RETHINKING RAG IN LONG VIDEOS: WHAT TO RETRIEVE AND HOW TO USE IT?

Anonymous authors
Paper under double-blind review

ABSTRACT

Retrieval-augmented generation is extending beyond text to long videos, where query-relevant chunks can be represented across multiple modalities and temporal granularities. Progress in this setting, VideoRAG, is limited by two gaps: existing benchmarks allow queries to be answered without the video, obscuring retrieval errors, and prior methods apply a single modality-granularity configuration per query, ignoring chunk-level variability. We address both by introducing `V-RAGBench`, a benchmark of ⟨query, evidence chunk, answer⟩ triplets over hour-scale videos, in which each answer depends on a unique evidence chunk, enabling decoupled evaluation of retrieval and generation, and `CARVE`, a training-free method that runs parallel retrievers across configurations and uses chunk-adaptive reranking to select a winning configuration for each chunk, which is then carried into generation. On `V-RAGBench`, `CARVE` outperforms eight recent VideoRAG baselines on both stages, with the evidence it supplies to the generator interleaving multiple configurations rather than sharing one. Its gains hold across egocentric and third-person long videos and extend to an expanded configuration space.

1 INTRODUCTION

Retrieval-augmented generation (RAG) grounds large language models (LLMs) in external knowledge, improving factual reliability (Choi et al., 2025; Singh et al., 2025; Zhao et al., 2026a). Its frontier is now extending from text corpora to *long-form video* (Jeong et al., 2025; Zhang et al., 2025b; Xue et al., 2025; Yeo et al., 2026b), such as first-person recordings from wearable devices (Bock et al., 2024; Yang et al., 2025) and video histories that agentic systems operate over (Chen et al., 2025; Rege et al., 2026). We refer to RAG over long videos as `VideoRAG`. Unlike a text passage, a video chunk has no single canonical form: it can be matched through different *modalities* (Zhang et al., 2024a; Maaz et al., 2024; Kahatapitiya et al., 2025), from raw visual features to textual abstractions, and at different *temporal granularities* (Lin et al., 2024c; Hu et al., 2025), from individual frames to whole clips. The problem thus moves beyond what to retrieve to "which modality and granularity to retrieve each chunk from, and how to use them across retrieval and generation?"

Despite this growing interest (Lin et al., 2024b; Ma et al., 2025; Sun et al., 2026), current benchmark datasets have not kept pace. While hour-scale source videos are available through Ego4D (Grauman et al., 2022) and EgoLife (Yang et al., 2025), the queries paired with them are scarce and were not designed for `VideoRAG`. In particular, recent work (Lim et al., 2026) shows that over half of widely used video QA samples can be answered without the video at all, relying on linguistic priors, world knowledge, or static cues. Such queries undermine `VideoRAG` evaluation, since a system can reach correct answers even when retrieval surfaces irrelevant chunks, so *high generation accuracy does not imply successful retrieval*, and final accuracy cannot serve as a proxy for the retrieval stage. Consequently, how modality and temporal granularity should be used across retrieval and generation cannot be identified, as each design choice is invisible behind the final accuracy.

The same gap appears on the methodological side. Recent work proposes techniques for `VideoRAG`, such as guided query refinement (Wang et al., 2025e; Xue et al., 2025), modality fusion (Luo et al., 2025; Yang et al., 2026), and query decomposition (Xu et al., 2025), while others place retrieval inside iterative agentic loops (Ma et al., 2025; Yeo et al., 2026b). Although retrieval is the component on which all these frameworks rest, its validation is reduced to downstream QA accuracy of the full pipeline. This contrasts with text-centric RAG, where retrieval is studied as a first-class object and audited through stage-wise metrics on query-evidence pairs (Formal et al., 2021; Choi et al., 2025).

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

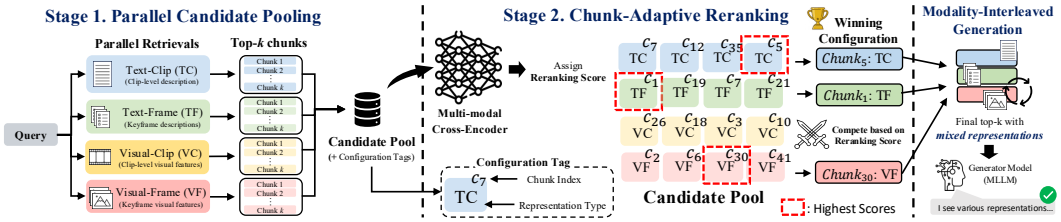

Figure 1: The overview of CARVE: Stage 1 builds a candidate pool via chunk-wise parallel retrieval, while Stage 2 performs chunk-adaptive reranking. The final top-k evidence with their winning configuration is passed to the generator in a modality-interleaved form.

We address this gap by introducing V-RAGBench, a benchmark designed to make retrieval and generation in VideoRAG measurable both in isolation and in their interaction. V-RAGBench consists of 2,100 ⟨query, evidence chunk, answer⟩ triplets over 216 hour-scale videos (1–9 hours) drawn from Ego4D (Grauman et al., 2022) and EgoLife (Yang et al., 2025). It jointly enforces three properties that no existing dataset satisfies together: each query is (i) *tied to non-recurring evidence*, so that the underlying event does not repeat elsewhere in the video; (ii) *visually-grounded*, so that the answer is not implied by the query itself or recoverable from parametric knowledge alone; and (iii) *evidence-localized*, so that the answer cannot be reconstructed from cues that recur across non-evidence chunks. These properties, verified by exhaustive human audits, make generation causally dependent on retrieval, enabling a stage-wise view of VideoRAG.

With V-RAGBench, we shift the question of VideoRAG from improving QA accuracy to which representation of a chunk best supports each stage, retrieval and generation. Each chunk admits four configurations[1] from two axes: modality (visual vs. textual-summary embeddings) and temporal granularity (frame vs. clip). Existing strategies (Hsu & Tzeng, 2025; Jeong et al., 2025; Han et al., 2026; Ren et al., 2026) apply one configuration, or a fixed fusion, uniformly to every chunk of a query—a *query-level* decision. However, this is suboptimal as the best configuration depends on the chunk's content, not only on the query; a visually salient moment is best surfaced by visual embeddings, a semantically rich one by its textual summary, and likewise for granularity. The question is thus at which level this decision should be made. We answer it at the chunk level with CARVE (Chunk-Aware Reranking for Video Evidence; Figure 1), which re-purposes a multi-modal cross-encoder, a standard reranking tool in text-RAG (Li et al., 2024; Abdallah et al., 2025; Wang et al., 2025c), to assign each chunk its winning configuration and feed it to the generator under that configuration. Being training-free and adding only parallel retrieval and reranking to the pipeline, CARVE lets us isolate the effect of the decision level and serves as a drop-in retrieval module for existing pipelines, including agentic ones, as shown in Appendix D.9.

Concretely, CARVE operates in two stages. Parallel retrieval runs one retriever per configuration, each contributing its top-k chunks to a shared candidate pool. Chunk-adaptive reranking then re-scores each candidate with a multi-modal cross-encoder under its retrieving configuration (*e.g.*, query–text relevance for textual retrievals, query–visual for visual ones), yielding a final ranking that interleaves chunks from different configurations, each tagged with its winning one. These top-k chunks serve both stages. Retrieval outputs them with their tags, and generation receives each chunk under its winning configuration, so the representation fed to the generator varies per chunk. Importantly, our experiments reveal that this chunk-level selection improves both retrieval and generation, surpassing existing VideoRAG methods and even trained query-level routers.

Our experiments on V-RAGBench reveal four key findings: (1) CARVE significantly outperforms eight baseline methods on both retrieval and generation, a comparison enabled by V-RAGBench's stage-wise measurability; (2) a modality–granularity ablation reveals large performance gaps across configurations at both stages, with no single configuration uniformly optimal, motivating chunk-level rather than query-level decisions; (3) CARVE's chunk-level decisions distribute fairly evenly across the four configurations, showing that per-chunk diversity is genuine rather than collapsing onto a single dominant choice; and (4) at generation, CARVE even surpasses trained query-level routers that learns to pick one configuration per query, without any additional training.

[1]Other retrieval signals, such as video knowledge graphs (Shen et al., 2025; Dixit & Liang, 2026) or metadata (*e.g.*, explicit temporal clue) (Ren et al., 2024; Guo et al., 2025), can aid retrieval but are not consistently available. We focus on content-based representations, which are universally accessible.

2 Related Work

Retrieval-Augmented Generation in Text. RAG retrieves query-relevant evidence from an external text corpus and conditions the generator on it (Lewis et al., 2020; Jiang et al., 2023; Arslan et al., 2024; Choi et al., 2025). Since retrieval quality largely dictates answer quality, progress has centered on three fronts. Retrievers have advanced from sparse and dense models (Karpukhin et al., 2020; Formal et al., 2021; Yan et al., 2024) to hybrid pipelines that combine both (Mandikal & Mooney, 2024; Sawarkar et al., 2024; Hsu & Tzeng, 2025). Queries are reformulated through rewriting (Ma et al., 2023; Wang et al., 2025b; Zhao et al., 2026b) or expansion (Azad & Deepak, 2019; Zhang et al., 2024b; Choi et al., 2025) to make retrieval intent explicit. Reranking re-scores first-stage candidates with a cross-encoder to improve precision (Li et al., 2024; Yu et al., 2024; Abdallah et al., 2025; Wang et al., 2025c). These techniques are developed for text passages, which have a single canonical form, and do not address which representation of a video chunk to retrieve from.

Retrieval-Augmented Generation in Long Video. Long-form video understanding has traditionally relied on query-independent representations of the whole video, such as sparse frame sampling (Lin et al., 2024a; Chasmai et al., 2025; Yao et al., 2025), visual-token compression (Zhang et al., 2023; Cheng et al., 2024; Huang et al., 2025; Wang et al., 2025a), or full-video textualization (Liu & Wan, 2021; Zhang et al., 2024a; Lee et al., 2025; Li et al., 2023), which can dilute query-relevant evidence as length grows. `VideoRAG` instead retrieves query-relevant chunks before generation. A video chunk, however, is inherently multimodal (Maaz et al., 2024; Zhang et al., 2024a; Kahatapitiya et al., 2025), expressible through visual frames or derived textual abstractions, and readable at temporal granularities from individual frames to segments (Lin et al., 2024c; Hu et al., 2025).

Existing work explores these axes separately. On modality, Jeong et al. (2025) retrieves chunks with multimodal embeddings and Luo et al. (2025) augments retrieval with visually-aligned auxiliary text such as OCR and ASR. On granularity, hierarchical and tree-structured representations index the video at multiple temporal scales (Wang et al., 2025d; Ren et al., 2026). A further line replaces one-shot retrieval with iterative agentic probing that repeatedly fetches frames or segments (Wang et al., 2024b; Chen et al., 2025; Xue et al., 2025; Yeo et al., 2026b). Across these methods, the modality–granularity choice, or a fixed fusion of the options, is made once per query, and evaluation rests mainly on downstream QA accuracy. Our work treats the choice as a chunk-level decision and introduces a benchmark that measures retrieval and generation separately.

Video Benchmark Datasets for QA. Video QA benchmarks have progressed from short third-person clips (Xu et al., 2017; Lei et al., 2018; Yu et al., 2019) toward longer and increasingly egocentric videos (Grauman et al., 2022; Bärmann & Waibel, 2022; Mangalam et al., 2023; Patel et al., 2025; Yang et al., 2025), a natural target of `VideoRAG` as wearable devices and personal agents accumulate hour-scale first-person video (Zhou et al., 2025; Rege et al., 2026). Yet existing benchmarks remain limited in video length, *e.g.*, EgoSchema (3-minute clips) (Mangalam et al., 2023), QaEgo4D (short Ego4D intervals) (Bärmann & Waibel, 2022), Video-MME ($\leq$60 min) (Fu et al., 2025), and LongVideoBench ($\leq$1 hour) (Wu et al., 2024). Ego4D (Grauman et al., 2022) and EgoLife (Yang et al., 2025) provide hour-scale source video, but their queries are scarce and serve other goals. Ego4D's Episodic Memory benchmark evaluates retrieval alone, and EgoLifeQA targets memory aggregation across days. Neither guarantees the three properties that `VideoRAG` evaluation requires, namely (i) non-recurring evidence, (ii) visual grounding, and (iii) evidence localization, which tie correct answers to the labeled evidence rather than to spurious cues. CG-Bench (Chen et al., 2024a) and MM-Lifelong (Chen et al., 2026) check shortcut answerability and clue grounding, addressing (ii) but not (i) and (iii). `V-RAGBench` jointly enforces all three on hour-scale video.

3 Formulating VideoRAG along Two Design Axes

We formalize `VideoRAG` as a retrieve-then-generate process over long videos. Prior studies (Ren et al., 2026; Yeo et al., 2026b) establish this pipeline but leave two design choices underspecified. First, a chunk admits multiple representations across modality (visual vs. textual) and granularity (frame vs. clip), yet these are combined at fixed scales rather than treated as decision variables. Second, the representation used for retrieval and the one used for generation are tied to a single choice by default, and their relationship has received little attention.

Following standard text RAG practice (Arslan et al., 2024; Singh et al., 2025), a `VideoRAG` system partitions a long video V into non-overlapping, fixed-interval temporal chunks $V = \{v_1, \ldots, v_{|V|}\}$

that serve as retrieval units. We default to 2-minute chunks, matching the empirical 2–3 minute span of bounded semantic content in long-form video (Islam et al., 2024). Unlike a text passage, each chunk v admits multiple representations $\phi_{m,g}(v)$ along two axes, a modality $m \in \{\text{vis}, \text{text}\}$ (visual features vs. textual abstractions) and a temporal granularity $g \in \{\text{frame}, \text{clip}\}$ (frame-level details vs. clip-level summaries).[2] Given a query q, a retriever $\mathcal{R}$ matches q against chunk representations to select k candidate chunks $V_q \subseteq V$, and an MLLM-based generator $\mathcal{G}$ produces an answer from those chunks under a chosen representation:

$$\widehat{a} = \mathcal{G}\big(q,\ \{\phi_{m_g,g_g}(v)\}_{v\in V_q}\big) \ \text{ where } \ V_q = \mathcal{R}(q,\ \{\phi_{m_r,g_r}(v)\}_{v\in V}), \tag{1}$$

(m_r, g_r) and (m_g, g_g) are the configurations chosen for retrieval and generation; and $\phi_{m,g}(v)$ is the representation of chunk v under modality m and granularity g. The goal is twofold: V_q should contain the chunk that supports the answer, and $\widehat{a}$ should match the reference answer a. Notably, whereas a text chunk has a single canonical form, a video chunk forces `VideoRAG` to decide *which* (m, g) *to use for retrieval and which for generation*, and these decisions need not coincide.

We now describe how the two design axes are instantiated in this work. Full implementation details (*e.g.*, decoding hyperparameters, temperature) are deferred to Appendix B, and the prompt templates used are provided in Appendix E.

