OMNIPLEX-RL: CONTEXT COMPACTION FOR NATIVE OMNI-MODAL AGENTIC REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Omni-modal agents answer questions from text corpora, knowledge bases, long documents and audio-video streams, reading evidence natively as pixels, pages, frames and sound. Prior work advances what such agents look at, via active perception trained with reinforcement learning (RL), and what they keep, via summaries, folding and textual memory. Yet native looks exhaust the window before the call budget, compaction turns raw evidence into text that cannot be re-opened, and folding leaves no consistent sample or group for credit. To address these challenges, we propose **OmniPlex-RL**, a native omni-modal agent that compacts its context into segments linked by an omni-modal hypergraph memory whose units keep the call re-opening their raw media. It trains each segment as a sample under single-rollout RL, with a per-modality moving batch baseline. On twelve text, multimodal and omni-modal benchmarks, OmniPlex-RL leads on all twelve, 4.5 F1 above the strongest baseline. Our software and data are publicly available[1].

1 INTRODUCTION

Omni-modal agents answer questions whose evidence lies in a text corpus, a picture-rich knowledge base, a hundred-page document or a long audio-video stream (Li et al., 2026f; Zhang et al., 2026a). Within a call budget the agent searches, reads pages and watches, and takes in what returns – pixels, rendered pages, frames and sound – natively rather than as a transcript (Figure 1). This keeps the perceptual detail of the evidence, but every look is paid in context: 2.5K tokens for a knowledge-base entry with its picture, 1.9K for three rendered pages and 2.0K for a 180 s audio-video window, while a whole document runs to 50K–130K.

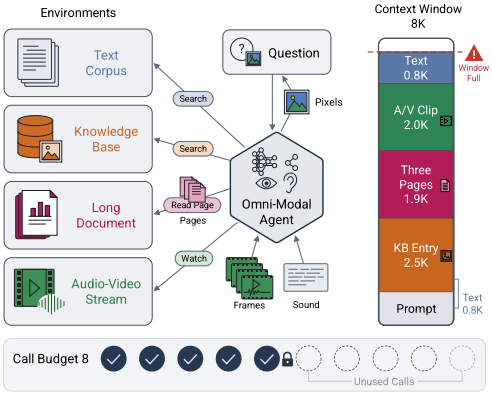


Figure 1: A native omni-modal agent pays for every look it takes in context.

Prior work advances *what to look at* and *what to keep*. Active-perception agents plan calls (Tao et al., 2026a), retrieve per round (Zhu et al., 2026), zoom into intervals (Chen et al., 2026) or coordinate sub-agents (Zhang et al., 2026b), trained with GRPO-style reinforcement learning (RL) (Shao et al., 2024; Xing et al., 2026); context management compresses trajectories by summaries and folds (Packer et al., 2024; Kang et al., 2026b; Sun et al., 2025; Ye et al., 2025), learns memory with RL (Zhou et al., 2026a; Zhang et al., 2026c), or store multimodal experience as scripts and text (Nie et al., 2026; Long et al., 2026).

However, as shown in Figure 2, three challenges remain. **(i) Saturation before the budget.** Beyond text, an 8K window holds four or five full-size looks, fewer than the eight calls allowed, so agents overflow or transcribe and lose grounding. **(ii) Evidence lost to compaction.** Existing carriers are text, scripts or time spans; a frame turned into a sentence is cut from its source (Zerhoudi et al., 2026) and cannot be re-opened. **(iii) Credit across folds.** Segments share no context, so a concatenated sample trains tokens on unseen prefixes (Zhang et al., 2026c), and groups of media-heavy rollouts are costly yet silent when they score alike, as in 68.4% of OmniAgent-RL's steps.

[1] Anonymous Github Link: `https://anonymous.4open.science/r/OmniPlex-RL`

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.

