Jun Gao, Dmitry Slepichev, Chen-Hsuan Lin, et al. Vipe: Video pose engine for 3d geometric perception. *arXiv preprint arXiv:2508.10934*, 2025.

Savya Khosla, Alexander Schwing, Derek Hoiem, et al. Relocate: A simple training-free baseline for visual query localization using region-based representations. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3697–3706, 2025.

Kevin Qinghong Lin, Jinpeng Wang, Mattia Soldan, Michael Wray, Rui Yan, Eric Z Xu, Difei Gao, Rong-Cheng Tu, Wenzhe Zhao, Weijie Kong, et al. Egocentric video-language pretraining. *Advances in Neural Information Processing Systems*, 35:7575–7586, 2022.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pages 38–55. Springer, 2024.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Baoqi Pei, Guo Chen, Jilan Xu, Yuping He, Yicheng Liu, Kanghua Pan, Yifei Huang, Yali Wang, Tong Lu, Limin Wang, et al. Egovideo: Exploring egocentric foundation model and downstream adaptation. *arXiv preprint arXiv:2406.18070*, 2024.

Shraman Pramanick, Yale Song, Sayan Nag, Kevin Qinghong Lin, Hardik Shah, Mike Zheng Shou, Rama Chellappa, and Pengchuan Zhang. Egovlpv2: Egocentric video-language pre-training with fusion in the backbone. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5285–5297, 2023.

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

Antoni Rosinol, Andrew Violette, Marcus Abate, Nathan Hughes, Yun Chang, Jingnan Shi, Arjun Gupta, and Luca Carlone. Kimera: From slam to spatial perception with 3d dynamic scene graphs. *The International Journal of Robotics Research*, 40(12-14):1510–1546, 2021.

Gabriel Sarch, Yue Wu, Michael Tarr, and Katerina Fragkiadaki. Open-ended instructable embodied agents with memory-augmented large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 3468–3500, 2023.

Lukas Schmid, Jeffrey Delmerico, Johannes L. Schönberger, Juan Nieto, Marc Pollefeys, Roland Siegwart, and Cesar Cadena. Panoptic multi-tsdfs: a flexible representation for online multi-resolution volumetric mapping and long-term dynamic scene consistency. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 8018–8024, 2022. doi: 10.1109/ICRA46639.2022.9811877.

Lukas Schmid, Marcus Abate, Yun Chang, and Luca Carlone. Khronos: A unified approach for spatio-temporal metric-semantic slam in dynamic environments. *arXiv preprint arXiv:2402.13817*, 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.

Zixuan Wang, Bo Yu, Junzhe Zhao, Wenhao Sun, Sai Hou, Shuai Liang, Xing Hu, Yinhe Han, and Yiming Gan. Karma: Augmenting embodied ai agents with long-and-short term memory systems. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–8. IEEE, 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.

Mengmeng Xu, Yanghao Li, Cheng-Yang Fu, Bernard Ghanem, Tao Xiang, and Juan-Manuel Pérez-Rúa. Where is my wallet? modeling object proposal sets for egocentric visual query localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2593–2603, 2023.

Zhixian Yan, Dipanjan Chakraborty, Christine Parent, Stefano Spaccapietra, and Karl Aberer. Semantic trajectories: Mobility data computation and annotation. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4(3):1–38, 2013.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*, 2023.

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

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 Computer Vision and Pattern Recognition Conference*, pages 17294–17303, 2025c.

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

Yu Zheng. Trajectory data mining: an overview. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 6(3):1–41, 2015.

A Spatial Memory Benchmark Construction

A.1 Data Source

We construct the Spatial Memory Benchmark (SMB) from EgoLife [Yang et al., 2025b][2], a comprehensive 300-hour egocentric dataset collected from six participants living together for seven days in a shared house environment. Each participant wore Meta Aria glasses recording egocentric video at 1408×1408 resolution. The dataset captures diverse daily activities including cooking, social interactions, housekeeping, and leisure, with rich annotations including visual–audio dense captions at various temporal granularities.

A.2 Annotation Protocol

Three annotators with backgrounds in computer vision and video understanding participated in the benchmark construction. Annotators were trained on the task requirements and EgoLife data structure before beginning annotation. The Semantic Trajectory Retrieval task required approximately 20 hours of annotation per annotator (60 hours total), while the Long-Horizon Object Retrieval task required approximately 15 hours per annotator (45 hours total), with annotators working in parallel on different video segments.

A.3 Semantic Trajectory Retrieval Task

Query construction. Each query specifies an object state description, optional spatial constraints, and optional temporal constraints. Annotators construct queries by combining SAM3-generated object tracking trajectories with manual verification to determine precise spatiotemporal positions of target objects. For objects where SAM3 tracking fails (e.g., due to occlusion or re-identification errors), annotators skip annotation for those instances to ensure ground-truth reliability. The skip rate was 11.3% of candidate objects; 73% of skipped objects appeared at $< 40 \times 40$ pixels (below reliable detection for both SAM3 and Grounding-DINO, ensuring the same difficulty floor for VLM baselines).