Chunk Modality. The modality axis $m \in \{\text{vis}, \text{text}\}$ exposes a chunk v through two complementary channels. The *visual* modality encodes v as raw visual features from an MLLM-based encoder (Qwen3-VL-Embedding-2B (Li et al., 2026) by default), preserving fine-grained appearance and motion cues that resist verbalization. In contrast, the *textual* modality prompts an MLLM (Qwen3-VL-8B) to summarize v in natural language and embeds the summary with the same encoder, surfacing semantic structure (*e.g.*, what is happening, where, and with whom) at the cost of pixel-level detail. Neither is universally preferable, as which is more useful depends on the query and chunk content.

Chunk Granularity. Even within a fixed modality, the granularity axis $g \in \{\text{frame}, \text{clip}\}$ exposes a chunk v at two temporal scales. The *frame*-level retains per-frame detail through a small set of keyframes, obtained by clustering all frame embeddings in the chunk with k-Means++ (Arthur & Vassilvitskii, 2007) ($n = 5$ by default) and taking the centroids, which cover distinct visual states rather than uniform time samples. The *clip*-level treats v as a single unit. Each is realized in a modality-specific way. Under the visual modality, the frame-level mean-pools the keyframe embeddings, while the clip-level encodes the entire chunk into one embedding. Under the textual modality, the same MLLM produces a summary capped at 512 tokens, applied per keyframe (concatenated before embedding) for the frame-level, while over the entire chunk for the clip-level. The cap accommodates the detailed descriptions used in prior multimodal summarization work (Chen et al., 2024b; Islam et al., 2024) while keeping verbose outputs from dominating the context. Neither granularity is universally preferable, as frame-level forms capture transient details but lose the surrounding flow, while clip-level forms retain temporal context but blur such details.

Scope and Objective. With (m, g) now made explicit, two questions follow: *how* it should be decided and *at what level*. In other words, for each chunk v, which representation $\phi_{m,g}(v)$ should be passed to the retriever and the generator? The common choice fixes one (m, g) per query and applies it uniformly across chunks (Jeong et al., 2025; Ren et al., 2026), and agentic variants (Yeo et al., 2026b) issue multiple retrievals per query but commit each call to a single (m, g). The most useful (m, g), however, may differ across chunks even within one query. For instance, identifying an object visible in only one keyframe is best handled by a frame-level visual embedding, while describing how an activity unfolds over the chunk is better expressed by a clip-level textual summary. Given a query q, we therefore decide (m, g) *per chunk* and carry the choice consistently into both retrieval and generation, keeping the standard retrieve-then-generate pipeline intact.

4 Construction of V-RAGBench

Diagnosing how the two design axes interact at retrieval and generation requires a benchmark in which the two stages are measurable both in isolation and their interaction, which in turn requires every query to be tied to a uniquely sufficient evidence chunk. We thus construct `V-RAGBench`, a

[2] We instantiate $\mathcal{M}$ and $\mathcal{G}$ with two values each, giving four (m, g) configurations for ease of exposition. The formulation and `CARVE` accommodate additional modalities and granularities without any change, and we extend the space to six configurations with ASR and OCR in Appendix D.8.

benchmark of ⟨query, evidence chunk, answer⟩ triplets built from hour-scale videos via four stages (see Figure 2): source selection, event extraction and deduplication, query generation, and filtering.

Source Video Selection. We draw source videos from egocentric corpora. Unlike third-person long videos (*e.g.*, movies, lectures) organized around a single narrative arc, first-person streams capture continuous, weakly structured daily activity in which the same actor revisits similar scenes and routines, so retrieval must separate the queried moment from many near-duplicate distractors, which makes it a demanding testbed for `VideoRAG`.[3]. We use Ego4D (Grauman et al., 2022) and EgoLife (Yang et al., 2025), retaining only videos *longer than one hour* and applying stratified sampling over length bins so that the corpus spans a wide range of durations. This yields 216 videos in total, comprising 42 EgoLife recordings (avg. 379.1 min., 177.8–552.7 min.) and 174 Ego4D videos (avg. 86.0 min., 60.0–423.6 min.); see Table 7. Retrieval is performed within each source video, so every query is grounded in a single long video. Domain and length statistics are in Appendix A.1.

Event Extraction and Deduplication. This stage converts each long video into a small set of distinct event chunks that form a non-redundant evidence pool for QA generation. For each video, we encode every sampled frame with EVA02-E-14-plus (Sun et al., 2023), a CLIP-based vision encoder, and apply kernel temporal segmentation (Potapov et al., 2014) on the frame embeddings to cut the video into semantically coherent, time-contiguous segments. The number of segments is capped at 150 for the shorter Ego4D videos and 900 for the day-long EgoLife recordings, with each segment constrained to 5–300s. We then re-embed each segment with Qwen3-VL-Embedding-8B (Li et al., 2026) in video mode, cluster the embeddings with k-Means++ ($k=50$ for Ego4D, $k=300$ for EgoLife), and retain only the segment closest to each centroid. This suppresses recurring routines (*e.g.*, repeated meal preparation) and leaves chunks that correspond to *distinct events*, which is a prerequisite for QA generation, since queries grounded in recurring events would admit multiple sufficient evidence chunks. Such queries are natural in the real world but are excluded here for rigorous `VideoRAG` evaluation. Details and statistics of this stage are in Appendix A.4.

Query Generation. For each event chunk, we prompt Gemini-3-flash-preview to generate up to four candidate queries, each assigned to one of three predefined categories (Appendix A.3); queries are generated only for categories the chunk visually supports. The prompt enforces *contextual localization*, requiring every query to embed anchor information from its source event so that the query alone describes the situation enough to be matched against the right chunk. This lets `V-RAGBench` dispense with direct temporal cues such as timestamps or when-phrases, so retrieval must rely purely on content (prompt and examples in Appendix E). Each generation yields a triple $\langle q, [t_s, t_e], a\rangle$ of query, evidence time span, and an *open-ended* answer rather than a multiple-choice option, so that downstream evaluation reflects the generator's ability rather than random guessing over a fixed choice set. This produces 67,370 candidate ⟨query, evidence chunk, answer⟩ triples for filtering.

Post-hoc Filtering. Event deduplication enforces uniqueness at the event level, but the generated queries must still be verified at the query level for *visual grounding*, so that the answer is not implied by the query or by priors, and *evidence localization*, so that the answer is not recoverable from non-evidence chunks. We thus introduce five filters, ordered from cheapest to most expensive (see Table 12), using models other than the query generator to limit shared biases except where video input length forces the same family. (1) *Semantic Similarity* removes queries too close in Qwen3-VL-embedding-8B, with per-video caps of 50 (Ego4D) and 300 (EgoLife) so that no single video dominates; (2) *Shortcut Bias* discards queries that GPT-5.2-chat answers correctly without visual context; (3) *Answerability* removes queries that GPT-5.2-chat judges unanswerable from the source clip; (4) *Empirical Answerability* requires Gemini-2.5-flash to answer correctly from the source clip, judged by an LLM-as-a-judge; and (5) *Evidence Uniqueness* retrieves the top-10 chunks per query, *excluding* the true evidence, and discards the query if Gemini-2.5-flash still answers correctly.

The surviving queries are thus (1) distinct, (2) free of parametric and commonsense shortcuts, (3, 4) grounded in sufficient evidence, and (5) not recoverable from any other chunk of the same video. Lastly, since the three query categories are unevenly populated, we subsample each to the smallest one ($n=700$), yielding 2,100 queries (1,800 train, 300 test).

Human Verification. To validate the output of the filtering pipeline, Table 1 reports human verification of answerability, evidence uniqueness, and query non-redundancy, where each item is labeled

[3]The benchmark is egocentric by construction, but the method is not. We also evaluate `CARVE` on third-person long videos (LongVideoBench) and on lecture videos with ASR/OCR (Video-MMMU) in Appendices D.3–D.8

Table 1: Human verification of V-RAGBench, each item labeled independently by two annotators.

Requirement	Property Verified	Items (Judgments)	Result (Inter-annotator agreement)
Answerability	Answerable from labeled evidence	577 sampled QA pairs (1,154)	96.9% answerable (κ 0.63)
Evidence Uniqueness	Non-recurring, localized evidence	10,206 test window–query pairs (20,412)	97.1% unanswerable (AC1 0.96)
Query Redundancy	Query-level non-redundancy	2,648 same-video test pairs (5,296)	0 duplicates (AC1 1.00)

independently by two annotators. For answer accuracy to reflect retrieval, the labeled chunk must suffice to answer the query and no other chunk may, so that a correct answer implies successful retrieval. Annotators confirm both. On 577 QA pairs sampled to represent the full benchmark at a 95% confidence level, 96.9% are judged answerable from the labeled chunk (IAA 0.63). Over every two-minute window of test videos, judged without any retriever, 97.1% of windows outside the labeled evidence are unanswerable (95% CI 96.8–97.5, IAA 0.96). Annotators also find no duplicate among the 2,648 same-video test query pairs (IAA 1.00). See Appendix A.6 for details.

5 CARVE: Chunk-Aware Reranking for Video Evidence

CARVE is a training-free instantiation of chunk-level configuration decision. It adds only parallel retrieval and a reranking step to the standard retrieve-then-generate pipeline, so its gain over query-level baselines is attributable to deciding (m, g) per chunk and carrying that decision into generation. It operates in two stages, parallel candidate pooling and chunk-adaptive reranking, as in Figure 1.

Stage 1: Parallel Candidate Pooling. Given a query q, this stage builds a *candidate pool* by performing four parallel retrievals, one per configuration (m, g) with $m \in \{\text{vis}, \text{text}\}$ and $g \in \{\text{frame}, \text{clip}\}$. The four embeddings $\phi_{m,g}(v)$ of each chunk v in Eq. (1) are precomputed offline, so each retrieval only encodes the query q and performs a nearest-neighbor lookup, incurring no chunk-encoding cost at inference regardless of how many configurations are queried. Adding configurations grows the candidate pool sublinearly, as analyzed in Appendix D.8.

For each configuration (m, g), we compute the dot-product similarity $\langle \cdot, \cdot \rangle$ between the query embedding q and each chunk embedding $\phi_{m,g}(v)$, and keep the k chunks with the highest scores as the top-k retrieval list $C_{(m,g)}(q)$ for that configuration:

$$C_{(m,g)}(q) \;=\; \text{top-}k \text{ chunks } v \text{ ranked by } \langle q,\, \phi_{m,g}(v) \rangle. \tag{2}$$

The four lists are then merged into a configuration-tagged candidate pool $P(q)$, where each retrieved chunk is paired with the configuration (m, g) that retrieved the chunk v:

$$P(q) \;=\; \bigcup_{(m,g)} \Big\{ \big(\underbrace{v}_{\text{chunk}},\, \underbrace{(m,g)}_{\text{tag}} \big) \;:\; v \in C_{(m,g)}(q) \Big\}. \tag{3}$$

The same chunk v may appear under multiple tags if it was retrieved by more than one configuration. These tags are kept, since they are the per-configuration signal from which Stage 2 chooses a winning (m^*, g^*) for each chunk, unlike late fusion or reciprocal-rank fusion, which merge them into one score and lose which configuration surfaced the chunk.

Stage 2: Chunk-Adaptive Reranking. Given the candidate pool $\mathcal{P}(q)$ from Stage 1, this stage selects the final top-k chunks, while assigning each surviving chunk a single *winning* configuration (m^*, g^*). Since the chunk embeddings in $\mathcal{P}(q)$ live in different modalities depending on the tag—textual when $m = \text{text}$, visual when $m = \text{vis}$—we use a multi-modal cross-encoder $\text{CE}(\cdot, \cdot)$[4] that jointly attends to a textual query and either textual or visual chunk modality. We use Qwen3-VL-Reranker-2B as the default reranker, though performance remains consistent across rerankers, as in Appendix D.6.

Importantly, each chunk v in $\mathcal{P}(q)$ is re-scored *only* under the tag (m, g) that retrieved it, not under all four configurations. The rerank score of a tagged candidate $(v, (m, g)) \in \mathcal{P}(q)$ is computed:

$$\tilde{s}\big(q;\, (v, (m, g))\big) \;=\; \text{CE}\big(q,\, \phi_{m,g}(v)\big). \tag{4}$$

For each chunk v, we keep only its highest-scoring tag as its winning configuration:

$$(m_v^*,\, g_v^*) \;=\; \underset{(m,g)\,:\,(v,(m,g)) \in \mathcal{P}(q)}{\arg\max} \; \tilde{s}\big(q;\, (v, (m, g))\big). \tag{5}$$

[4]Temporal granularity g does not change the modality of $\phi_{m,g}(v)$ since frame- and clip-level forms under the same modality share the same encoder branch and embedding space, differing only in temporal scope (Section 3).

CE scores are comparable across modalities, so chunks can be ranked together by their winning scores, yielding the final top-k list (V_q^*) with each chunk tagged by its winning configuration:

$$V_q^* \;=\; \Big\{ \big(\, v,\, (m_v^*, g_v^*)\,\big) \;:\; v \in \text{top-}k \text{ chunks ranked by } \tilde{s}\big(q;\, (v, (m_v^*, g_v^*))\big) \Big\}. \tag{6}$$

This chunk-adaptive reranking improves retrieval by judging each chunk under its most discriminative configuration and letting chunks from different configurations compete on a comparable scale, so the top-k naturally interleaves the strongest evidence across modalities and granularities.

Modality-Interleaved Generation. V_q^* benefits not only retrieval but also generation. Unlike prior work that fixes a single (m_g, g_g) for the whole query (Jeong et al., 2025; Ren et al., 2026) or fuses several representations uniformly, we expose each chunk to the generator $\mathcal{G}$ *only* through its winning representation $\phi_{m_v^*, g_v^*}(v)$:

$$\widehat{a} \;=\; \mathcal{G}\big(q,\, \{\phi_{m_v^*,\, g_v^*}(v) \;:\; (v, (m_v^*, g_v^*)) \in V_q^*\}\big)\,. \tag{7}$$

The generator context thus mixes representations chunk by chunk, with each chunk rendered under its winning configuration. Empirically, this propagation from retrieval to generation—not the retrieval gain alone—is where CARVE's strongest improvements appear.

The simplicity of CARVE is deliberate. Being training-free with a standard retrieval interface, it replaces the retrieval module of existing VideoRAG pipelines without modification, including agentic frameworks, and improves both stages there, as shown in Appendix D.9.

6 Evaluation on V-RAGBench

Our evaluation of CARVE on the test set of V-RAGBench consists of three parts. First, we compare CARVE against recent VideoRAG methods on both retrieval and generation to demonstrate its superiority (Table 2). Second, we study how modality and temporal granularity should be selected under our method, showing that CARVE's chunk-level choices outperform query-level alternatives (Table 3). Third, we further show the efficiency and effectiveness of chunk-level decisions via latency, selection distribution, and comparison against two trained query-level routers (Tables 4–6).