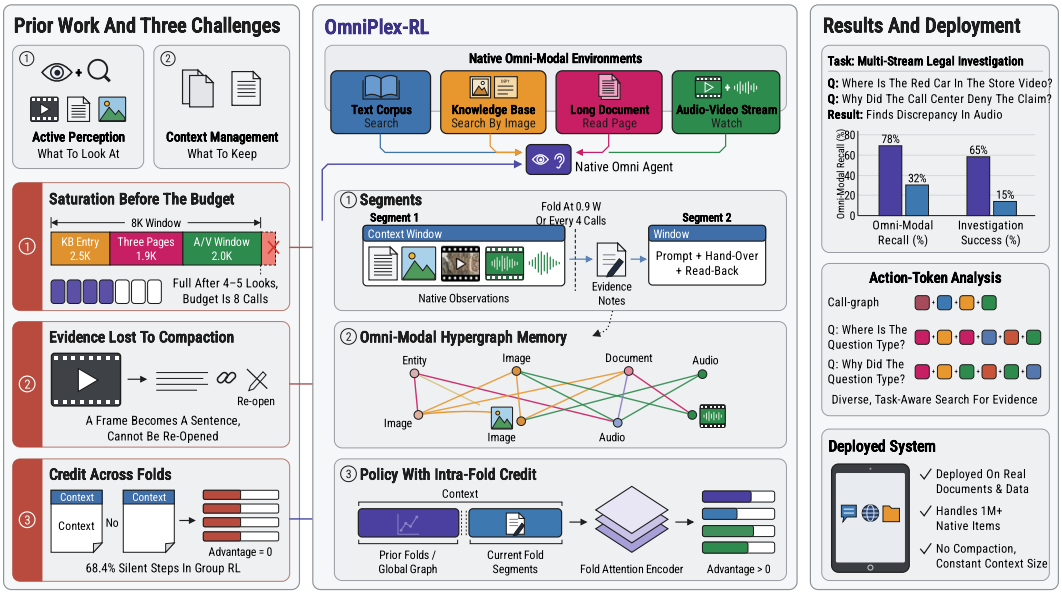


Figure 2: Overview of OmniPlex-RL: prior work and its three challenges (left), the segments, hypergraph memory and single-rollout RL that address them (middle), and key results (right).

To address these challenges, we propose **OmniPlex-RL**, a native omni-modal agent trained by single-rollout RL. For (i), a rollout is cut into **segments** that close near the window or every four calls, after the policy re-reads each media observation into an **evidence note**. For (ii), each observation is a unit of an **omni-modal hypergraph memory**, an n-ary hyperedge over its entities, images and audio addressed by the call that produced it, so a successor reads back relevant units and can re-open their raw media. For (iii), each segment trains on its own context under a **per-modality moving batch baseline**, with an **interaction floor** against answering blind, without looking first.

We evaluate on twelve benchmarks over text, knowledge bases, documents and audio-video streams, five held out, all methods sharing Qwen2.5-Omni-3B (Xu et al., 2025), data, an 8K window and eight calls. OmniPlex-RL leads on all twelve with 27.0 average F1, 4.5 above the best baseline and 5.7 above the best trained omni agent, as much on held-out suites; its lead is widest where it folds (+6.1), where its memory triples how often gold answers reach the successor, at 2.8 calls a question.

2 Related Work

We relate OmniPlex-RL to omni-modal agents and agent context management (see Appendix G).

Omni-modal agents. Agents on native backbones (Xu et al., 2025; Team, 2026) have moved from answering to active perception. OmniAgent-AG (Tao et al., 2026a) lets a text planner decide where to listen and look, OmniRAG-Agent (Zhu et al., 2026) and OmniReasoner (Chen et al., 2026) make retrieval and zooming native actions, and OmniAgent-RL (Xing et al., 2026) and Orchestra-o1 (Zhang et al., 2026b) learn such decisions with RL. Light-Omni (Nie et al., 2026) and M3-Agent (Long et al., 2026) add persistent memory, but as text or latent state that cannot re-open raw evidence after compaction, so a compacted agent can no longer go back and check what it saw.

Agent context management. Text agents bound context with memory hierarchies (Packer et al., 2024), history compression (Kang et al., 2026b) and trajectory folding (Sun et al., 2025; Ye et al., 2025). As compaction can drop what later steps need (Zerhoudi et al., 2026), recent work keeps content addressable (Wang et al., 2026d) or retains visual evidence (Su et al., 2026), and MEM1 (Zhou et al., 2026a) and MemAct (Zhang et al., 2026c) learn it with RL. OmniPlex-RL brings both to omni-modal agents: compacted evidence stays re-openable in its raw modality, and each segment trains on its own context, with only one rollout per prompt and a per-modality moving batch baseline.