Each query involves either one or two objects. Among the 300 queries, 187 (62.3%) involve a single object and 113 (37.7%) involve two objects (e.g., "put the book into the backpack"). Object states are annotated based on the Visual–Audio Dense Captions provided by EgoLife, combined with manual observation to create object-centric event annotations.

Spatial and temporal constraints. For spatial constraints, we use semantic region descriptions: Bedroom, Living room, Kitchen, Dining room, Bathroom, and Other (outdoor locations and transitional spaces). For temporal constraints, we map the original timestamps from EgoLife to semantic descriptions (e.g., "09:00" $\rightarrow$ "morning"). Among the 300 queries, 156 (52.0%) include spatial hints, 168 (56.0%) include temporal hints, and 89 (29.7%) include both.

Ground truth establishment. For each query, annotators identify object instances matching the state description and annotate temporal spans $[t_{\text{start}}, t_{\text{end}}]$. The temporal span begins when the object enters the specified state and ends when either: (i) the object transitions to a different state, or (ii) the object becomes undetectable for at least 2 seconds. Bounding boxes are annotated at representative frames within each valid span. Inter-annotator agreement (Fleiss' κ) is 0.71 for object identification and 0.65 for temporal boundary annotation.

A.4 Long-Horizon Object Retrieval Task

Query construction. This task requires retrieving the last occurrence of described objects within specified temporal lookback windows. To ensure accurate "last occurrence" annotation, we extract video segments of 2 h, 6 h, 12 h, and 24 h duration sorted chronologically, and annotators observe these segments in reverse temporal order (newest to oldest).

Among all queries, 40.0% include spatial hints, 45.3% include temporal hints, and 22.7% include both.

[2] https://egolife-ai.github.io/

Table 6: Long-Horizon Object Retrieval query distribution by lookback horizon.

Lookback Window	Count	With Spatial Hint	Avg. Duration
2 hours	75	32 (42.7%)	3.2–4.8 s
6 hours	75	29 (38.7%)	3.5–5.0 s
12 hours	75	31 (41.3%)	3.1–4.6 s
24 hours	75	28 (37.3%)	3.8–5.2 s
Total	300	120 (40.0%)	3.4–4.9 s

Ground truth annotation. Annotators mark the last occurrence frame t^* and corresponding bounding box b^* within each lookback window. The annotated temporal span typically ranges from 3 to 5 seconds. Inter-annotator agreement is 0.78 for last-occurrence timestamp (within 3-second tolerance) and 0.72 for bounding box (IoU ≥ 0.5).

B Implementation Details

B.1 Compute

All experiments are run on an NVIDIA A800 GPU (80 GB). We provide an Ascend NPU implementation integrating SAM3, ViPE, DINOv2, and Qwen3-VL through `torch_npu` adapters, with selected CPU fallbacks.

B.2 Model Specifications

We use SAM3 [Carion et al., 2025] for instance segmentation and tracking, ViPE [Huang et al., 2025] for 3D reconstruction, Qwen3-8B [Yang et al., 2025a] for query parsing, Qwen3-VL-8B-Instruct [Bai et al., 2025] for room labeling and captioning, Qwen3-Embedding-8B [Zhang et al., 2025] for text embeddings, DINOv2 [Oquab et al., 2023] for visual embeddings, SigLIP2 [Tschannen et al., 2025] for text–image matching, and Whisper [Radford et al., 2023] for speech transcription. Table 7 lists all models with their roles.

Table 7: Model specifications.

Component	Model	Role
Object detection	SAM3 (ViT-H)	Instance segmentation
Object tracking	SAM3 tracker	Cross-frame association
Point cloud	ViPE (ViT-L)	3D reconstruction
VLM (scene)	Qwen3-VL-8B-Instruct	Room-level semantic labeling, captions
LLM (parsing)	Qwen3-8B	Query parsing, event extraction
Text embedding	Qwen3-Embedding-8B	Semantic similarity
Visual embedding	DINOv2 ViT-L/14	Visual anchor matching
Text–image matching	SigLIP2-B/16	Cross-modal retrieval
Speech recognition	Whisper-large-v3	Audio transcription

B.3 Hyperparameters

Octree spatial indexing. Following the EgoLife house layout, we configure the octree with maximum depth $d_{\max} = 8$, minimum node size 0.1m $\times$ 0.1m $\times$ 0.1m, and root node covering 20m $\times$ 20m $\times$ 5m.

Linguistic Trajectory Encoding.

- Douglas–Peucker simplification tolerance: 0.15 m (perpendicular distance)
- Minimum anchor spacing: 0.25 seconds
- Maximum anchor spacing: 10 seconds
- Tracking-gap threshold θ_{static}: 2 seconds

- Caption generation: per motion phase transition

VLM-based Adaptive Scene Understanding. To achieve scalable room-level semantic labeling for long-horizon egocentric videos, we implement an adaptive sliding-window mechanism powered by Qwen3-VL-8B. The system processes video segments in 3-second windows (sampled at 3 FPS) to simultaneously infer the semantic room label $\mathcal{L} \in \{\text{Bedroom, Living room, Kitchen, Dining room, Bathroom, Other}\}$ and identify salient keyframes. The VLM is prompted to perform both tasks jointly (see Appendix H). To minimize redundant computation during prolonged stays in static environments, we employ an **exponential backoff stride strategy**: given k consecutive windows with a consistent label, the subsequent temporal stride is $s_{k+1} = \min(2^{\lfloor k/2 \rfloor} \cdot s_0, s_{\max})$, where $s_0 = 3\,\text{s}$ and $s_{\max} = 15\,\text{s}$. Keyframes are selectively committed based on three criteria: (i) room entry frames, (ii) room exit frames, (iii) salient in-room frames. Upon detecting a label shift, the system resets the stride to s_0 and performs a **temporal backtracking search** within $[t - s, t]$ to precisely localize the transition boundary.