Compared Methods. We compare CARVE against eight RAG methods applicable to VideoRAG, grouped into two categories. (i) *System-level* methods apply a fixed policy regardless of the query, including VideoRAG-A (Jeong et al., 2025), a late fusion method that linearly interpolates text and visual scores; GQR (Uzan et al., 2026), which refines query embeddings at test time via complementary retriever signals; and the multimodal encoders FreeRet (Zhu et al., 2026) and GME (Zhang et al., 2025c), which retrieve in a single embedding space. (ii) *Query-level* methods adapt the policy per query but apply the same decision to all chunks, including RRF (Rackauckas, 2024), which expands a query into variants and merges their ranks by reciprocal rank fusion; DAT (Hsu & Tzeng, 2025), a hybrid fusion method that weights dense and sparse retrievers via LLM-assessed reference scores; and VideoRAG-B (Luo et al., 2025) and UniversalRAG(-LoRA) (Yeo et al., 2026a), which respectively decompose or route a query across auxiliary databases of different modalities and granularities, where (-LoRA) is fine-tuned on the V-RAGBench training set. These baselines comprehensively cover late fusion of per-modality scores (VideoRAG-A), reciprocal-rank fusion of ranked lists (RRF), and hybrid fusion with adaptive retriever weights (DAT). For fairness, all baselines share the same chunk embeddings as CARVE and differ only in how representations are selected or combined. See Appendix B.2 for implementation details.

Evaluation Metrics. We evaluate retrieval and generation separately. For retrieval, we report two metrics: *nDCG@k*, which rewards methods that rank the annotated evidence higher; and *Recall@k*, which measures whether the annotated evidence appears among the top-k retrieved chunks. Since retrieval relies on deterministic embedding similarity, these metrics carry no run-to-run variance.

For generation, we report the aggregate *pass rate*, defined as the fraction of queries whose generated answer is judged correct against the reference by an LLM-as-a-judge (Qwen3.6-35B-A3B), which is necessary since answers are in open-ended form rather than multiple choice (see the prompt in Figure 8; 99.0% of its decisions are upheld by human annotators in Appendix B.4). To assess generalization across generators, we evaluate each retrieval method with three backbones: {Qwen3-VL-8B, Qwen3-VL-32B, and Gemma-4-26B}. We fix $k = 5$ throughout experiments, except in Appendix D.5 where we vary k to show that CARVE consistently outperforms baselines. Detailed definitions of all metrics are provided in Appendix B.3.

Table 2: Retrieval and generation performance on V-RAGBench. Generation is reported as pass rate under three generator backbones. Retrieval is deterministic, so significance is tested on pass rate only, averaged over three runs (refer to Appendix D.1 for details). *: $p < 0.05$ for CARVE against the second-best method (Welch's t-test). The best values are marked in bold.

		Retrieval Performance		Generation Performance (w. Three Backbones)		
Category	Method	Recall@5	nDCG@5	Qwen3-VL-8B	Qwen3-VL-32B	Gemma-4-26B
System-Level	VideoRAG-A (Jeong et al., 2025)	0.510	0.332	0.238	0.312	0.305
	GQR (Uzan et al., 2026)	0.503	0.340	0.167	0.110	0.153
	FreeRet (Zhu et al., 2026)	0.263	0.187	0.227	0.200	0.160
	GME (Zhang et al., 2025c)	0.413	0.253	0.253	0.282	0.212
Query-Level	RRF (Rackauckas, 2024)	0.463	0.298	0.180	0.130	0.187
	DAT (Hsu & Tzeng, 2025)	0.460	0.312	0.223	0.147	0.200
	VideoRAG-B (Luo et al., 2025)	0.487	0.325	0.312	0.303	0.236
	UniversalRAG (Yeo et al., 2026a)	0.447	0.298	0.259	0.253	0.228
	UniversalRAG-LoRA (Yeo et al., 2026a)	0.470	0.311	0.237	0.270	0.220
Chunk-Level	**CARVE (Ours)**	**0.603**	**0.433**	**0.352***	**0.368***	**0.318***

6.1 Main Result: Performance on Retrieval and Generation

Table 2 compares CARVE with four system-level and four query-level baselines on V-RAGBench. On retrieval, CARVE achieves the best Recall@5 (0.603) and nDCG@5 (0.433), exceeding the strongest baselines, VideoRAG-A at 0.510 Recall@5 and GQR at 0.340 nDCG@5. This retrieval gain translates into stronger generation on all three backbones. CARVE reaches a pass rate of 0.318–0.368, a 4.3–17.9% relative improvement over the second-best method with every margin statistically significant. The chunk-level decision strengthens both stages: chunk-adaptive reranking surfaces the evidence chunk under its winning configuration, which is then carried forward as interleaved evidence for generation, improving answers on every backbone. The same trend holds beyond egocentric video. On the third-person LongVideoBench and on Video-MMMU lecture videos with ASR and OCR added, CARVE again ranks first on all three backbones, as reported in Appendices D.3–D.8. It also stays first on an extended 500-query set in Appendix D.1. Remaining failures are mostly cases where retrieval succeeds but the generator misorders the actions in time, which we detail in Appendix D.4.

We next ask whether the four configurations synergize or a single one already captures most of the gain, by varying which configurations are available in both stages of CARVE. In Table 3, the top block uses one configuration, the middle block expands along one axis, and the bottom row uses all four. The best single configuration is visual clip-level at 0.507 Recall@5, the best partial combination is textual frame-and-clip at 0.567, and the full setting reaches 0.603 Recall@5, 0.433 nDCG@5, and 0.352 pass rate. Combining is not always beneficial, though. Visual frame-and-clip drops to 0.497 Recall@5, below visual clip-level alone, whereas adding a modality improves over both constituents in both cases. This suggests that cross-modal expansion is the more reliable source of synergy than expanding granularity within a single modality.

Table 3: Ablation over modality–granularity configurations in CARVE.

Configuration	R@5	nDCG@5	Pass Rate	Latency
m={text}, g={frame}	0.430	0.301	0.207	2.5s
m={text}, g={clip}	0.433	0.293	0.157	1.5s
m={vis}, g={frame}	0.477	0.336	0.323	5.2s
m={vis}, g={clip}	0.507	0.369	0.293	7.3s
m={text, vis}, g={frame}	0.513	0.352	0.303	5.0s
m={text, vis}, g={clip}	0.543	0.398	0.300	5.5s
m={text}, g={frame, clip}	0.567	0.413	0.217	3.3s
m={vis}, g={frame, clip}	0.497	0.355	0.327	6.1s
m={text, vis}, g={frame, clip}	0.603	0.433	0.352	4.6s

Importantly, this synergy comes at a reasonable latency cost. The full setting runs at 4.6s per query, which is modestly above text-only configurations (1.5-3.3s) but actually faster than visual-only ones (5.2-7.3s). This is because each chunk is rendered through its winning configuration, so visual evidence is passed only for chunks that benefit from it, not every retrieved chunk.

6.2 In-depth Analysis of Interleaved Chunk Representations

We further analyze how CARVE's interleaved chunk representations work in practice from three angles: the distribution of winning configurations across ranks, the reranking strategy itself, and a comparison against trained query-level routers at the generation stage.

Distribution of Winning Configurations. Table 4 shows the distribution of winning configurations across the top-5 ranked chunks. The distribution does not collapse onto a single dominant configuration. At every rank, all four are selected with non-trivial frequency, with {vis, clip} and {text, frame} winning most often and {text, clip} and {vis, frame} contributing consistently. This spread is not an artifact of score-scale mismatch, as the cross-encoder's scores largely overlap across the four configurations, as detailed in Appendix D.7. Different chunks are therefore genuinely best surfaced through different modality–granularity configurations, supporting our design choice of treating both axes as chunk-level decisions.

Table 4: Distribution of winning configuration over chunk-aware reranking.

Configuration	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5
{text, frame}	0.37	0.27	0.25	0.27	0.25
{text, clip}	0.15	0.13	0.11	0.11	0.13
{vis, frame}	0.24	0.26	0.28	0.27	0.28
{vis, clip}	0.24	0.34	0.36	0.35	0.34

Reranking Configuration Strategy. We next examine whether CARVE's reranking rule of rescoring each candidate under its retrieving configuration in Eqs. 4–5 is the right strategy. Table 5 compares against three alternatives: (i) applying a single fixed configuration to all candidates regardless of how they were retrieved, (ii) assigning a configuration randomly to each candidate (Random), and (iii) feeding all four configurations jointly into the reranker without selection as reranked multimodal late fusion (Concatenation). CARVE substantially outperforms all alternatives. This shows that scoring each chunk under its retrieving configuration tag, rather than under a fixed or heuristic alternative, is what drives the gain. In particular, Concatenation fails to match CARVE despite using all four configurations jointly, indicating that the chunk-level alignment between retrieval and reranking matters more than access to more representations.

Table 5: Reranking configuration choice under CARVE.

Method	R@5	nDCG@5	Latency
m={text}, g={frame}	0.497	0.343	0.8s
m={text}, g={clip}	0.490	0.339	0.5s
m={vis}, g={frame}	0.547	0.391	4.1s
m={vis}, g={clip}	0.560	0.396	8.0s
Random	0.513	0.312	3.1s
Concatenation	0.513	0.339	8.2s
CARVE (Ours)	0.603	0.433	3.4s

CARVE also achieves a favorable accuracy–latency trade-off. Reranking visual features is costly due to the larger token count (4.1s and 8.0s for visual-only configurations), but by passing only each chunk's winning configuration, CARVE runs at 3.4s while still achieving the best retrieval, substantially faster than visual-only and Concatenation (8.2s). The same trend for retrieval performance and latency holds under a different reranker backbone (LamRA-Rank-7B (Liu et al., 2025), see Appendix D.6).

Comparison against Query-Level Routing. We also examine whether CARVE's chunk-level interleaving of representations is more effective than rendering all retrieved chunks under a single, query-level representation. Table 6 compares CARVE with (i) fixed-configuration pipelines, which commit to one configuration for every query, and (ii) trained query-level routers, a non-LLM router (MLP) and an LLM router, which choose among all four configurations per query and apply the choice uniformly to all chunks (Appendix B.2). Given access to every configuration and training on the V-RAGBench training split, the routers represent the strongest query-level policy we can construct. Yet CARVE outperforms every alternative without training. As the routers receive the same retrieved chunks as CARVE, this margin reflects how evidence is rendered, suggesting that no single configuration suits every chunk. CARVE sidesteps this by varying the representation chunk by chunk through the same reranking step, at competitive latency.

Table 6: Comparison of CARVE against query-level routing in generation.

Method	Pass Rate	Latency
m={text}, g={frame}	0.207	2.5s
m={text}, g={clip}	0.157	1.5s
m={vis}, g={frame}	0.323	5.2s
m={vis}, g={clip}	0.293	7.3s
Non-LLM Router w. Training	0.329	5.7s
LLM Router w. Training	0.310	5.3s
CARVE (Ours)	**0.352**	4.6s

7 Conclusion

We introduced V-RAGBench, which makes retrieval and generation in VideoRAG separately measurable, and CARVE, which decides modality and granularity per chunk rather than per query. Our results suggest several insights. First, no single configuration suits every chunk, so a chunk-level decision outperforms query-level routing, including trained routers. Second, the decision is most beneficial when carried into generation, where each chunk is rendered under its winning configuration. Third, cross-modal expansion is a more reliable source of synergy than finer granularity within one modality. These findings also hold on third-person videos and a larger configuration space, indicating chunk-level representation selection as a promising direction for VideoRAG.

AI USE STATEMENT

In this work, we used generative AI tools for grammar checking and proofreading of author-written text. We have reviewed all AI-assisted edits. Large language and multimodal models are also used as components of the benchmark construction and evaluation pipeline, namely for QA generation, filtering, and LLM-as-a-judge evaluation of `V-RAGBench`, as described in Section 4 and Appendices A.5, B.4, and E. All references were manually curated and confirmed by the authors.

ETHICS STATEMENT

We do not foresee specific ethical risks arising directly from the proposed method or benchmark beyond those generally associated with multimodal retrieval and video understanding research. Nevertheless, we acknowledge that egocentric video inherently captures personal, daily-life activities, and any downstream deployment of `VideoRAG` systems should consider privacy, consent, and responsible use of personal video data.

For the human evaluation components described in Appendix A.6 and Appendix B.4, we recruited graduate-student annotators to verify the reliability of our filtering pipeline and the LLM-as-a-judge evaluation. All annotators were compensated above the U.S. federal minimum wage, were informed of the task scope and purpose in advance, and participated voluntarily. No personally identifiable information about the annotators is disclosed in the dataset, paper, or supplementary materials, and their contributions are used solely for the validation analyses reported in this work.

REPRODUCIBILITY STATEMENT

We take several steps to ensure the reproducibility of both `V-RAGBench` and `CARVE`. The benchmark construction pipeline is described in Section 4, and source video selection, event extraction and deduplication, query generation, and the five post-hoc filters are detailed in Appendices A.1 to A.5, including dataset statistics (Table 7), domain and category definitions (Tables 8 and 9), and filtering statistics (Table 12). All prompts used for QA generation, filtering, memory construction, answering, and judging are provided verbatim in Appendix E, and the human verification protocols for the benchmark and the LLM-as-a-judge are reported in Appendices A.6 and B.4. `CARVE` is formalized in Section 5 and Algorithm 1, and its implementation, together with those of all baselines and the trained routers, is specified in Appendix B, including model checkpoints (Tables 14 and 15), hyperparameters, decoding settings, and compute resources. Evaluation metrics are defined in Appendix B.3, and run-to-run variance and significance tests are reported in Table 2 and Appendix D.1. The benchmark, source code, and evaluation scripts will be publicly released upon acceptance.

REFERENCES

Abdelrahman Abdallah, Mahmoud Abdalla, Bhawna Piryani, Jamshid Mozafari, Mohammed Ali, and Adam Jatowt. Rankarena: A unified platform for evaluating retrieval, reranking and rag with human and llm feedback. In *CIKM*, 2025.

Muhammad Arslan, Hussam Ghanem, Saba Munawar, and Christophe Cruz. A survey on rag with llms. *Procedia Computer Science*, 246:3781–3790, 2024.

David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. In *SODA*, 2007.

Hiteshwar Kumar Azad and Akshay Deepak. Query expansion techniques for information retrieval: a survey. *Information Processing & Management*, 56(5):1698–1735, 2019.

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.

Marius Bock, Hilde Kuehne, Kristof Van Laerhoven, and Michael Moeller. Wear: An outdoor sports dataset for wearable and egocentric activity recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 8(4), 2024. doi: 10.1145/3699776.

Leonard Bärmann and Alex Waibel. Where did i leave my keys? — episodic-memory-based question answering on egocentric videos. In *CVPRW*, 2022.

Mustafa Chasmai, Gauri Jagatap, Gouthaman KV, Grant Van Horn, Subhransu Maji, and Andrea Fanelli. Moment sampling in video llms for long-form video qa. *arXiv preprint arXiv:2507.00033*, 2025.

Boyu Chen, Zhengrong Yue, Siran Chen, Zikang Wang, Yang Liu, Peng Li, and Yali Wang. Lvagent: Long video understanding by multi-round dynamical collaboration of mllm agents. In *ICCV*, 2025.

Guo Chen, Yicheng Liu, Yifei Huang, Yuping He, Baoqi Pei, Jilan Xu, Yali Wang, Tong Lu, and Limin Wang. Cg-bench: Clue-grounded question answering benchmark for long video understanding. *arXiv preprint arXiv:2412.12075*, 2024a.

Guo Chen, Lidong Lu, Yicheng Liu, Liangrui Dong, Lidong Zou, Jixin Lv, Zhenquan Li, Xinyi Mao, Baoqi Pei, Shihao Wang, Zhiqi Li, Karan Sapra, Fuxiao Liu, Yin-Dong Zheng, Yifei Huang, Limin Wang, Zhiding Yu, Andrew Tao, Guilin Liu, and Tong Lu. Towards multimodal lifelong understanding: A dataset and agentic baseline. *arXiv preprint arXiv:2603.05484*, 2026.