Table 1: Comparison of the agents in Table 2. OmniAgent-AG/Orchestra-o1: text surrogates (Σ^*: text); OmniRAG-Agent/OmniReasoner: raw observations, never compacted; OmniAgent-RL/Light-Omni: textual or latent memory under group or raw-reward credit; OmniPlex-RL (Ours): an addressed hypergraph that re-opens raw media, trained on each segment with a per-modality baseline.

Method	Context Carrier	Policy Optimisation
OmniAgent-AG (Tao et al., 2026a)	$\tilde{o}_t = \pi_0(\cdot \mid o_t, \mathrm{mode}_t) \in \Sigma^*$ Self-calls turn each look into text.	$\theta = \theta_0$ Training-free, one call per step.
Orchestra-o1 (Zhang et al., 2026b)	$\mathbf{s}_{t+1} = \mathbf{s}_t \oplus \mathrm{msg}(o_t) \in \Sigma^*$ Reads messages of fresh sub-agents.	$\hat{A}_u = (R(u) - \bar{R}_{\mathrm{grp}})/\sigma_{\mathrm{grp}}$ DA-GRPO on a rule-based rubric.
OmniRAG-Agent (Zhu et al., 2026)	$\mathbf{s}_{t+1} = \mathbf{s}_t \oplus g_t \oplus o_t$ Keeps every retrieved frame and page.	$\hat{A}_u = (R(u) - \bar{R})/\sqrt{V}$ Batch baseline, one call per round.
OmniReasoner (Chen et al., 2026)	$\mathbf{s} = \mathbf{p}(x) \oplus g \oplus o_{\mathrm{zoom}}$ A preview plus one time-anchored zoom.	$\hat{A}_u = (R(u) - \bar{R})/\sqrt{V}$ Answer or zoom once, batch baseline.
OmniAgent-RL (Xing et al., 2026)	$m \leftarrow \pi_\theta(\cdot \mid m, o_t) \in \Sigma^*$ Rewrites a textual memory as it looks.	$\hat{A}_{u,t} = \hat{A}_u \, \bar{H}_{u,t}/\bar{H}$ TAURA: entropy-scaled group advantage.
Light-Omni (Nie et al., 2026)	$(\varsigma, \boldsymbol{\xi}) \leftarrow \mathrm{consolidate}(\varsigma, \boldsymbol{\xi}, o_t)$ An episodic script and a latent state.	$\hat{A}_u = R(u)$ Reward-weighted tuning, one rollout.
OmniPlex-RL (Ours)	$\mathcal{G}_k = (\mathcal{V}_k, \mathcal{F}_k),\ \mathrm{addr}(f_t) = a_t$ Hypergraph units re-open raw media.	$\hat{A}_u = \mathrm{clip}\left((R(u) - \mu_d)/(\sigma_d + \delta)\right)$ Per-segment, per-modality baseline.

3 METHODOLOGY: OMNIPLEX-RL

As Table 1 traces, prior omni-modal agents carry evidence across contexts as text, raw observations or latent state. OmniPlex-RL (Figure 3) pairs a native agent (§3.1) with segments and an omni-modal hypergraph memory (§3.2), trained by single-rollout RL (§3.3), answering challenges (i)–(iii) of §1.

3.1 NATIVE AGENT AND SATURATION BEFORE THE BUDGET

We first formalise how the agent interacts with its environment and then measure what one native look costs, which shows that a single context saturates long before the call budget is fully spent.

Interaction loop. A question x with gold answers $\mathcal{Y}^\star$ is posed against an environment $\mathcal{E}$ with tools $\mathcal{T}$: `search` returns five passages or knowledge-base entries with pictures (`search_by_image` queries with the question's photograph), `read_page` up to three rendered pages, and `watch` four frames and up to 30 s of sound from at most 180 s of the stream. The policy π_θ with parameters θ keeps one context $\mathbf{s}_t$ at turn t: it emits a response g_t, whose first tool call a_t is executed, and the observation o_t is appended exactly as returned,

$$g_t \sim \pi_\theta(\cdot \mid \mathbf{s}_t), \quad a_t = \mathrm{call}(g_t) \in \mathcal{T}, \quad o_t = \mathcal{E}(a_t), \quad \mathbf{s}_{t+1} = \mathbf{s}_t \oplus g_t \oplus o_t, \quad \mathbf{s}_1 = \mathbf{p}(x), \quad (1)$$

where $\oplus$ is concatenation and the prompt $\mathbf{p}(x)$ holds the instructions, the tools, the question and its media. A response has at most $L{=}320$ tokens and an episode at most $T_{\max}{=}16$ turns and $K{=}8$ calls; `finish` submits the answer $\hat{y}$. A reply without a call is reminded once and a second one is read as the answer, an option letter answers a choice question at once, an episode ending without `finish` is asked once for its answer, and the question's photograph is first described natively beside $\mathbf{p}(x)$; Appendix D.2 gives the scoring rule and evaluation protocol of each of the twelve suites.