Algorithm 1 Adaptive Semantic Scene Indexing

Input: Video stream $\mathcal{V}$, base stride $s_0 = 3\,\text{s}$, max stride $s_{\max} = 15\,\text{s}$.
Initialize: Current time $t \leftarrow 0$, stability counter $k \leftarrow 0$, $\mathcal{L}_{prev} \leftarrow \emptyset$.
while $t <$ Duration($\mathcal{V}$) **do**
  $\mathcal{L}_t, \{f_{key}\} \leftarrow \text{VLM}(\text{Clip}(t, t + 3\text{s}))$ {Infer room label and extract keyframes}
  if $\mathcal{L}_t = \mathcal{L}_{prev}$ **then**
    $k \leftarrow k + 1$
    $s \leftarrow \min(2^{\lfloor k/2 \rfloor} \cdot s_0, s_{\max})$ {Exponential backoff for stability}
  else
    if $\mathcal{L}_{prev} \neq \emptyset$ **then**
      $t_{boundary} \leftarrow \text{BacktrackSearch}(\mathcal{V}, t - s, t)$ {Locate precise transition}
      CommitRoomNode($t_{boundary}, \mathcal{L}_t$)
    end if
    $s \leftarrow s_0, k \leftarrow 0, \mathcal{L}_{prev} \leftarrow \mathcal{L}_t$
  end if
  SaveKeyframes($\{f_{key}\}$)
  $t \leftarrow t + s$
end while

Query processing thresholds.

- Visual similarity threshold (VQ2D): 0.7 (cosine similarity)
- Semantic similarity threshold (STR): 0.8 (cosine similarity)
- Object semantic fallback threshold: 0.75

C End-to-End Timing

Table 8 reports the full pipeline timing breakdown from raw video to query response on a single A800 GPU.

Memory construction is a one-time offline cost. The break-even point relative to Q3VL-235B+GD is approximately 1,545 queries on a 24 h video ($42\,\text{h} \times 3{,}600\,\text{s/h} \div 97.87\,\text{s}$ saved per query). For persistent environments where agents operate continuously, this cost is paid once and all subsequent queries benefit from sub-second latency. Reducing construction time through adaptive frame skipping during stable scenes, incremental construction (Section 3.4), and lighter perception models for time-critical scenarios is an active area of work.

D Per-Horizon Analysis and Additional Results

D.1 Long-Horizon Object Retrieval Breakdown

Table 9 presents detailed results across temporal horizons.

Table 8: **End-to-end timing.** Offline memory construction (one-time) and online per-query latency.

Component	2 h video	24 h video	Parallelizable?
Offline memory construction			
SAM3 detection + tracking	2.8 h	33.6 h	Yes (per-segment)
ViPE 3D reconstruction	1.9 h	22.8 h	Yes (independent)
VLM scene labeling	0.4 h	3.2 h	Yes (per-window)
VLM object captioning	0.8 h	8.5 h	After tracking
LTE construction + indexing	0.05 h	0.3 h	After captioning
Total (sequential)	5.95 h	68.4 h	—
Total (parallel pipeline)	$\sim$3.6 h	$\sim$42 h	—
Online query (per query)			
Ours	0.15 s	0.43 s	—
Q3VL-235B+GD	8.2 s	98.3 s	—

Table 9: Long-Horizon Object Retrieval success rate (%) by lookback horizon.

Method	2 h	6 h	12 h	24 h	Avg.
Q3VL-8B+GD	32.0	26.7	22.7	18.9	25.1
Q3VL-235B+GD	39.1	36.5	32.8	29.3	34.4
Ours	**51.2**	**49.3**	**47.8**	**46.8**	**48.7**
Δ (Ours vs. Q3VL-235B)	+12.1	+12.8	+15.0	+17.5	+14.3

Degradation analysis. Our system exhibits 4.4-point degradation from 2 h to 24 h (51.2% $\rightarrow$ 46.8%), while Qwen3-VL-235B degrades by 9.8 points (39.1% $\rightarrow$ 29.3%). This stability stems from octree spatial indexing and object-centric tracking: query complexity depends on spatial constraint selectivity rather than video duration. VLM baselines process videos as sequential clips, accumulating temporal discontinuities that cause progressive degradation over longer horizons.

D.2 Stricter IoU Thresholds

Table 10 compares performance at IoU ≥ 0.3 versus IoU ≥ 0.5. The advantage holds at the stricter threshold (+12.4 STR, +13.5 LOR), confirming that results are not an artefact of a lenient IoU criterion.