Lin Chen, Xilin Wei, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Bin Lin, Zhenyu Tang, Li Yuan, Yu Qiao, Dahua Lin, Feng Zhao, and Jiaqi Wang. Sharegpt4video: Improving video understanding and generation with better captions. *arXiv preprint arXiv:2406.04325*, 2024b.

Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, et al. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. *arXiv preprint arXiv:2406.07476*, 2024.

Jeonghwan Choi, Minjeong Ban, Minseok Kim, and Hwanjun Song. Word2passage: Word-level importance re-weighting for query expansion. In *Findings of ACL*, 2025.

Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960.

Cheng Cui, Yubo Zhang, Ting Sun, Xueqing Wang, Hongen Liu, Manhui Lin, Yue Zhang, Tingquan Gao, Changda Zhou, Jiaxuan Liu, Zelun Zhang, Jing Zhang, Jun Zhang, and Yi Liu. Pp-ocrv5: A specialized 5m-parameter model rivaling billion-parameter vision-language models on ocr tasks. *arXiv preprint arXiv:2603.24373*, 2026.

Aradhya Dixit and Tianxi Liang. Semantic event graphs for long-form video question answering. *arXiv preprint arXiv:2601.06097*, 2026.

Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. Splade: Sparse lexical and expansion model for first stage ranking. In *SIGIR*, 2021.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *CVPR*, 2025.

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 *CVPR*, 2022.

Yongxin Guo, Jingyu Liu, Mingda Li, Dingxin Cheng, Xiaoying Tang, Dianbo Sui, Qingbin Liu, Xi Chen, and Kevin Zhao. Vtg-llm: Integrating timestamp knowledge into video llms for enhanced video temporal grounding. In *AAAI*, 2025.

Kilem L. Gwet. Computing inter-rater reliability and its variance in the presence of high agreement. *British Journal of Mathematical and Statistical Psychology*, 61(1):29–48, 2008. doi: 10.1348/000711006X126600.

Gyuwon Han, Young Kyun Jang, and Chanho Eom. Cova: Text-guided composed video retrieval for audio-visual content. *arXiv preprint arXiv:2601.22508*, 2026.

Hsin-Ling Hsu and Jengnan Tzeng. Dat: Dynamic alpha tuning for hybrid retrieval in retrieval-augmented generation. *arXiv preprint arXiv:2503.23013*, 2025.

Kai Hu, Feng Gao, Xiaohan Nie, Peng Zhou, Son Tran, Tal Neiman, Lingyun Wang, Mubarak Shah, Raffay Hamid, Bing Yin, and Trishul Chilimbi. M-llm based video frame selection for efficient video understanding. In *CVPR*, 2025.

Kairui Hu, Penghao Wu, Fanyi Pu, Wang Xiao, Xiang Yue, Bo Li, Yuanhan Zhang, and Ziwei Liu. Video-mmmu: Evaluating knowledge acquisition from multidisciplinary professional videos. In *ACL*, 2026.

Xiaohu Huang, Hao Zhou, and Kai Han. Prunevid: Visual token pruning for efficient video large language models. In *Findings of ACL*, 2025.

Md Mohaiminul Islam, Ngan Ho, Xitong Yang, Tushar Nagarajan, Lorenzo Torresani, and Gedas Bertasius. Video recap: Recursive captioning of hour-long videos. In *CVPR*, 2024.

Soyeong Jeong, Kangsan Kim, Jinheon Baek, and Sung Ju Hwang. Videorag: Retrieval-augmented generation over video corpus. In *Findings of ACL*, 2025.

Zhengbao Jiang, Frank Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active retrieval augmented generation. In *EMNLP*, 2023.

Kumara Kahatapitiya, Kanchana Ranasinghe, Jongwoo Park, and Michael S Ryoo. Language repository for long video understanding. In *Findings of ACL*, 2025.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *EMNLP*, 2020.

Min Jung Lee, Dayoung Gong, and Minsu Cho. Video summarization with large language models. In *CVPR*, 2025.

Jie Lei, Licheng Yu, Mohit Bansal, and Tamara Berg. Tvqa: Localized, compositional video question answering. In *EMNLP*, 2018.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *NeurIPS*, 2020.

KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023.

Mingxin Li, Yanzhao Zhang, Dingkun Long, Keqin Chen, Sibo Song, Shuai Bai, Zhibo Yang, Pengjun Xie, An Yang, Dayiheng Liu, Jingren Zhou, and Junyang Lin. Qwen3-vl-embedding and qwen3-vl-reranker: A unified framework for state-of-the-art multimodal retrieval and ranking. *arXiv preprint arXiv:2601.04720*, 2026.

Xiaonan Li, Changtai Zhu, Linyang Li, Zhangyue Yin, Tianxiang Sun, and Xipeng Qiu. Llatrieval: Llm-verified retrieval for verifiable generation. In *NAACL*, 2024.

Geuntaek Lim, Sungjune Park, Jaeyun Lee, Inwoong Lee, Taeoh Kim, Dongyoon Wee, Minho Shim, and Yukyung Choi. Video-oasis: Rethinking evaluation of video understanding. *arXiv preprint arXiv:2603.29616*, 2026.

Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. In *EMNLP*, 2024a.

Sheng-Chieh Lin, Chankyu Lee, Mohammad Shoeybi, Jimmy Lin, Bryan Catanzaro, and Wei Ping. Mm-embed: Universal multimodal retrieval with multimodal llms. *arXiv preprint arXiv:2411.02571*, 2024b.

Yijie Lin, Jie Zhang, Zhenyu Huang, Jia Liu, Zujie Wen, and Xi Peng. Multi-granularity correspondence learning from long-term noisy videos. In *ICLR*, 2024c.

Hui Liu and Xiaojun Wan. Video paragraph captioning as a text summarization task. In *ACL*, 2021.

Yikun Liu, Yajie Zhang, Jiayin Cai, Xiaolong Jiang, Yao Hu, Jiangchao Yao, Yanfeng Wang, and Weidi Xie. Lamra: Large multimodal model as your advanced retrieval assistant. In *CVPR*, 2025.

Yongdong Luo, Xiawu Zheng, Guilin Li, Shukang Yin, Haojia Lin, Chaoyou Fu, Jinfa Huang, Jiayi Ji, Fei Chao, Jiebo Luo, and Rongrong Ji. Video-rag: Visually-aligned retrieval-augmented long video comprehension. In *NeurIPS*, 2025.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. Query rewriting in retrieval-augmented large language models. In *EMNLP*, 2023.

Ziyu Ma, Chenhui Gou, Hengcan Shi, Bin Sun, Shutao Li, Hamid Rezatofighi, and Jianfei Cai. Drvideo: Document retrieval based long video understanding. In *CVPR*, 2025.

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. In *ACL*, 2024.

Priyanka Mandikal and Raymond Mooney. Sparse meets dense: A hybrid approach to enhance scientific document retrieval. *arXiv preprint arXiv:2401.04055*, 2024.

Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. In *NeurIPS*, 2023.

Alkesh Patel, Vibhav Chitalia, and Yinfei Yang. Advancing egocentric video question answering with multimodal large language models. *arXiv preprint arXiv:2504.04550*, 2025.

Danila Potapov, Matthijs Douze, Zaid Harchaoui, and Cordelia Schmid. Category-specific video summarization. In *ECCV*, 2014.

Zackary Rackauckas. Rag-fusion: a new take on retrieval-augmented generation. *arXiv preprint arXiv:2402.03367*, 2024.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *ICML*, 2023.

Aniket Rege, Arka Sadhu, Yuliang Li, Kejie Li, Ramya Korlakai Vinayak, Yuning Chai, Yong Jae Lee, and Hyo Jin Kim. Agentic very long video understanding. In *ACL*, 2026.

Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. Timechat: A time-sensitive multimodal large language model for long video understanding. In *CVPR*, 2024.

Xubin Ren, Lingrui Xu, Long Xia, Shuaiqiang Wang, Dawei Yin, and Chao Huang. Videorag: Retrieval-augmented generation with extreme long-context videos. In *KDD*, 2026.

Kunal Sawarkar, Abhilasha Mangal, and Shivam Raj Solanki. Blended rag: Improving rag (retriever-augmented generation) accuracy with semantic search and hybrid query-based retrievers. In *MIPR*, 2024.

Xiaoqian Shen, Wenxuan Zhang, Jun Chen, and Mohamed Elhoseiny. Vgent: Graph-based retrieval-reasoning-augmented generation for long video understanding. In *NeurIPS*, 2025.

Aditi Singh, Abul Ehtesham, Saket Kumar, Tala Talaei Khoei, and Athanasios V Vasilakos. Agentic retrieval-augmented generation: A survey on agentic rag. *arXiv preprint arXiv:2501.09136*, 2025.

Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023.

Shitong Sun, Ke Han, Yukai Huang, Weitong Cai, and Jifei Song. Egograph: Temporal knowledge graph for egocentric video understanding. *arXiv preprint arXiv:2602.23709*, 2026.

Omri Uzan, Asaf Yehudai, Roi Pony, Eyal Shnarch, and Ariel Gera. Guided query refinement: Multimodal hybrid retrieval with test-time optimization. In *ICLR*, 2026.

Han Wang, Yuxiang Nie, Yongjie Ye, Yanjie Wang, Shuai Li, Haiyang Yu, Jinghui Lu, and Can Huang. Dynamic-vlm: Simple dynamic visual token compression for videollm. In *ICCV*, 2025a.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024a.

Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. Videoagent: Long-form video understanding with large language model as agent. In *ECCV*, 2024b.

Yujing Wang, Hainan Zhang, Liang Pang, Binghui Guo, Hongwei Zheng, and Zhiming Zheng. Maferw: Query rewriting with multi-aspect feedbacks for retrieval-augmented large language models. In *AAAI*, 2025b.

Zihan Wang, Zihan Liang, Zhou Shao, Yufei Ma, Huangyu Dai, Ben Chen, Lingtao Mao, Chenyi Lei, Yuqing Ding, and Han Li. Infogain-rag: Boosting retrieval-augmented generation through document information gain-based reranking and filtering. In *EMNLP*, 2025c.

Ziyang Wang, Shoubin Yu, Elias Stengel-Eskin, Jaehong Yoon, Feng Cheng, Gedas Bertasius, and Mohit Bansal. Videotree: Adaptive tree-based video representation for llm reasoning on long videos. In *CVPR*, 2025d.

Ziyang Wang, Honglu Zhou, Shijie Wang, Junnan Li, Caiming Xiong, Silvio Savarese, Mohit Bansal, Michael S Ryoo, and Juan Carlos Niebles. Active video perception: Iterative evidence seeking for agentic long video understanding. *arXiv preprint arXiv:2512.05774*, 2025e.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. Longvideobench: A benchmark for long-context interleaved video-language understanding. In *NeurIPS*, 2024.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. C-pack: Packed resources for general chinese embeddings. In *SIGIR*, 2024.

Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. Video question answering via gradually refined attention over appearance and motion. In *MM*, 2017.

Zeyu Xu, Junkang Zhang, Qiang Wang, and Yi Liu. E-vrag: Enhancing long video understanding with resource-efficient retrieval augmented generation. *arXiv preprint arXiv:2508.01546*, 2025.

Zhucun Xue, Jiangning Zhang, Xurong Xie, Yuxuan Cai, Yong Liu, Xiangtai Li, and Dacheng Tao. Adavideorag: Omni-contextual adaptive retrieval-augmented efficient long video understanding. *arXiv preprint arXiv:2506.13589*, 2025.

Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. Corrective retrieval augmented generation. *arXiv preprint arXiv:2401.15884*, 2024.

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 *CVPR*, 2025.

Songyuan Yang, Weijiang Yu, Ziyu Liu, Guijian Tang, Wenjing Yang, Huibin Tan, and Nong Xiao. Graph-to-frame rag: Visual-space knowledge fusion for training-free and auditable video reasoning. *arXiv preprint arXiv:2604.04372*, 2026.

Linli Yao, Haoning Wu, Kun Ouyang, Yuanxing Zhang, Caiming Xiong, Bei Chen, Xu Sun, and Junnan Li. Generative frame sampler for long video understanding. In *Findings of ACL*, 2025.

Woongyeong Yeo, Kangsan Kim, Soyeong Jeong, Jinheon Baek, and Sung Ju Hwang. Universalrag: Retrieval-augmented generation over corpora of diverse modalities and granularities. In *ACL*, 2026a.

Woongyeong Yeo, Kangsan Kim, Jaehong Yoon, and Sung Ju Hwang. Worldmm: Dynamic multimodal memory agent for long video reasoning. In *CVPR*, 2026b.

Yue Yu, Wei Ping, Zihan Liu, Boxin Wang, Jiaxuan You, Chao Zhang, Mohammad Shoeybi, and Bryan Catanzaro. Rankrag: Unifying context ranking with retrieval-augmented generation in llms. In *NeurIPS*, 2024.

Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *AAAI*, 2019.

Biao Zhang, Paul Suganthan, Gaël Liu, Ilya Philippov, Sahil Dua, Ben Hora, Kat Black, Gus Martins, Omar Sanseviero, Shreya Pathak, et al. T5gemma 2: Seeing, reading, and understanding longer. *arXiv preprint arXiv:2512.14856*, 2025a.

Ce Zhang, Taixi Lu, Md Mohaiminul Islam, Ziyang Wang, Shoubin Yu, Mohit Bansal, and Gedas Bertasius. A simple llm framework for long-range video question-answering. In *EMNLP*, 2024a.

Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023.

Le Zhang, Yihong Wu, Qian Yang, and Jian-Yun Nie. Exploring the best practices of query expansion with large language models. In *Findings of EMNLP*, 2024b.

Xiaoyi Zhang, Zhaoyang Jia, Zongyu Guo, Jiahao Li, Bin Li, Houqiang Li, and Yan Lu. Deep video discovery: Agentic search with tool use for long-form video understanding. *arXiv preprint arXiv:2505.18079*, 2025b.

Xin Zhang, Yanzhao Zhang, Wen Xie, Mingxin Li, Ziqi Dai, Dingkun Long, Pengjun Xie, Meishan Zhang, Wenjie Li, and Min Zhang. Bridging modalities: Improving universal multimodal retrieval by multimodal large language models. In *CVPR*, 2025c.

Penghao Zhao, Hailin Zhang, Qinhan Yu, Zhengren Wang, Yunteng Geng, Fangcheng Fu, Ling Yang, Wentao Zhang, Jie Jiang, and Bin Cui. Retrieval-augmented generation for ai-generated content: A survey. *Data Science and Engineering*, 11:1–29, 2026a.

Yixuan Zhao, Zihao Fan, Yingying Cao, Zhengjia Lyu, and Jingyuan Li. Rewritegen: Autonomous query optimization for retrieval-augmented large language models via reinforcement learning. *Electronics*, 15(5):1058, 2026b.

Wenqi Zhou, Kai Cao, Hao Zheng, Yunze Liu, Xinyi Zheng, Miao Liu, Per Ola Kristensson, Walterio Mayol-Cuevas, Fan Zhang, Weizhe Lin, and Junxiao Shen. X-lebench: A benchmark for extremely long egocentric video understanding. *arXiv preprint arXiv:2501.06835*, 2025.