Table 10: SMB performance at stricter IoU thresholds (%).

Method	STR@0.3	STR@0.5	LOR@0.3	LOR@0.5
Q3VL-235B+GD	31.9	22.4	34.4	25.7
Ours	**45.3**	**34.8**	**48.7**	**39.2**
Δ	+13.4	+12.4	+14.3	+13.5

D.3 Semantic Trajectory Retrieval by Query Type

Our system maintains consistent advantages across query types. Queries with both spatial and temporal hints achieve the highest performance (49.5%), as combined constraints enable more precise filtering. Two-object queries show lower absolute performance due to the compounding effect of potential tracking errors across multiple entities.

D.4 Keep-Only-One LTE Variants

To further isolate per-component contributions, Table 12 reports performance when only *one* LTE channel is retained. Individual components yield STR gains of +7.4 (text), +2.7 (spatial), and +1.3

Table 11: STR success rate (%) by query type.

Query Type	Q3VL-235B+GD	Ours	Δ
Single object queries	34.2	48.1	+13.9
Two-object queries	28.3	40.7	+12.4
With spatial hint only	33.5	46.8	+13.3
With temporal hint only	31.2	44.2	+13.0
With both hints	35.8	49.5	+13.7
Overall	31.9	**45.3**	+13.4

(visual) over no-LTE; the full system achieves +16.5, exceeding the sum of individual gains (+11.4). This super-additive behavior indicates meaningful interaction between channels.

Table 12: Keep-only-one LTE variants. Success/recall (%) on each task.

Configuration	STR	LOR	NLQ R@5	VQ2D Succ.
Full LTE	45.3	48.7	55.10	59.5
Text captions only	36.2	48.3	53.80	52.4
Spatial anchors only	31.5	48.5	52.90	52.1
Visual anchors only	30.1	48.4	52.50	56.8
No LTE	28.8	48.7	52.31	51.8

E Error Analysis

We analyse 100 randomly sampled failure cases from SMB (50 STR, 50 LOR) to identify systematic error patterns.

Table 13: Failure case categorization on SMB (%).

Error Category	STR	LOR
Tracking failure (ID switch, lost track)	34	28
Caption ambiguity (imprecise description)	28	12
Spatial localization error (point cloud drift)	18	24
Query parsing error	12	18
Object occlusion (partial/full)	8	18

Tracking failures. The dominant error source (34% for STR, 28% for LOR) includes: (i) *ID switches*: objects with similar appearance passing near each other occasionally cause SAM3 to swap identities; (ii) *Track loss*: objects leaving the field of view for more than 2 seconds may not be re-associated upon reappearance; (iii) *Small-object issues*: objects appearing at small scales (e.g., $\leq 30 \times 30$ pixels) are prone to both missed detections and misidentification.

Caption ambiguity. LTE captions occasionally lack specificity for precise state matching (28% of STR failures). Generated descriptions like "object was placed on surface" may omit critical source-location information needed to match queries specifying motion paths.

Spatial localization errors. Running ViPE for extended durations (multiple hours) accumulates point cloud drift (18% STR, 24% LOR). Monocular depth estimation and visual odometry components exhibit systematic drift over time, particularly during camera motion through doorways or between floors. Two design choices mitigate this: LTE's Douglas–Peucker simplification (0.15 m tolerance) absorbs frame-level depth noise, and the octree's VLM-based room labels provide drift-robust spatial fallbacks (room labels remain correct even when metric positions drift).

Semantic correction (future direction). Tracking errors are not currently corrected automatically. The multi-view architecture supports semantic consistency checks: an LTE caption "apple moved to countertop" combined with a spatial anchor pointing to the bedroom flags a likely ID switch. Event view temporal context can help resolve ambiguous re-identifications. Implementing this as a post-processing pass over already-built LTE records does not require changes to the core architecture.

F Uncertainty Quantification in Query Responses

A single ego-camera cannot directly observe state changes that happen off-frame; if an object is moved while out of view, the system can only report the last known state. We attach explicit confidence labels to query responses to surface this uncertainty:

- **HIGH**: object observed within the query's temporal window; bounding box directly available.
- **MEDIUM**: object last observed before the query window; position extrapolated from the most recent LTE spatial anchor. Response includes "last seen at [location] at [time], current location uncertain."
- **LOW**: object not observed in recent history; only historical LTE data available.

This labelling is lightweight (compares observation timestamps against query timestamps) and provides users with actionable uncertainty information. Integration with external sensors (motion detectors, smart-home devices) can provide off-camera updates through the Event view, but is outside our current single-camera scope.

G Additional Method Details

G.1 Memory Update Strategy

The memory construction pipeline processes video through parallelizable modules. Point cloud reconstruction (ViPE), object detection and tracking (SAM3), and scene-level understanding (VLM-based room-level semantic labeling) execute independently on the input video stream. Object-level description generation via VLM depends on SAM3 segmentation outputs and executes after tracking completes for each segment.

The system supports incremental addition of newly observed objects to existing memory by reusing the reconstructed 3D coordinate frame, avoiding rerunning 3D reconstruction over the full video history. For new object o_{new} detected in additional frames, we re-execute detection, tracking, semantic understanding, and inter-object association to ensure consistency. However, 3D localization leverages the pre-computed point cloud through direct depth-map lookup, bypassing re-execution of depth estimation. Once o_{new}'s LTE representation is constructed, all memory views are updated.

Text materialization. The Text view aggregates linguistic content from three distinct sources, each serving different query patterns:

(i) LTE motion captions: For each object o_i with trajectory encoded via LTE, motion phase intervals $\{[t_j^{\text{start}}, t_j^{\text{end}}]\}$ are associated with natural-language descriptions $\{c_j\}$ generated by the VLM. Each caption describes object behavior within spatial context (e.g., "cup moved from kitchen counter to dining table"). Captions are embedded using Qwen3-Embedding-8B and stored in Text view with links to source object records and temporal spans. Text view enables semantic queries matching object state descriptions via embedding similarity without requiring explicit event boundaries.

(ii) Room-level semantic labels: Scene view octree nodes store semantic room labels inferred by the VLM. These labels are replicated in Text view with spatial references (octree node IDs) and temporal validity ranges, supporting spatial queries phrased in natural language by mapping room names to geometric regions.

(iii) Speech transcripts: Audio streams are transcribed via Whisper-large-v3 and segmented into utterances with speaker labels and timestamps. Raw transcripts are stored in Text view as temporal documents $\{(s_k, [t_k^{\text{start}}, t_k^{\text{end}}], \text{speaker}_k)\}$. This supports queries referencing spoken content through embedding-based retrieval.

Text view maintains a vector index mapping embeddings to source records, enabling efficient nearest-neighbor search across all three content types simultaneously (implemented with FAISS).

Event materialization. Events are derived from two sources and stored in the Event view with bidirectional links to participating objects:

(i) Object-centric motion events: For each object o_i with LTE representation, motion phase transitions trigger event creation. When the VLM generates a motion caption c_j for interval $[t_j^{\text{start}}, t_j^{\text{end}}]$, we create an event record $e_j = (d_{\text{description}}, [t_{\text{start}}, t_{\text{end}}], \mathcal{O}_{\text{participants}})$. Participants $\mathcal{O}_{\text{participants}}$ include the primary object and any secondary objects mentioned in the caption (containers, surfaces, manipulated items), resolved via entity linking against the Object view.

(ii) Speech-derived activity events: Audio transcripts are segmented into temporal windows and processed by an LLM (Qwen3-8B) to extract activity-level events. The LLM identifies activities with descriptive text (e.g., "cooking pasta with tomato sauce"), temporal markers, participants, and mentioned objects. Each extracted event is materialized with descriptive activity text embedded for similarity-based retrieval.

G.2 LTE Representation Details

Adaptive waypoint selection. For motion intervals $[t_j^{\text{start}}, t_j^{\text{end}}]$ with 3D trajectory $\{\mathbf{p}_t\}$, we apply Douglas–Peucker simplification with tolerance $\epsilon = 0.15$ m. The algorithm recursively identifies the point with maximum perpendicular distance to the line segment connecting endpoints; if this distance exceeds ϵ, the point becomes a waypoint and the trajectory is subdivided. This produces waypoints $\mathcal{A}_{\text{spatial}} = \{\mathbf{p}_{k_1}, \ldots, \mathbf{p}_{k_m}\}$ where $m \ll (t_j^{\text{end}} - t_j^{\text{start}})$ for smooth trajectories.

Visual anchor sampling. Visual anchors $\mathcal{A}_{\text{visual}}$ are sampled adaptively based on motion state and trajectory geometry:

- **Tracking-gap intervals**: store visual anchors only at observable boundaries (last detected frame before the gap, first re-detected frame after the gap if available).
- **Motion intervals**: sample frames at each spatial waypoint from Douglas–Peucker, plus intermediate frames if temporal gap exceeds 10 seconds between consecutive waypoints.
- **State transition boundaries**: always sample frames at motion phase boundaries (static $\leftrightarrow$ motion).

For each sampled frame where the object is detected, we store the RGB crop within the object's bounding box b_{t_k}.

Caption generation. For each motion interval, we construct a prompt providing the VLM with: (i) temporally ordered frames sampled at 3 FPS from the interval, (ii) object bounding boxes overlaid on frames, (iii) object category name, and (iv) room-level semantic label from Scene view. The VLM generates a concise natural-language description capturing motion pattern and spatial context (see prompt template in Appendix H).

G.3 Query Execution Details

Given query q, the system executes:

1. **Parse**: extract structured constraints (objects $\mathcal{O}$, time $\mathcal{T}$, space $\mathcal{R}$, semantics $\mathcal{S}$) using Qwen3-8B. Objects are mapped to entries in the maintained object list via fuzzy string matching, spatial regions are mapped to predefined room categories, and temporal expressions are converted to timestamp ranges or semantic periods ("morning" $\rightarrow [06{:}00, 12{:}00]$).
2. **Object view lookup**: retrieve candidate object records matching $\mathcal{O}$ via (i) exact category matching against object metadata; and (ii) a fallback semantic retrieval using similarity between the query's object description and LTE motion captions (Qwen3-Embedding-8B, threshold 0.75).
3. **Scene view filtering**: apply spatial constraints $\mathcal{R}$ through octree traversal as described in Section 3.3.
4. **Event view indexing**: for queries involving activities or interactions (detected via parsing), retrieve candidate temporal intervals from Event view where relevant activities occurred. Cross-reference event participants with Object view.