Yuhan Zhu, Xiangyu Zeng, Chenting Wang, Xinhao Li, Chunxu Liu, Yicheng Xu, Ziang Yan, Yi Wang, and Limin Wang. Freeret: Mllms as training-free retrievers. In *ICML*, 2026.

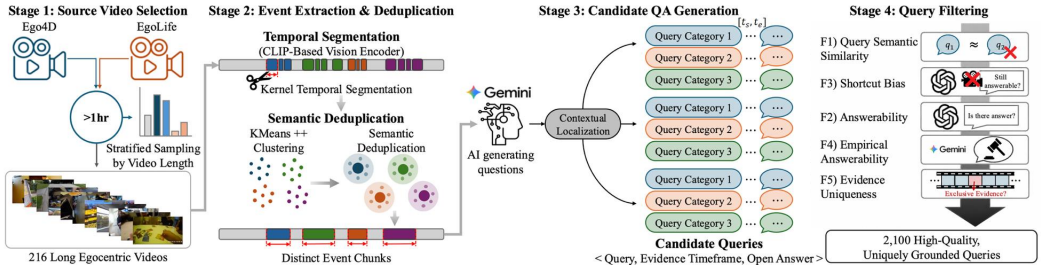

Figure 2: Overview of the V-RAGBench Construction Pipeline. We construct V-RAGBench through four stages: source video selection, event extraction and deduplication, candidate QA generation, and query filtering. Hour-scale egocentric videos are segmented into distinct event chunks, used to generate open-ended QA candidates, and then rigorously filtered for redundancy, answerability, shortcut bias, empirical answerability, and evidence uniqueness, yielding a balanced set of uniquely grounded queries.

A Dataset

The overall pipeline of V-RAGBench construction is described in Figure 2.

A.1 Dataset statistics

V-RAGBench is a benchmark for egocentric video understanding constructed from two complementary sources: long-form daily-life recordings from EgoLife and diverse activity-centric videos from Ego4D. The dataset comprises 2,100 queries grounded in 216 uncut source videos. Detailed dataset statistics, including the distribution of queries across source datasets, categories, domains, and evidence chunk durations, are summarized in Table 7.

A.2 Definitions of the source video domains

Ego4D videos cover heterogeneous activities, while EgoLife videos are uniformly daily-life. We group Ego4D videos into five high-level domains — *Domestic*, *Social*, *Stationary*, *Manual*, and *Outdoor* — by collapsing the Ego4D metadata categories. For each in-distribution domain we sample 36 training and 6 test videos based on the within-domain length distribution; the *Outdoor* domain is held out entirely for test, providing an out-of-domain split. For EgoLife, one of the seven days per participant is sampled as test, with non-overlapping day assignments across participants to prevent leakage. The resulting test set spans both familiar and unseen domains, supporting both standard and out-of-distribution evaluation. Formal definitions of these domains are provided in Table 8.

A.3 Definitions of the query categories

The 2,100 queries are organized into three reasoning categories that target distinct capabilities of egocentric video understanding: *Action Flow & Movement*, which probes temporal and sequential reasoning; *Object-Centric Visual Understanding*, which focuses on fine-grained recognition of object attributes and states; and *Interaction*, which examines both physical manipulation and social engagement. The precise definitions of each category are given in Table 9.

A.4 Event selection and de-duplication details

Event Chunk Construction Details. Section 4 describes the overall event extraction and deduplication procedure. Here we clarify two implementation consequences of this design. First, temporal segmentation and semantic deduplication play separate roles: segmentation determines valid time-contiguous event candidates, while deduplication only selects among those candidates and does not modify their temporal boundaries. Thus, each retained chunk remains an original contiguous video interval.

Table 7: Dataset statistics detailing the distribution of 2,100 queries across categories, domains, and chunk durations, alongside 216 uncut source videos.

Scope	Attribute	Sub-category	Count (% / Details)
Source Videos	Total		216
	Source	EgoLife	42 (Mean: 379.1 min, Range: 177.8–552.7 min)
		Ego4D	174 (Mean: 86.0 min, Range: 60.0–423.6 min)
Queries	Total		2,100
	Source	EgoLife	1,320 (62.86%)
		Ego4D	780 (37.14%)
	Category	Action Flow & Movement	700 (600 Train / 100 Test)
		Interaction	700 (600 Train / 100 Test)
		Object-Centric Visual Understanding	700 (600 Train / 100 Test)
	Domain	Daily	1,320 (62.86%)
		Domestic	203 (9.67%)
		Social	195 (9.29%)
		Stationary	180 (8.57%)
		Manual	169 (8.05%)
		Outdoor	33 (1.57%)
	Evidence Chunk Duration	$<$ 5s	330 (15.71%)
		5–30s	1,350 (64.29%)
		30–60s	168 (8.00%)
		1–5 min	252 (12.00%)

Second, all event-level statistics are computed after deduplication. Since the retained chunks are selected as representatives of semantically similar segment groups, the final evidence pool is intentionally smaller and less redundant than the raw segmentation output. In particular, repeated daily routines are counted only through their selected representative chunks, which reduces the chance that a generated query can be answered from multiple near-duplicate events in the same video.

Because the representatives are not forced into a fixed temporal length, the resulting chunks naturally span a range of durations. We report the duration distribution of these deduplicated event chunks in Table 10 and Table 11.

A.5 QA Filter Details

Table 12 shows that the filtering pipeline is highly selective: only $5{,}907$ of the initial $67{,}370$ candidates remain after all five stages, corresponding to 8.77% of the original candidate pool. This low retention rate reflects the strict grounding requirements of `V-RAGBench`: retained queries must be semantically non-redundant, answerable from the annotated evidence, robust to shortcut bias, empirically answerable by a strong VLLM, and not recoverable from non-evidence chunks in the same video. The $5{,}907$ retained queries form the post-filter candidate pool before category balancing. Since the three query categories are unevenly populated after filtering, we balance the benchmark by downsampling each category to the size of the smallest category, which is *Interaction* (See Table 9). We therefore randomly sample 700 queries per category, yielding the final balanced set of $2{,}100$ queries.

Semantic Similarity Filter. Event-level visual deduplication does not fully eliminate redundancy at the query level, since different retained events can still induce near-identical natural-language questions. We therefore apply this filter over query embeddings rather than event embeddings, removing candidates that are semantically too close to already retained queries. This reduces benchmark inflation from repeated phrasings of similar actions.

Table 8: Egocentric video domains covered in V-RAGBench

Source Dataset	Domain	Definition
EgoLife	Daily	Long-form egocentric daily-life videos from EgoLife, covering extended personal routines across time. Unlike Domestic, which is a specific Ego4D activity domain, Daily refers to broader continuous life-logging episodes that may include household routines, desk work, leisure, transitions, and other ordinary activities across a day.
Ego4D	Domestic	Videos of household or home-centered everyday activities, such as cooking, cleaning, organizing, eating, or moving around indoor living spaces. The key characteristic is routine domestic activity within a home-like environment.
	Manual	Videos centered on hands-on physical manipulation, craftwork, repair, assembly, tool use, or labor-intensive activities. The key characteristic is that the camera wearer is actively manipulating objects or materials.
	Social	Videos involving interpersonal interaction, such as conversation, collaboration, instruction, assistance, or shared activity with other people. The key characteristic is that understanding the scene depends on social context, other people's actions, or communicative behavior.
	Stationary	Low-motion egocentric videos where the camera wearer remains largely fixed in place and attends to a stable target over time, such as a computer screen, television, desk, document, or workspace. The key characteristic is not the presence of a screen itself, but the static first-person viewpoint and sustained attention to the same visual area.
	Outdoor	Videos captured in outdoor or large-scale physical environments, often involving walking, navigation, travel, sports, outdoor work, or other movement-heavy activities. The key characteristic is that the camera wearer interacts with a changing external environment rather than a fixed indoor workspace or household setting.

Table 9: Definitions of the query categories used in the dataset.

Category	Description
Action Flow & Movement	Questions requiring understanding of event sequences, movement, and scene transitions. These questions emphasizes temporal flow before, during, and after key actions. The expected answer must describe a temporal or sequential fact.
Object-Centric Visual Understanding	Questions that require recognizing specific details about objects: identity, visual attributes (color, shape, size), current state, or count. The expected answer must identify a specific object detail.
Interaction	Questions about the camera wearer's interaction with their environment, encompassing both the physical manipulation of objects (how things are held, used, or transferred) and social engagement (roles, relationships, and interpersonal behaviors). The expected answer must describe a physical interaction with an object or a social fact involving other people.

Shortcut Bias Filter. Some generated queries can be answered from language priors or commonsense knowledge without inspecting the video. To identify such cases, we provide GPT-5.2-chat with the query alone, without any visual context, and compare its answer against the reference answer. If the blind answer matches the reference, the query is discarded as shortcut-biased. Table 13 shows examples of queries filtered as they exhibit shortcut bias.

Answerability Filter. For each candidate, we evaluate the paired $(q, \text{event clip})$ with GPT-5.2-chat and ask whether the answer can be determined from the visual evidence. Candidates judged unanswerable from their source clip are discarded.

Empirical Answerability Filter. After the previous filters, we further check whether the query is practically answerable by a strong VLLM using only the source evidence chunk. We run Gemini 2.5-

Table 10: Descriptive statistics of event chunk durations (in seconds) by dataset.

Split	Count	Mean (s)	Median (s)	Min (s)	Max (s)
All (Overall)	19,751	28.03	8.60	4.70	300.00
EgoLife	11,109	31.65	7.60	4.70	300.00
Ego4D	8,642	23.39	11.00	5.00	300.00

Table 11: Distribution of event chunk durations scaled to the final subset of 2,100 queries.

Duration Bin (s)	Count	Percentage
0–5	331	15.78%
5–10	810	38.59%
10–30	594	28.27%
30–60	167	7.97%
60–120	77	3.66%
120–300	116	5.51%
300–500	5	0.22%
Total	2,100	100.00%

flash on the paired clip and evaluate the generated answer with the binary LLM-as-a-judge described in Appendix B.4. Queries that the model fails to answer correctly are removed, since they are unlikely to be reliably answerable in downstream evaluation.

Evidence Uniqueness Filter. To test whether the annotated evidence is uniquely sufficient, we simulate retrieval over non-evidence clips from the same source video. Each video is split into 2-minute chunks, which are embedded with Qwen3-VL-Embedding-8B. For every query, we retrieve the top-10 most similar chunks while explicitly excluding the chunk containing the true evidence. Gemini 2.5-flash is then asked to answer using only this non-evidence retrieval set, and the output is judged with the same binary LLM-as-a-judge as in the empirical answerability filter. If the answer is still judged correct, the query is removed because its answer is recoverable from elsewhere in the video rather than uniquely grounded in the annotated evidence. Figure 3 presents the case study of queries whose evidence is not uniquely grounded in a single evidence chunk.

A.6 Human Verification of Query Filters

This section details the human verification summarized in Table 1 of Section 4, which consists of three checks, answerability, evidence uniqueness, and query redundancy.

Annotators and Protocol. Every item in each check is verified independently by two annotators. Annotators are graduate students with at least one first-author publication at a top-tier AI conference and English proficiency equivalent to CEFR C2. Each annotator received a 30-minute instructional briefing before annotation, and for the exhaustive audits additionally completed standardized written training and a calibration session. Annotators label each item without access to the other annotator's judgment, and each check reports the resulting inter-annotator agreement (IAA). We report Cohen's κ (Cohen, 1960) for answerability and Gwet's AC1 (Gwet, 2008) for the two exhaustive audits, since κ is undefined when no positive case exists and severely understates agreement when positives are around 3%.

Answerability. We sample 577 QA pairs at random, corresponding to a 95% confidence interval with a margin of ± 4 over the full set of 2,100 queries (1,154 judgments). Each annotator is provided with the source video clip, the query, and the ground-truth answer, and judges whether the answer can be reasonably inferred from the video. 96.9% of the sampled queries are judged answerable by both annotators ($\kappa = 0.63$, substantial agreement), confirming that retained queries are reliably answerable from their evidence.

Query Redundancy. This audit verifies query-level non-redundancy, the property targeted by the Semantic Similarity filter. All 2,648 pairs of test queries drawn from the same video were judged for

Table 12: Filtering statistics for the combined dataset. The table reports the number of queries remaining and the number filtered at each major post-hoc filtering stage. Percentages are relative to the initial candidate count.

Filtering Stage	Remaining	Filtered (%)
Initial Candidates	67,370	–
Semantic Similarity	52,783	14,587 (21.65%)
Shortcut Bias	28,331	24,452 (36.30%)
Answerability	19,751	8,580 (12.74%)
Empirical Answerability	11,041	8,710 (12.93%)
Evidence Uniqueness	5,907	5,134 (7.62%)
Final	5,907	(8.77% kept)

Table 13: Examples of benchmark video queries with shortcut bias, which are answerable *without* access to the ground-truth source video based on context and parametric knowledge.

Q1: While I am standing at the wooden island with the carton of milk and metal mixing bowls, what is the color of the liquid inside the clear plastic measuring cup?

Blind prediction: White. *GT:* White.

Why answerable: The question explicitly mentions a carton of milk on the same surface as the measuring cup. Common sense strongly suggests the liquid being measured is the milk, making "white" inferable without watching the video.

Q2: What visible item indicates the woman's professional role while she sits and talks to me?

Blind prediction: A visible name badge or ID badge. *GT:* A badge.

Why answerable: The phrase "visible item indicates professional role" is a near-defining description of a name badge or ID badge. Parametric knowledge of workplace conventions makes this the default answer regardless of video content.

Q3: Upon approaching the man sitting at the white table, how does the person acknowledge my presence?

Blind prediction: He looks up and greets you (e.g., says hello or nods). *GT:* By greeting.

Why answerable: Standard social conventions dictate that acknowledging an approaching person involves looking up and greeting them. The query frames the action as "acknowledge," which already implies a greeting-style response.

semantic duplication (5,296 judgments). Neither annotator flagged any pair, with 100% agreement (AC1=1.0).

Evidence Uniqueness. This audit verifies that each query is grounded in a single evidence chunk over the entire video, covering properties (i) and (iii) in Section 4, and uses no retriever. Annotators watched all 2,094 two-minute windows of the test videos and judged, for each of the 10,206 window–query pairs, whether the query is answerable from that window alone (20,412 judgments). 97.1% of pairs outside the labeled evidence were judged unanswerable by both annotators (95% CI 96.8–97.5%, raw agreement 95.95%, AC1=0.955), which establishes evidence uniqueness over the full corpus independently of any retriever.

B Implementation details

Model Specification. Table 14 lists all open-source models used in our experiments, while Table 15 summarizes the proprietary models. The total API cost incurred in our experiments is approximately $1,200, including both QA generation and QA filtering.

Resources. All experiments run on $8\times$ NVIDIA H200 (141 GB) GPUs with CUDA 13.0.