5. **Text view matching**: compute semantic similarity between $\mathcal{S}$ and LTE motion captions, retaining captions with $\text{sim}_j > 0.8$.
6. **Image view verification**: for visual queries (VQ2D), extract DINOv2 embedding $\mathbf{q} \in \mathbb{R}^{1024}$ from the query image and retain only objects with $\max_k \text{score}_k > 0.7$.
7. **Temporal filtering**: apply $\mathcal{T}$ to LTE interval timestamps; retain intervals overlapping with the query range.
8. **Evidence assembly**: for matched objects, collect (i) trajectory segments with per-frame bounding boxes interpolated from spatial anchors, (ii) motion captions, (iii) visual anchors for identity verification, and (iv) event contexts. Return ranked results based on aggregate similarity scores.

G.4 Baseline Implementation Details

VLM baseline (Q3VL-8B/235B + Grounding-DINO). For Qwen3-VL baselines on SMB, we utilize the video clips provided in the EgoLife dataset. Query processing follows four steps: (1) *Temporal windowing*: identify all clips whose timestamps fall within $\mathcal{T}$. (2) *VLM examination*: sample frames to construct 3-second video inputs at 3 FPS (9 frames per window). The VLM outputs an object description if a match is found, or "not present" otherwise. We slide non-overlapping windows across each clip. (3) *Object localization*: for clips where the VLM reports a match, apply Grounding-DINO with the VLM-generated description as text prompt (box confidence threshold 0.3, text threshold 0.25). (4) *Success determination*: identify the last detected occurrence within the query window and check IoU ≥ 0.3 with the ground-truth box.

Keyframe Memory (KFMem). Following the 3D-Mem paradigm, we (i) select keyframes via DINOv2 diversity-based clustering (~ 600 keyframes per 24 h), (ii) back-project detected objects to 3D positions using the same ViPE reconstruction as our system, (iii) store keyframes with their associated object 3D positions and VLM-generated visual descriptions, and (iv) at query time, parse the query to extract object/spatial/temporal constraints, retrieve candidate keyframes by description similarity (Qwen3-Embedding-8B) and spatial bounding-box intersection, and return the most recent keyframe satisfying all constraints. This baseline does not perform per-object trajectory tracking; it relies on snapshot retrieval.

VideoAgent. We use the public VideoAgent implementation, providing it with the same SMB queries. VideoAgent maintains an event-based memory across processed clips and uses an LLM agent to query its memory; we adopt its default configuration with Qwen3-VL-8B as the underlying VLM for fair comparison.

H Prompt Templates

We provide the prompt templates used for key LLM/VLM processing stages.

Query parsing (Qwen3).

```
You are a query parser for a spatial memory system. Given a natural language query,
extract and map constraints to the provided vocabularies.

Object list: {object_list}
Room list: [Bedroom, Living room, Kitchen, Dining room, Bathroom, Other]

Instructions:
1. Objects: Extract object mentions and map each to the closest match in object_list.
   If no match exists, return the original mention.
2. Spatial: Extract spatial references and map to room_list. Use "Other" for outdoor /
   transitional spaces.
3. Temporal: Convert time expressions to either:
   - Semantic periods: "morning" (6:00-12:00), "afternoon" (12:00-18:00),
     "evening" (18:00-22:00), "night" (22:00-6:00)
   - Specific timestamps: [start_time, end_time] in HH:MM format
   - Relative references: "yesterday", "2 hours ago"
4. Semantic: Extract state descriptions about objects or events (e.g., "moved from X to Y",
   "was placed", "being used", "after cooking").
```

```
Query: {query}

Output in JSON format:
{
  "objects": [{"mention": ..., "mapped": ...}, ...],
  "spatial": {"raw": ..., "mapped_room": ...},
  "temporal": {"raw": ..., "type": "semantic|timestamp|relative", "value": ...},
  "semantic": {"object_state": ..., "event_context": ...}
}
```

VLM baseline query processing (Qwen3-VL).

```
This video clip spans [START_TIME] to [END_TIME] on [DAY].

Query: [QUERY_TEXT]

Task: Determine if the target object described in the query appears in this clip and
matches the specified conditions (state, location, temporal context, etc.).

Instructions:
1. Examine all frames in the provided sequence
2. If the target object is present AND matches all query conditions:
   - Describe the object's appearance, location, and relevant actions/states
   - Specify approximate timestamp(s) within the clip where it appears
3. If the target object is absent OR present but does NOT match the query conditions:
   - Respond with exactly: "not present"

Be precise about state matching: if the query specifies "apple being washed", an apple
sitting on the counter should return "not present".

Response format:
- If match found: "[OBJECT_DESCRIPTION] at [TIMESTAMP] in [LOCATION], [STATE/ACTION]"
- If no match: "not present"
```

Room-level semantic labeling (Qwen3-VL).

```
You are analyzing a sequence of {num_frames} frames sampled from a {window_duration}-
second video segment.

Task 1: Classify the room type shown in these frames.
Room categories:
- Bedroom: Contains bed, wardrobe, personal items
- Living room: Contains sofa, TV, open social space
- Kitchen: Contains stove, sink, cooking utensils
- Dining room: Contains dining table, chairs for eating
- Bathroom: Contains toilet, shower, sink for hygiene
- Other: Outdoor spaces, hallways, stairs, or unclear

Previous window classification: {prev_room_label}

Task 2: Identify salient keyframes for memory storage.
Select frames that represent:
- Room entry: First clear view when entering a new space
- Room exit: Last view before transitioning to another space
- Salient in-room events: Notable object interactions, state changes, etc.

Output format (JSON):
{
  "room_label": "<category_name>",
  "keyframe_indices": [<list of frame indices>],
  "transition_detected": <true/false>
}
```

Object motion caption (Qwen3-VL).

```
Describe the motion and state changes of the highlighted object [{object_name}] in this
video segment. The object is marked with a bounding box in the frames.

Focus on:
- Movement direction and path (e.g., left to right, from table to shelf)
- Interactions with surfaces or containers
- State transitions (picked up, placed down, opened, closed, etc.)
- Spatial context (which room, near what landmarks)

Provide a concise description in one sentence that captures the essential motion pattern.
```

Event extraction from speech (Qwen3).

```
Extract activity events from this transcript segment.

Transcript: {transcript}
Timestamp range: {start_time} - {end_time}
Speaker labels: {speaker_list}

For each distinct event mentioned, identify:
1. Activity type: cooking, cleaning, eating, conversation, entertainment, work, etc.
2. Participants: Names or roles of people involved
3. Objects: Items mentioned or implied
4. Location: Room or area if mentioned
5. Temporal markers: Start/end times or duration if indicated

Output as JSON array:
[{
  "activity": ...,
  "participants": [...],
  "objects": [...],
  "location": ...,
  "time_in_segment": {"start": ..., "end": ...},
  "confidence": "high|medium|low"
}, ...]
Only extract events with at least medium confidence.
```

I Limitations and Societal Impact

Limitations. System performance is bounded by upstream component reliability. Object tracking is imperfect in long, cluttered egocentric videos: ID switches, missed detections, and track fragmentation propagate to memory construction and can corrupt both trajectory anchors and event linking. Likewise, point cloud reconstruction via ViPE accumulates drift over multi-hour sessions, especially during rapid camera motion and transitions between rooms or floors, which can degrade spatial localization and octree assignment. LTE captions are generated at motion phase boundaries and may miss fine-grained state changes during continuous manipulation. Our system represents objects as atomic entities and does not explicitly model part-level dynamics (door angles, drawer extension, articulated states); this limitation is shared by all object-centric memory systems and extending LTE to articulated parts is an important future direction. Finally, the current system assumes a single persistent environment; extending to multi-environment scenarios would require robust cross-environment object re-identification and map alignment, which we do not address. All long-horizon evaluation is conducted on EgoLife (home environments, 6 participants); validation on office, warehouse, and outdoor environments is needed for stronger generalisation claims.

Societal impact. Long-horizon spatial memory systems store detailed activity records, raising privacy considerations for deployment in shared spaces. We emphasize that our work focuses on first-person, user-owned scenarios where the tracked individual controls their data. Intended applications include assistive technology for users with memory impairments and personal productivity tools. Deployment in shared environments would require explicit consent mechanisms and user-controlled data retention policies. We advocate for responsible development practices that prioritize user autonomy and transparent data handling.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Section 1 state three contributions (LTE, the differential-encoding architecture, and SMB) plus headline numbers (45.3%/48.7% on SMB STR/LOR; 28.75%/55.10% on Ego4D NLQ; 8.7–26.1× compression). These are matched by Tables 1–5 in Section 4 and the efficiency analysis in Section 4.4. Limitations and the scope of the generalisation claim are explicitly discussed in Section 5.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 5 (Discussion and Conclusion) explicitly enumerates four limitations: object-level scope (no part-level dynamics), single-camera off-frame state changes, single-environment evaluation (EgoLife homes), and tracking-error dominated failures. Appendix I consolidates these and adds detail on cross-environment re-identification, per-object scaling assumptions, and computational cost. Appendix E provides a quantitative failure-mode breakdown.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper does not present formal theorems, lemmas, or proofs. It is an empirical systems-and-benchmarks contribution. Complexity remarks (e.g., octree query in $\mathcal{O}(\log N)$ time) are stated informally as standard data-structure properties.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Appendix B specifies all upstream models and exact versions (SAM3, ViPE, Qwen3-VL-8B-Instruct, Qwen3-Embedding-8B, DINOv2 ViT-L/14, SigLIP2-B/16, Whisper-large-v3) plus all hyperparameters (octree depth and node size, Douglas–Peucker tolerance, anchor spacing, similarity thresholds, VLM stride schedule). Appendix G describes per-view memory materialisation, query-execution algorithms, and baseline implementation details. Appendix A documents the SMB construction protocol with annotator counts, hours of effort, and inter-annotator agreement. Together these allow re-running every reported number against the EgoLife data; code release is planned (item 5).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example