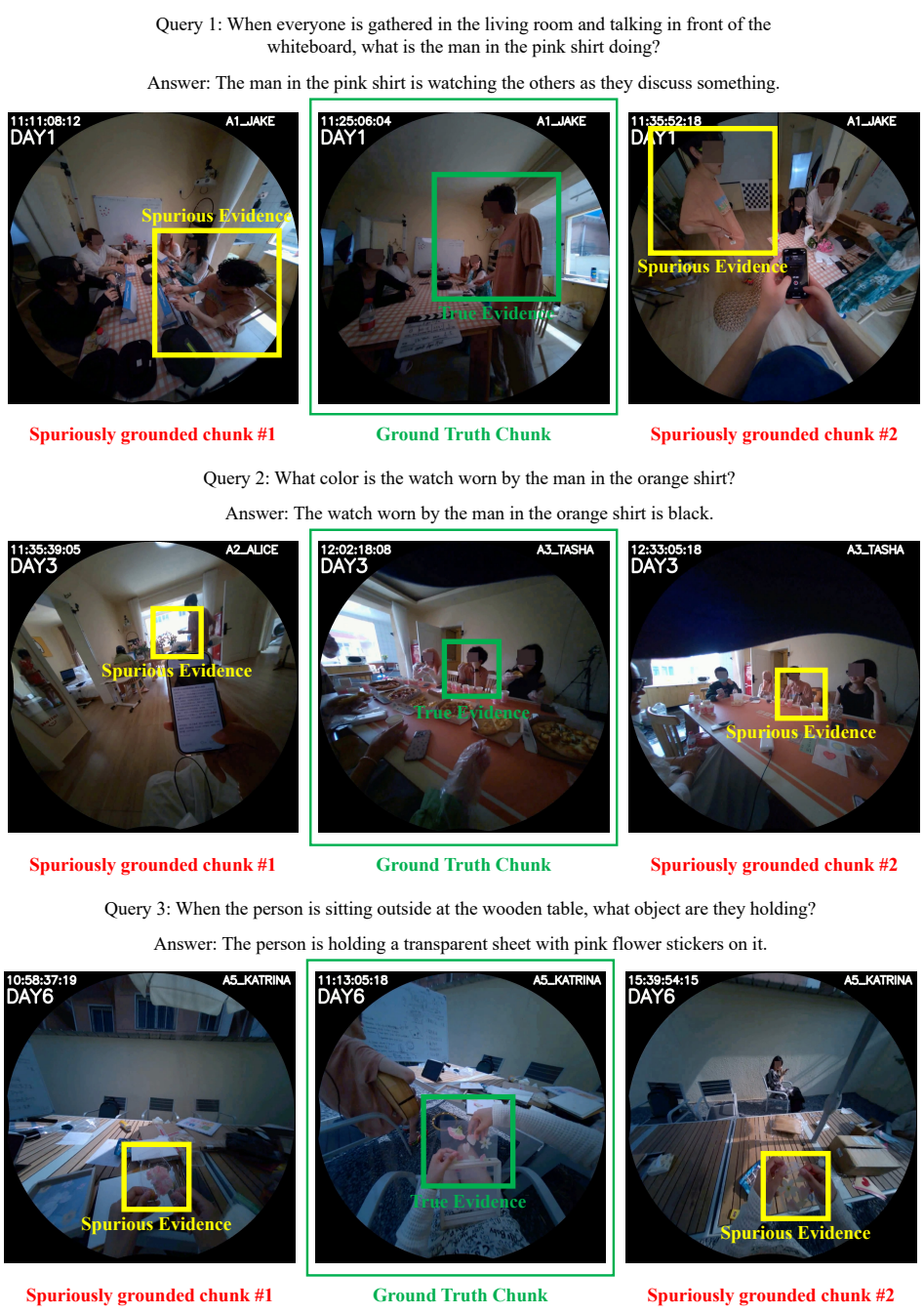

Figure 3: Examples of the queries not grounded with a unique evidence chunk.

B.1 CARVE IMPLEMENTATIONS.

Representation Construction. We split each video into non-overlapping 120-second segments and build four representations per segment, indexed identically across representations. The VF encodes 5 keyframes per segment; the VC encodes up to 64 frames per segment (maximum 1,843,200 pixels per frame, sampled at 1 fps). The TF view captions the 5 keyframes with Qwen3-VL-8B-Instruct (Bai et al., 2025) up to 512 tokens; the TC view captions per-second frames at 1 fps.

Table 14: Pretrained models used in our pipeline.

Model	Checkpoint	Purpose
Qwen3-VL-8B-Instruct	Qwen/Qwen3-VL-8B-Instruct	Video QA, VLM for construct Text Clip and Text Frames in Section 3
Qwen3-VL-32B-Instruct	Qwen/Qwen3-VL-32B-Instruct	Video QA
Gemma-4-26B-A4B	google/gemma-4-26B-A4B	Video QA
EVA02-E-14-plus	timm/eva02_enormous_patch14_plus_clip_224.laion2b_s9b_b144k	Frame embedding extraction for segmentation in Section 4
Qwen3.6-35B-A3B	Qwen/Qwen3.6-35B-A3B	LLM as Judge in Section 6
Qwen3-VL-Embedding-2B	Qwen/Qwen3-VL-Embedding-2B	Embedding model in Section 5
Qwen3-VL-Reranker-2B	Qwen/Qwen3-VL-Reranker-2B	Reranking in Section 5
LamRA-Rank-7B	code-kunkun/LamRA-Rank	Reranking in Section 5
Qwen3-VL-Embedding-8B	Qwen/Qwen3-VL-Embedding-8B	Semantic similarity check in Section 4

Table 15: Proprietary API used.

Model	Purpose
gemini-3-flash-preview	QA generation in Section 4
gemini-2.5-flash	Uniqueness, empirical answerability filtering in Section 4
gpt-5.2-chat-latest	Answerability, Shortcut bias filter in Section 4

Retrieval and Generation. At inference, we retrieve the top-$K{=}5$ chunks per query, with reranking using the default Qwen3-VL-Reranker-2B (Li et al., 2026). The retrieved chunks are passed to Qwen3-VL-8B-Instruct with a maximum of 512 new tokens under its default decoding setup. Every generated answer is judged against the reference by a binary judge (Qwen3.6-35B-A3B), using identical judge weights and prompt across our method and all baselines.

B.2 Baseline Implementations

System-level Baselines.VideoRAG-A (Jeong et al., 2025) is a retrieval method that encodes each video segment by linearly interpolating its textual and visual representations. We set α as 0.5 in this set up. **GQR** (Uzan et al., 2026) refines the retriever's query embedding by gradient descent against a mixture of the two retriever's score distributions, using KL minimization. We set two retrievers as text and video retrievers with learning rate $5\mathrm{e}{-4}$. **FreeRet** (Zhu et al., 2026) repurposes an MLLM (Qwen2-VL-2B-Instruct (Wang et al., 2024a) in our setup) as a training-free two-stage retriever, extracting segment embeddings from the last self-attention sublayer and reranking the candidates via MCQ-based scoring. **GME** (Zhang et al., 2025c) is a unified multimodal embedding model based on Qwen2-VL-2B-Instruct (Wang et al., 2024a). We apply both methods directly to the raw video segments and query.

Query-level Baselines.VideoRAG-B (Luo et al., 2025) prompts Qwen3-VL-8B-Instruct (Bai et al., 2025) to decouple each query into three memory-specific sub-queries, retrieves from each memory, and merges the results before passing them to the generator. Following the original paper, we use one visual retriever and two text retrievers. **RRF** (Rackauckas, 2024) is the RAG-Fusion variant of reciprocal rank fusion: the MLLM (Qwen3-VL-8B-Instruct (Bai et al., 2025)) rewrites each query into two modality-targeted paraphrases — one for text-clip, one for text-frames — which drive two retrievers operating on the text-clip and text-frame embeddings, respectively. The resulting ranked lists are aggregated by the standard RRF formula with $k{=}60$. **DAT** (Hsu & Tzeng, 2025) adaptively balances two retrievers per query using LLM. We set the paper's dense and sparse setting with text clip and frames, α that interpolates the min-max-normalized similarities over the candidate union. **UniversalRAG** (Yeo et al., 2026a) routes each query to a single optimal (m, g) configuration using a trained classifier and applies the selected configuration uniformly to all retrieved chunks. We use LoRA-based router(Qwen3-VL-2B-Instruct) and a encoder-based router (T5Gemma 2 270M (Zhang

et al., 2025a) with a linear classification head) and replace the paper's sigmoid-thresholded decoding with argmax at inference for consistency with the top-K retrieval setting used by all other baselines.

Router Implementations. To evaluate the benefits of chunk-level decisions in generation, we compare `CARVE` against two types of query-level routers that select a single configuration (m, g) per query for generation. To train LLM-based Router, we fine-tune Qwen3-VL-2B-Instruct (Bai et al., 2025) as a multi-label classifier over the four configurations to find best generation configuration. We apply LoRA ($r = 32$) and train for 5 epochs with a $2\mathrm{e}{-5}$ learning rate. Labels are derived from our train set, where a configuration is labeled positive if it generates a correct answer. At inference, we select the configuration with the highest predicted score (argmax) and apply it uniformly to all retrieved chunks for generation. To train encoder based router for generation, we train an MLP-based classifier on top of a frozen BGE-large-en-v1.5 (Xiao et al., 2024) encoder. We use the frozen `[CLS]` representation as input to a classification head consisting of three hidden blocks with a size of 512. The model is optimized using binary cross-entropy with a $1\mathrm{e}{-3}$ learning rate. Similar to the LLM router, we use *argmax* at inference to determine a single, query-uniform configuration.

B.3 Evaluation Metrics

We evaluate retrieval and generation separately. Let $\mathcal{Q}$ denote the set of evaluation queries. For each query $q \in \mathcal{Q}$, let G_q be the set of ground-truth evidence chunks annotated for q, and let $R_q^K = [r_1, r_2, \ldots, r_K]$ be the ranked list of the top-K retrieved chunks. In our main experiments, we set $K = 5$.

Recall@K. Recall@K measures whether the ground-truth evidence appears in the top-K retrieved chunks. For a query q, we define

$$\mathrm{Recall@K}(q) = \frac{|R_q^K \cap G_q|}{|G_q|}. \tag{8}$$

The dataset-level score is the average over all queries:

$$\mathrm{Recall@K} = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \mathrm{Recall@K}(q). \tag{9}$$

Since each query in `V-RAGBench` is designed to have a uniquely sufficient evidence chunk, this reduces to checking whether that evidence chunk is retrieved within the top-K.

nDCG@K. nDCG@K evaluates not only whether the evidence is retrieved, but also how highly it is ranked. We use binary relevance:

$$\mathrm{rel}_i = \mathbb{1}[r_i \in G_q], \tag{10}$$

where r_i is the chunk ranked at position i. The discounted cumulative gain is

$$\mathrm{DCG@K}(q) = \sum_{i=1}^{K} \frac{\mathrm{rel}_i}{\log_2(i+1)}. \tag{11}$$

The ideal discounted cumulative gain is

$$\mathrm{IDCG@K}(q) = \sum_{i=1}^{\min(K, |G_q|)} \frac{1}{\log_2(i+1)}. \tag{12}$$

Then,

$$\mathrm{nDCG@K}(q) = \frac{\mathrm{DCG@K}(q)}{\mathrm{IDCG@K}(q)}. \tag{13}$$

The final nDCG@K is averaged over all queries:

$$\mathrm{nDCG@K} = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \mathrm{nDCG@K}(q). \tag{14}$$

Pass Rate. For generation, we report aggregate pass rate. Let $\hat{a}_q$ be the model-generated answer for query q, and let a_q be the ground-truth answer. We use an LLM-as-a-judge $J(\cdot)$ to determine whether $\hat{a}_q$ is correct:

$$J(q, \hat{a}_q, a_q) \in \{0, 1\}, \tag{15}$$

where 1 indicates that the generated answer matches the key factual content of the reference answer. The pass rate is

$$\text{PassRate} = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} J(q, \hat{a}_q, a_q). \tag{16}$$

Latency. We additionally report latency for experiments where methods differ in online inference cost. For each query q, let T_q denote the wall-clock time required by the evaluated online pipeline. Depending on the experiment, this includes query encoding, retrieval, reranking, evidence rendering, and generation, but excludes offline preprocessing such as video chunking, memory construction and index construction. The average latency is

$$\text{Latency} = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} T_q. \tag{17}$$

When reporting retrieval-only latency, T_q includes only the online retrieval and reranking steps; when reporting end-to-end latency, it additionally includes generation.

B.4 LLM-AS-A-JUDGE DETAILS

For evaluating answer correctness on open-ended questions, we use Qwen3.6-35B-A3B as the backbone model for the LLM-as-a-judge. The prompt used for judgment is shown in Figure 8.

Human Verification. To assess the reliability of the LLM-as-a-judge, we conduct human validation of its outputs at two levels, a random sample over the full set of predictions and an exhaustive audit of the test set. We apply the same annotator qualification criteria and protocol as in Appendix A.6. Each annotator is provided with the model-generated answer, the ground-truth answer, and the binary decision from the LLM-as-a-judge indicating whether the prediction is correct, and independently assesses whether the judge's label is correct.

For the sampled verification, we randomly sample 600 instances from the full set of $2{,}100$ predictions (corresponding to a 95% confidence interval with a margin of error of ± 4). We find substantial agreement between human annotators and the LLM-as-a-judge, with a Cohen's Kappa score of 0.61, and both annotators agree that the judge made an incorrect decision in only 2.2% of the instances.

For the exhaustive audit, one evaluation run over the test set yields 300 binary decisions, and both annotators verified each of them, 600 judgments in total. 99.0% of the decisions were upheld, and only $3/300$ were unanimously flagged as incorrect. At the level of individual judgments, LLM–human agreement is 95.8% (Cohen's κ 0.91) and human–human agreement 93.7% (κ 0.87), so the judge is as consistent with human annotators as they are with each other. Since the labels are balanced in this audit, κ is well-defined, and Gwet's AC1 concurs (0.88–0.92). These results confirm that the binary LLM-as-a-judge is reliable for evaluating open-ended answers at scale.

C CARVE ALGORITHM

Algorithm 1 shows the formalization of the algorithm of CARVE.

Algorithm 1 CARVE: Chunk-Aware Reranking for Video Evidence

Require:
Per-wearer video corpus $\mathcal{V}$ (2-min non-overlapping chunks)
Query q
Multimodal encoder f (Qwen3-VL-Embedding-2B by default)
Multimodal cross-encoder $\mathrm{CE}(\cdot,\cdot)$ (Qwen3-VL-Reranker-2B by default)
Captioner MLLM (Qwen3-VL-8B) for textual summaries (cap 512 tokens)
Keyframe count n (default 5)
Per-configuration retrieval cap k_0, final list size k
Generator $\mathcal{G}$

Ensure:
Multi-representation index $\mathcal{I}$
Tagged final top-k list $V_q^\star = \{(v,\ (m_v^\star, g_v^\star))\}$
Predicted answer $\widehat{a}$

OFFLINE: CHUNK-WISE MULTI-REPRESENTATION INDEXING

for each chunk $v \in \mathcal{V}$ **do**
  $\mathcal{K}(v) \leftarrow n$ keyframes from k-means++ *centroids* over per-frame embeddings of v
    ▷ centroids cover distinct visual states, not uniform time samples
  $\phi_{\mathrm{vis,frame}}(v) \leftarrow \mathrm{MeanPool}\big(\{f(\kappa) : \kappa \in \mathcal{K}(v)\}\big)$
  $\phi_{\mathrm{vis,clip}}(v) \leftarrow f(v)$ ▷ single embedding over the entire chunk
  $\phi_{\mathrm{text,frame}}(v) \leftarrow f\big(\,\|_{\kappa\in\mathcal{K}(v)}\,\mathrm{Summarize}(\kappa)\,\big)$
    ▷ per-keyframe summaries concatenated before encoding
  $\phi_{\mathrm{text,clip}}(v) \leftarrow f(\mathrm{Summarize}(v))$
  $\mathcal{I} \leftarrow \mathcal{I} \cup \big\{\,(m, g, v,\ \phi_{m,g}(v)) : (m,g) \in \{\mathrm{vis},\mathrm{text}\} \times \{\mathrm{frame},\mathrm{clip}\}\,\big\}$
end for

ONLINE: RETRIEVAL, RERANKING, AND GENERATION

// Stage 1: Parallel candidate pooling — one query embedding, four index shards
Compute $f(q)$ once
$\mathcal{P}(q) \leftarrow \emptyset$ ▷ configuration-tagged candidate pool
for each $(m,g) \in \{\mathrm{vis},\mathrm{text}\} \times \{\mathrm{frame},\mathrm{clip}\}$ **do**
  $C_{(m,g)}(q) \leftarrow \text{Top-k}_{0\,v\in\mathcal{V}}\ \langle f(q),\ \phi_{m,g}(v)\rangle$ ▷ cf. Eq. 2
  $\mathcal{P}(q) \leftarrow \mathcal{P}(q) \cup \big\{\,(v,\ (m,g)) : v \in C_{(m,g)}(q)\,\big\}$ ▷ cf. Eq. 3
end for

// Stage 2: Chunk-adaptive reranking — score each candidate under the tag that surfaced it
for each $(v,\ (m,g)) \in \mathcal{P}(q)$ **do**
  $\tilde{s}\big(q;\ (v,(m,g))\big) \leftarrow \mathrm{CE}(q,\ \phi_{m,g}(v))$ ▷ cf. Eq. 4
end for

// Per-chunk winning configuration: keep each chunk's highest-scoring tag
for each chunk v appearing in $\mathcal{P}(q)$ **do**
  $(m_v^\star, g_v^\star) \leftarrow \underset{(m,g)\,:\,(v,(m,g))\in\mathcal{P}(q)}{\arg\max}\ \tilde{s}\big(q;\ (v,(m,g))\big)$ ▷ cf. Eq. 5
end for

// Final cross-modal top-k: rank chunks by their winning scores on a comparable scale
$V_q^\star \leftarrow \Big\{\,(v,\ (m_v^\star, g_v^\star)) : v \in \text{top-}k \text{ chunks by } \tilde{s}\big(q;\ (v,(m_v^\star, g_v^\star))\big)\Big\}$ ▷ cf. Eq. 6

// Modality-interleaved generation: each chunk speaks through its winning representation
$\widehat{a} \leftarrow \mathcal{G}\Big(q,\ \big\{\,\phi_{m_v^\star, g_v^\star}(v)\,\big\}_{(v,\,(m_v^\star,g_v^\star))\,\in\,V_q^\star}\Big)$ ▷ cf. Eq. 7
return $\mathcal{I},\ V_q^\star,\ \widehat{a}$

D ADDITIONAL ANALYSIS

This section provides additional analyses that support the main results. We first verify the statistical robustness of the reported gains through repeated runs and an extended query set (Appendix D.1), and show that the advantage of CARVE does not stem from reranking alone (Appendix D.2). We then examine generalization to third-person long videos beyond the egocentric domain (Appendix D.3). Next, we analyze failure cases (Appendix D.4) and the robustness of CARVE to the retrieval depth and the reranker backbone (Appendices D.5 and D.6). We further verify that reranker scores are comparable across modalities through distribution and normalization analyses (Appendix D.7). Finally, we show that CARVE scales to an expanded configuration space with ASR and OCR memories (Appendix D.8), and that it can be integrated into agentic frameworks as a retrieval module and compared with a memory-based framework (Appendix D.9).

D.1 STATISTICAL ROBUSTNESS

Significance Testing. Since retrieval is deterministic, we test significance on the generation pass rate. We run the five methods with the highest average pass rate three times per backbone under identical settings and compare CARVE against the second-best method with a two-sided Welch's t-test, which gives $p < 0.05$ on every backbone. Table 16 reports the mean and run-to-run standard deviation, and the means are reported in Table 2. The run-to-run standard deviation is smaller than the margin of CARVE over the runner-up on every backbone. The remaining methods are omitted because their gaps to CARVE far exceed this variance.

Table 16: Mean and run-to-run standard deviation of the generation pass rate over three runs. $^*p < 0.05$ against the second-best method (Welch's t-test).

Method	Qwen3-VL-8B	Qwen3-VL-32B	Gemma-4-26B
VideoRAG-A	0.2378 ± 0.0201	0.3123 ± 0.0040	0.3045 ± 0.0021
VideoRAG-B	0.3117 ± 0.0136	0.3034 ± 0.0204	0.2355 ± 0.0051
UniversalRAG	0.2588 ± 0.0297	0.2534 ± 0.0115	0.2279 ± 0.0252
GME	0.2533 ± 0.0115	0.2823 ± 0.0139	0.2116 ± 0.0044
CARVE	$\mathbf{0.3523} \pm 0.0108^*$	$\mathbf{0.3680} \pm 0.0010^*$	$\mathbf{0.3180} \pm 0.0051^*$

Consistency on an Extended Query Set. The official test split is fixed to preserve its balance across the three query categories and the held-out Outdoor domain (Appendix A.2). To confirm that method rankings do no on the particular queries in this split, we add 200 randomly drawn queries that were not used to train any method and evaluate all methods on the resulting 500 queries. The pass-rate rankings are virtually identical between the two sets. Spearman ρ is 1.000 on Qwen3-VL-8B, a perfect match, and 0.988 on Qwen3-VL-32B and Gemma-4-26B, where only one of 45 method pairs is reversed. CARVE ranks first on all three backbones in both sets, so the reported gains reflect the methods rather than the sampled queries.

D.2 BASELINE WITH RERANKING

To verify that CARVE's gains are not solely attributable to the reranking stage, we apply all baselines with the same reranker used in our method. As illustrated in Table 17, while the addition of the reranker generally yields performance improvements, these gains remain marginal and, CARVE significantly outperforms even the reranked baseline. This result demonstrates that the advantage of CARVE does not merely stem from the reranking mechanism as a standalone tool. Instead, it originates from the system's unique ability to select and interleave the most discriminative modality-granularity configurations at the chunk level.

D.3 GENERALIZATION TO THIRD-PERSON LONG VIDEOS

We further evaluate whether CARVE generalizes beyond the challenging egocentric videos. Since CARVE selects the configuration of each chunk based on its content, the selected configurations can

Table 17: Baselines with reranking. Best values in bold.

Method	Recall@5	nDCG@5
VideoRAG-A	0.510	0.343
VideoRAG-B	0.487	0.330
GQR	0.503	0.341
RRF	0.463	0.324
UniversalRAG	0.447	0.324
UniversalRAG-LORA	0.470	0.344
DAT	0.460	0.315
FreeRet	0.263	0.212
GME	0.413	0.305
CARVE	**0.603**	**0.433**

Table 18: Retrieval and generation performance on LongVideoBench validation set. Best values in bold.

	Retrieval	Generation		
Method	nDCG@5	Qwen3-VL-8B	Qwen3-VL-32B	Gemma-4-26B
VideoRAG-A	0.504	0.631	0.709	0.631
GQR	0.466	0.589	0.567	0.518
FreeRet	0.360	0.638	0.674	0.539
GME	0.442	0.652	0.645	0.567
RRF	0.393	0.596	0.624	0.560
DAT	0.440	0.617	0.631	0.610
VideoRAG-B	0.494	0.660	0.674	0.624
UniversalRAG	0.496	0.667	0.709	0.638
UniversalRAG-LoRA	0.481	0.638	0.660	0.631
CARVE	**0.545**	**0.695**	**0.766**	**0.645**

naturally vary across different video domains. We evaluate CARVE against eight baselines on the validation set of LongVideoBench (Wu et al., 2024), which contains third-person long videos spanning ten categories, including lectures, movie recaps, and news. To provide a sufficiently challenging retrieval setting, we restrict the evaluation to the longest duration group, (900, 3600] seconds, resulting in 141 questions. Because LongVideoBench does not guarantee a unique supporting chunk for each query, we report nDCG@5 as a reference metric and focus primarily on generation accuracy. As shown in Table 18, CARVE achieves the highest nDCG@5 and the best generation accuracy across all three generators. These results show that the benefits of chunk-level configuration selection extend beyond egocentric videos to third-person long videos across diverse domains.

D.4 Error Analysis on Retrieval and Generation

Table 19: Retrieval cases where generation fails, across two generator scales.

Case	Retrieval	Qwen3-VL-8B	Qwen3-VL-32B
Case 1	Fail	89	86
Case 2	Hit	105	104

To analyze the distribution of failures, we categorize the failed queries into two types under the Qwen3-VL-8B and Qwen3-VL-32B generators. In *Case 1*, both retrieval and generation fail. In *Case 2*, retrieval succeeds but the generator fails to produce the correct answer. As shown in Table 19, both failure types occur at comparable rates across the two generator scales, with Case 2 slightly more frequent than Case 1 for both the 8B and 32B generators.

Table 20: Representative failure cases of CARVE on V-RAGBench.

	Case 1. Retrieval failure	Case 2. Generation failure
Query	After I pass the red refrigerator in the hallway, what room do I enter next?	After I place a rolled tortilla on the black stovetop, what is the next action I perform?
Retrieval	Fail. The ground-truth chunk is not retrieved.	Hit. The ground-truth chunk is retrieved.
Ground truth	A room with a whiteboard and a projector screen.	Picking up a fresh tortilla.
Generated answer	*"After passing the red refrigerator in the hallway, you enter the dining area . . . which contains a large table covered with a checkered tablecloth, flowers, and groceries."*	*"The next action you perform to continue the assembly is adding more filling to the tortilla. . . . after placing the first rolled tortilla on the griddle, you immediately pick up another tortilla, scoop filling from the pan, and roll it."*
Cause	The query anchor recurs throughout the video, so a visually similar but incorrect occurrence is retrieved.	The correct action appears in the reasoning, but the generator fails to order the actions in time.

Qualitative analysis. To understand the source of each failure type, we examine a representative example for each case in Table 20. In Case 1, the key anchor in the query, the red refrigerator, appears multiple times throughout the video. The reranker assigns high scores to these similar occurrences, causing the ground-truth chunk to be missed during retrieval. This highlights a key challenge in long-video retrieval. In Case 2, the correct chunk is successfully retrieved, yet the model still produces an incorrect answer. The generated response correctly identifies the next action, picking up another tortilla, while describing the evidence, but ultimately answers with a later action, adding more filling. The generator therefore identifies the relevant evidence but fails to preserve the temporal order of the actions, indicating a temporal reasoning failure rather than a retrieval or recognition failure.

D.5 Effect of Top-K

We also vary the final retrieval depth k to test whether CARVE is sensitive to the number of chunks passed to the generator. This setting is important because increasing k can improve recall by adding more candidate evidence, but it can also introduce distractor chunks that make generation harder. A robust VideoRAG method should therefore maintain strong performance across different retrieval depths, rather than relying on a single tuned value of k.

Across the tested values of k, CARVE remains stable and consistently competitive, showing that its gains do not depend on a particular retrieval budget. This indicates that chunk-wise parallel reranking improves the ordering of evidence itself, rather than merely benefiting from retrieving more chunks. In other words, CARVE is effective because it selects better-ranked evidence, not because it relies on a larger context window. This experiment shows that **chunk-adaptive reranking produces a robust evidence ranking across retrieval depths, making CARVE less sensitive to the final top-k choice**.

D.6 Reranker Backbone Robustness

Table 21 shows the retrieval performance of each representation configuration under a different reranker backbone, LamRA-Rank-7B. As with Qwen3-VL-Reranker-2B, CARVE achieves the best R@5 and nDCG@5, outperforming both the strongest single configuration and the Concatenation variant while requiring less latency than Concatenation. Since LamRA-Rank-7B differs from Qwen3-VL-Reranker-2B in both model architecture and prompt template, this confirms that the advantage of chunk-level configuration selection does not depend on a particular reranker or prompt format.

D.7 Score Calibration Analysis

In CARVE, candidates retrieved under textual and visual configurations are ranked together by their raw cross-encoder scores. This design assumes that the scores are comparable across modalities,

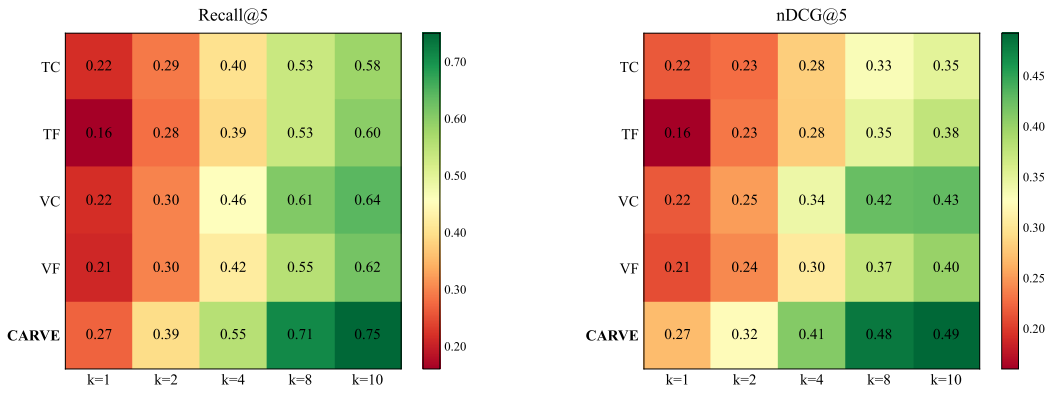

Figure 4: Heatmaps of Recall@k and nDCG@k under different final retrieval depths.

Table 21: Reranking configuration choice under CARVE with LamRA-Rank-7B. Best retrieval values in bold.

Method	R@5	nDCG@5	Latency
m={text}, g={frame}	0.520	0.339	1.0s
m={text}, g={clip}	0.503	0.318	0.5s
m={vis}, g={frame}	0.537	0.357	4.5s
m={vis}, g={clip}	0.563	0.380	8.7s
Random	0.537	0.347	4.4s
Concatenation	0.543	0.367	11.3s
CARVE	**0.577**	**0.401**	4.0s

since a systematic bias toward one modality would make the final ranking reflect the modality rather than evidence relevance. We verify this assumption from three perspectives. All scores lie on the $[0, 1]$ scale of Qwen3-VL-Reranker-2B.

Overall score distributions. Table 22 reports the Wasserstein-1 distance (W_1) between score distributions. The distance between textual and visual candidates is only 0.074, and every pairwise distance among the four configurations stays below 0.1. Moreover, the distance within the textual modality is 0.066, which is nearly as large as the cross-modal distance, whereas two cross-modality pairs are even smaller at 0.053 and 0.054. This shows that score differences vary across individual configurations rather than modalities, indicating no bias toward either modality.