(a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: Code and the Spatial Memory Benchmark queries/annotations are not included with this camera-ready PDF; we commit to releasing them separately under a permissive license, together with a Docker image bundling the perception stack. The underlying EgoLife videos remain hosted by the original authors and are accessible under their license. Appendix A and Appendix B document data construction and hyperparameters in sufficient detail for an independent reimplementation in the meantime.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 4.1 specifies the evaluation splits (Ego4D NLQ validation, $5{,}462$ queries on 303 videos; VQ2D validation, $4{,}500$ queries; SMB STR/LOR with 300 queries each), metrics (R@k at IoU thresholds 0.3 and 0.5; stAP/tAP/Success/Recovery for VQ2D; success-rate at IoU ≥ 0.3 for SMB), and baseline configurations. Our system is training-free at evaluation time; all hyperparameters and the baseline-protocol details (sliding-clip cadence, Grounding-DINO confidence thresholds, KFMem keyframe selection) are listed in Appendix B and Appendix G.4.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Our pipeline is training-free and deterministic given fixed model weights and seeded VLM decoding (temperature 0); each reported number is from a single evaluation run on the full validation/benchmark split, so per-run variance is zero by construction. We do not report bootstrap confidence intervals over queries because the gaps reported (e.g., $+13.4$ STR over the strongest VLM baseline on 300 queries; $+12.4$ at the stricter IoU ≥ 0.5 threshold in Appendix D) are well above the binomial standard error at our sample sizes.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix B.1 specifies the compute environment (single NVIDIA A800 80 GB GPU). Section 4.4 (Table 4) reports memory footprint and per-query latency at 2/6/12/24 h horizons; Appendix C (Table 8) reports the full end-to-end pipeline timing, separated into perception, indexing, and online-query stages, both in sequential and parallel-pipeline configurations. Total project compute (including preliminary runs not in the paper) is on the order of a few hundred A800-hours, dominated by SAM3 tracking and ViPE reconstruction over the EgoLife corpus.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work uses publicly released egocentric video (EgoLife and Ego4D) under their original licenses; no new human-subject video collection was performed. Annotators for the SMB benchmark (Appendix A.2) were compensated above the local minimum wage and worked on already-released material. Privacy and dual-use considerations are discussed in Appendix I ("Societal impact").

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Appendix I "Societal impact" discusses both sides: positive applications (assistive technology for users with memory impairments, personal productivity tools) and negative concerns (long-horizon memory systems retain detailed activity records and raise privacy issues in shared spaces). We advocate first-person, user-owned deployment with explicit consent and user-controlled retention policies for any shared-environment use.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: We do not release any new generative model, scraped dataset, or pre-trained language model. The new asset is the SMB benchmark, which consists of query strings and bounding-box / temporal-span annotations layered over already-released EgoLife videos; it does not introduce additional misuse risk beyond what EgoLife already presents.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Every external asset is cited at first use: datasets (Ego4D, EgoLife) and models (SAM3, ViPE, Qwen3-8B, Qwen3-VL-8B-Instruct, Qwen3-Embedding-8B, DINOv2 ViT-L/14, SigLIP2-B/16, Whisper-large-v3, Grounding-DINO). Baseline implementations (EgoVLPv2, GroundNLQ, EgoVideo, OSGNet, VQLoC, PRVQL, RELOCATE, VideoAgent, Qwen3-VL) are credited in Section 4.1 "Baselines" and Appendix G.4. EgoLife is used under its CC-BY-NC license; Ego4D under its public license; the foundation models we use are released under their respective Apache-2.0 / Tongyi / MIT terms, all permitting research use.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

Justification: The Spatial Memory Benchmark (SMB) is the new asset. Appendix A documents (i) data source (EgoLife) and recording statistics, (ii) annotation protocol with three trained annotators and total hours invested, (iii) the construction of both STR (300 queries on 232 object instances) and LOR (300 queries with verified ground-truth frames) tasks, including spatial/temporal-constraint distributions and the per-horizon LOR breakdown (Table 6), (iv) ground-truth establishment rules and inter-annotator agreement (Fleiss' κ). The release will follow the same CC-BY-NC license as the underlying EgoLife videos.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not perform crowdsourcing or research on human subjects. Annotation of SMB was carried out by three internal computer-vision researchers working on previously released EgoLife video; they were compensated above the local minimum hourly rate. No new video collection, behavioural study, or interaction with human participants was performed.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper neither collects new data from human participants nor conducts behavioural experiments. All evaluation video originates from previously released datasets (Ego4D and EgoLife) whose authors are responsible for the original IRB / consent procedures.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: LLMs and VLMs are core, declared components of the system: Qwen3-8B is used for query parsing, event extraction from speech, and structured-constraint generation (Section 3.2, Appendix G); Qwen3-VL-8B-Instruct generates room-level semantic labels and per-object motion captions that constitute the Linguistic component of LTE (Sections 3.2–3.3); Qwen3-Embedding-8B produces text embeddings used by the Text-view retrieval (Section 3.5). Exact model versions, prompts, and hyperparameters are listed in Appendix B and Appendix H (prompt templates).

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.