Scores in the top-ranked region. We further examine the high-score region, where candidates from different modalities directly compete for the final top-k. In this region, the textual and visual distributions nearly coincide, with 15.2% of textual candidates and 14.9% of visual candidates scoring 0.6 or higher. The upper percentiles also show no consistent direction, as the 75th percentile is 0.527 for text and 0.555 for visual, whereas the 90th percentile is 0.652 for text and 0.629 for visual. The minor distributional gap in Table 22 therefore arises from the low score range, which rarely affects the final top-k ranking. This is consistent with Table 4, where all four configurations are regularly selected at every rank, with no modality dominating.

Score normalization. To examine whether the cross-modal reranking in CARVE is biased toward a particular modality, we compare it with variants that remove scale differences through normalization. For each query, we normalize the reranker scores in the candidate pool using per-modality z-score, per-modality min-max, or per-configuration z-score normalization, and then reselect the final top-k from the same pool based on the normalized scores. As shown in Table 23, all three methods degrade retrieval. This indicates that the raw score differences across configurations are not a modality bias but a relevance signal, since a configuration that reveals a chunk well assigns it a higher score.

Table 22: Wasserstein-1 distance between reranker score distributions of different configurations.

Pair	Type	W_1
Text (all) vs. Visual (all)	Modality	0.074
Text clip vs. Text frame	Within text	0.066
Visual clip vs. Visual frame	Within visual	0.004
Text frame vs. Visual frame	Cross-modality	0.053
Text frame vs. Visual clip	Cross-modality	0.054
Text clip vs. Visual clip	Cross-modality	0.099
Text clip vs. Visual frame	Cross-modality	0.098

Table 23: Effect of reranking score normalization. Best values in bold.

Method	R@5	nDCG@5
Z-score (modality)	0.580	0.354
Min-max (modality)	0.583	0.347
Z-score (configuration)	0.547	0.347
Raw score (CARVE)	**0.603**	**0.433**

D.8 Scalability to an Expanded Configuration Space

Table 24: Accuracy on Video-MMMU with six configurations. Best values in bold.

Method	Qwen3-VL-8B	Qwen3-VL-32B	Gemma-4-26B
VideoRAG-A	0.705	0.773	0.761
VideoRAG-B	0.716	0.750	0.716
UniversalRAG-LLM	0.682	0.705	0.727
DAT	0.705	0.727	0.727
RRF	0.659	0.739	0.716
CARVE	**0.750**	**0.784**	**0.773**

Effectiveness with additional configurations. Since parallel candidate pooling and chunk-adaptive reranking are agnostic to the number of configurations, CARVE admits new modality-specific memories without architectural changes. We verify this on Video-MMMU (Hu et al., 2026), a benchmark of lecture videos in which spoken narration and on-screen text carry essential information. We include videos longer than ten minutes and questions that require audio or on-screen text, resulting in 88 questions in total. We then extend the configuration space from four to six by adding an ASR memory transcribed with whisper-large-v3-turbo (Radford et al., 2023) and an OCR memory extracted with PP-OCRv5 (Cui et al., 2026). We compare against baselines that can incorporate additional memory types and exclude methods whose memory space is fixed by design. Since Video-MMMU provides no evidence-interval annotations, we report only end-to-end accuracy. As shown in Table 24, CARVE achieves the best accuracy with all three generators. Notably, CARVE consistently outperforms VideoRAG-A despite using only the best configuration for each retrieved chunk, while VideoRAG-A provides the generator with all available memory representations. This suggests that selecting the right representation for each chunk is more effective than simply providing the generator with more information.

Computational overhead. Table 25 reports how the candidate pool and online latency scale as configurations are added under the same setting. The number of raw candidates grows linearly from 3 to 18, whereas the number of unique candidates grows only $2.25\times$ from 3.00 to 6.75. This gap arises because different modalities and granularities often retrieve the same chunk, which raises the duplication rate to 62.5% and reduces the marginal gain from 1.26 to 0.51 unique chunks per added configuration. Consequently, total online latency increases by only $2.08\times$, from 254.7 ms to 528.9 ms. The additional memory overhead is also negligible, as incorporating ASR and OCR increases the total memory footprint by only 0.033%.

Table 25: Candidate pool growth and online latency as the configuration space expands on Video-MMMU.

#Config	Configurations	Raw cand.	Unique cand.	Dup. rate	Marginal gain	Latency (ms)
1	VC	3	3.00	0.0%	–	254.7
2	VC + VF	6	4.26	29.0%	1.26	346.0
4	VC + VF + TC + TF	12	5.72	52.3%	0.73	416.3
6	4 Configs + ASR + OCR	18	6.75	62.5%	0.51	528.9

D.9 Compatibility with Agentic VideoRAG

Table 26: Compatibility of CARVE with agentic VideoRAG frameworks.

Method	nDCG@5	Pass Rate
WorldMM	0.292	0.133
WorldMM + CARVE	**0.427**	**0.360**
VideoAgent	0.210	0.193
VideoAgent + CARVE	**0.281**	**0.273**

Table 27: Comparison with EgoRAG. Best values in bold.

	Retrieval		Generation		
Method	R@5	nDCG@5	Qwen3-VL-8B	Qwen3-VL-32B	Gemma-4-26B
EgoRAG	0.283	0.191	0.193	0.203	0.137
CARVE	**0.603**	**0.433**	**0.352**	**0.368**	**0.318**

Recent VideoRAG research has moved toward agentic frameworks that search for evidence through iterative reasoning or structured memory. To examine whether CARVE is complementary to these frameworks, we plug CARVE into agentic methods as their retrieval module. We additionally compare CARVE with a framework whose retrieval depends on its memory structure.

Integration into agentic frameworks. Agentic video retrieval frameworks use multi-step reasoning to determine what information to retrieve and whether additional retrieval is needed. However, within each retrieval step, they typically represent all video chunks using a fixed configuration. CARVE complements this process by selecting the most suitable representation for each chunk, without modifying the agent's planning. We therefore integrate CARVE into two representative frameworks, WorldMM (Yeo et al., 2026b) and VideoAgent (Wang et al., 2024b), replacing only their retrieval modules while keeping all other components unchanged. As shown in Table 26, this simple integration improves both retrieval and generation performance for both frameworks. The pass rate increases from 0.133 to 0.360 for WorldMM and from 0.193 to 0.273 for VideoAgent. These results show that chunk-level representation selection remains effective in agentic retrieval settings and can be directly incorporated into existing frameworks.

Comparison with hierarchical memory retrieval. EgoRAG (Yang et al., 2025) organizes long videos into a temporal hierarchy of memories and retrieves evidence by progressively narrowing the search from coarse to fine temporal scales. Each query thus follows a single coarse-to-fine search path before the final clips are retrieved. In contrast, CARVE operates at the chunk level and selects the most suitable representation for each chunk independently. Because EgoRAG's retrieval is tightly coupled with its hierarchical memory structure, we compare against the full EgoRAG pipeline rather than replacing its retrieval module with CARVE. As shown in Table 27, CARVE improves R@5 from 0.283 to 0.603 and increase the pass rate across all three generators. In EgoRAG, early retrieval errors at coarse levels can propagate to later stages, whereas CARVE scores the original chunks directly without sequentially restricting the search space.

E PROMPTS

E.1 QA GENERATION PROMPT

Figure 5 presents the prompt used to generate visually grounded QA pairs from egocentric video segments, with each question anchored to a unique observable event in the frames.

E.2 QA FILTERING PROMPT

Figure 6 shows the prompt used to verify whether each generated QA pair is answerable from the provided frames and whether the answer is correct. In addition, Figure 7 presents the prompt used to check whether a question can be answered without visual input, using only common sense and the question itself.

E.3 LLM JUDGE PROMPT

Figure 8 presents the prompt used by the LLM judge to evaluate whether a candidate answer matches the key factual content of the reference answer.

E.4 MEMORY GENERATION PROMPT

Figure 9 presents the prompt used to generate text-clip memories that summarize what the camera wearer sees, does, and experiences over an egocentric video clip. In addition, Figure 10 presents the prompt used to generate text-keyframe memories that describe the camera wearer's visual experience at a specific moment in detail.

E.5 QA ANSWERING PROMPT

Figure 11 presents the prompt used to answer questions from retrieved textual descriptions, including either keyframe or clip memory contexts. In addition, Figure 12 presents the prompt used to answer questions from combined textual memories that include both clip-level descriptions and keyframe-level descriptions. Furthermore, Figure 13 shows the prompt used to answer questions from retrieved video segments, while Figure 14 shows the prompt used to answer questions from retrieved keyframe images. Finally, Figure 15 presents the prompt used to answer questions from combined visual evidence, including both video segments and keyframe images.

You will receive a sequence of frames from a segment of an egocentric video.

Objective
Create up to **4** QA pairs, grounded strictly in visible evidence from the frames.

- Generate fewer than 4 if the segment does not naturally support all categories.
- Each question must localize a single, unique moment in the segment.
- Do not use prior knowledge or common sense.

Category Rule
- Attempt to create at least one QA pair per category from the list below.
- However, ONLY generate a QA pair for a category if there is clear, strong visual evidence in the frames.
- If a category is not clearly supported by the visible content, DO NOT force a question for that category.
{category_block}

Step-by-step Procedure (DO NOT reveal these steps in the output)
For each of the categories, do:

Step 1) Select ONE visually unique event boundary (anchor).
- The anchor must include:
    (1) a concrete object
        - Include color, material, location, or distinctive visual detail.
        - Avoid generic words
        - Avoid camera motion, viewpoint shift, reframing, gaze shift, zoom, panning, tilting, or scene transition as the anchor event.
        - Include specific identifying details such as color, material, relative location, shape, size, or any other distinctive visual attribute.
    (2) a clearly observable physical action
    (3) a resulting visible outcome or state changes

Step 2) Draft a question using the anchor:
- e.g., "After/When <ANCHOR EVENT>, <QUESTION>?" or any natural phrasing that clearly references the anchor event.
- The anchor provides temporal grounding.
- The question and answer must belong to the category.

Step 3) Ensure the question asks about a DIFFERENT visual detail than the anchor.
- The anchor must NOT contain the answer.
- If the answer could be inferred from the anchor alone, rewrite.

Step 4) Write the reasoning (justification).
- Describe the specific visual evidence observed in the frames that is relevant to the question.
- State what objects, actions, or state changes are visible and where they appear.

Step 5) Based on the reasoning above, write the answer as exactly ONE concrete visual fact.
- The answer must be derived from the evidence described in the reasoning.

Constraints
- Each question anchors to a unique event.
- Avoid anchors based on generic or common actions (e.g., holding or looking at a phone).
- Avoid mechanical or formulaic phrasing.
- Each answer uses only observable visual evidence.
- Each answer states only a single fact; not multiple actions.
- No timestamps or frame numbers.
- Maintain an egocentric viewpoint; use "I" only for the wearer's visible actions.
- Do NOT base questions on fine-grained phone screen details.
- Do NOT ask about frequency, repetition, or habitual behavior.

After producing the QA pairs, add a 1-2 line validation confirming all constraints are met. Revise output if needed.

Output
Produce 1 to 4 QA pairs, all grounded exclusively in the provided frames.

Figure 5: Prompt for QA Generation.

Evaluate each QA pair below.
For each, determine if the question can be answered from these frames and if the answer is correct.

{qa_block}

Figure 6: Prompt for GPT Answerability Filtering.

You have NO frames or visual input.
Try to answer using ONLY common sense and the question.

Question:
{question}

Figure 7: Prompt for GPT Blind Check Filtering.

You are a strict evaluator judging whether a candidate answer to a question about video content is correct, using a reference answer as the ground truth.

Question:
{question}
Reference Answer:
{reference}
Candidate Answer:
{candidate}

Evaluation criteria:

- YES: The candidate answer captures the key factual content of the reference answer. It does not need to match word-for-word, but must convey the core information. Some additional details or differences in phrasing are acceptable.
- NO: The candidate answer is factually wrong, contradicts the reference, misses the central point, is vague or generic where the reference is specific, or admits it cannot answer the question when the reference provides a clear answer.

Important:
Focus on whether the essential facts match, not surface-level wording.
A shorter answer that gets the key facts right is YES.
A longer answer that misses the main point is NO.

Answer with ONLY "YES" or "NO" followed by a one-sentence reason.

Format: [YES/NO]: [REASON]

Figure 8: Prompt for LLM Judge.

You are analyzing an egocentric (first-person) video clip from someone's day.
Describe what the person is seeing, doing, and experiencing during this clip.
Use first-person perspective ("I am...", "I see...") and be specific about:

- Activities and actions being performed
- Objects being handled or interacted with
- Environment and location details
- Any notable events or transitions
- People present and interactions with them
- Time of day indicators if visible

Provide a natural, coherent description that captures the essence of what happened during this time period.

Figure 9: Prompt for Text Clip Memory Generation.

```
You are analyzing an egocentric (first-person) image from someone's day.
Describe in detail what I am seeing and doing at this exact moment.
Use "I" perspective and be extremely specific and detailed. Focus on:

- What I am currently looking at (center of my visual field)
- What my hands are doing (if visible)
- Objects I am interacting with or holding
- The environment and location details
- Any text, labels, or screens visible
- Other people present and their actions

Provide a detailed, vivid description of this exact moment.
```

Figure 10: Prompt for Text Key Frame Memory Generation.

```
You are a helpful assistant answering questions about video content.
The following segments are language descriptions of a first-person view video, arranged in
chronological order with timestamps.

Retrieved context:
{context}
Question:
{question}

Based on the descriptions above, provide a clear and concise answer.
If there is not enough information to answer, say so.
```

Figure 11: Prompt for Text Key Frame and Text Clip Answering.

```
You are a helpful assistant answering questions about video content.
The following segments are language descriptions of a first-person view video, arranged in
chronological order with timestamps.
Each segment includes two levels of description:

- [Coarse Description]: A high-level textual summary of the entire segment.
- [Fine Descriptions]: Detailed textual descriptions of individual keyframe moments within the
segment, capturing specific visual details.

Retrieved context:
{context}
Question:
{question}

Provide a clear, concise answer based on the retrieved context above.
Use both coarse and fine descriptions to form a comprehensive answer.
If the context doesn't contain enough information to answer the question, say so.
```

Figure 12: Prompt for Text Combined Answering.

```
You are a helpful assistant answering questions about video content.
The video segments above are from a first-person view video, arranged in chronological order.

Question:
{question}

Based on the video content above, provide a clear and concise answer.
If there is not enough information to answer, say so.
```

Figure 13: Prompt for Visual Clip Answering.

You are a helpful assistant answering questions about video content.
The keyframe images above are from a first-person view video, arranged in chronological order.

Question:
{question}

Based on the images above, provide a clear and concise answer.
If there is not enough information to answer, say so.

Figure 14: Prompt for Visual Key Frame Answering.

You are a helpful assistant answering questions about video content.
The video segments and keyframe images above are from a first-person view video, arranged in chronological order.

Question:
{question}

Based on all the visual content above, provide a clear and concise answer.
If there is not enough information to answer, say so.

Figure 15: Prompt for Visual Combined Answering.