Cost of a native look. Observations enter as pixels, pages, frames and sound rather than transcripts, so a look is paid in the tokens it adds. With $n_{\mathrm{txt}}(\mathbf{s})$ the text tokens of a context $\mathbf{s}$ and $\tau(\omega)$ the tokens of a medium ω in $\mathbf{s}$ (one per $28{\times}28$-pixel patch, 25 per second of audio), the length of a context, the cost of a call and the full-size looks one context can hold are

$$\ell(\mathbf{s}) = n_{\mathrm{txt}}(\mathbf{s}) + \sum_{\omega \in \mathbf{s}} \tau(\omega), \qquad c(a_t) = \ell(\mathbf{s}_{t+1}) - \ell(\mathbf{s}_t), \qquad n_{\mathrm{fill}}(x) = \left\lfloor \frac{W - \ell(\mathbf{p}(x))}{\underline{c}_{\mathcal{E}}} \right\rfloor, \quad (2)$$

where $W{=}8192$ is the window, prompt included, and $\underline{c}_{\mathcal{E}}$ the least cost $c(a_t)$ of a full-size look, one at the per-call limits of $\mathcal{E}$. A knowledge-base search costs 2.5K tokens and a three-page read 1.9K (Table 4), so $n_{\mathrm{fill}} \le 5 < K$ beyond text, while a text search of 0.8K fits all K calls in one window.

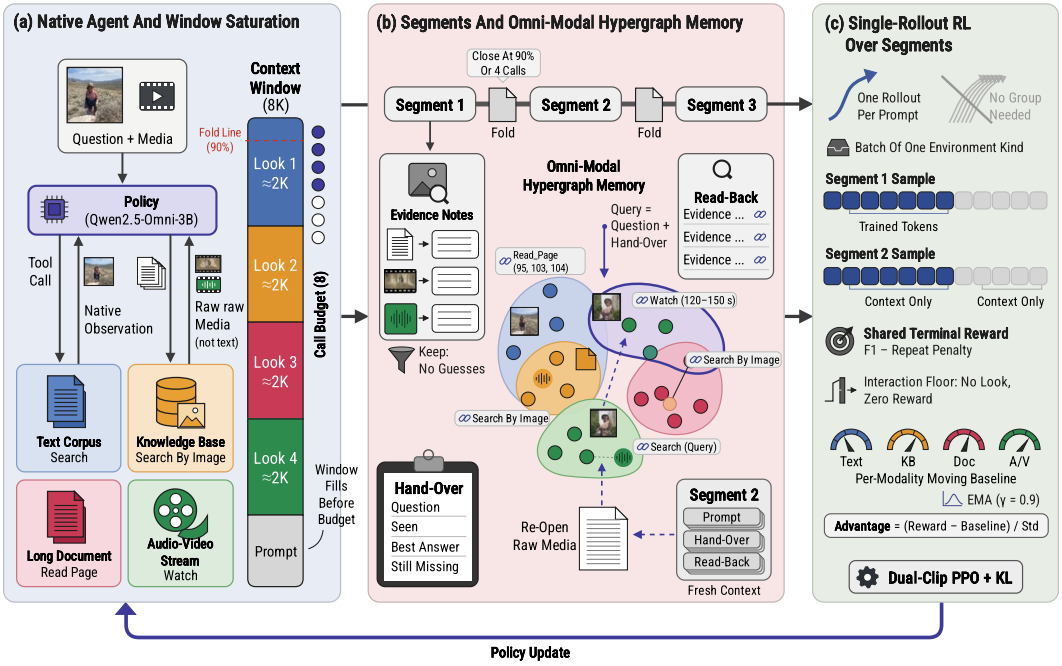


Figure 3: Framework of OmniPlex-RL: a native agent looks until its window saturates (§3.1), closes segments into an omni-modal hypergraph memory (§3.2), and learns via single-rollout RL (§3.3).

Proposition 1. *Beyond text, a few full-size native looks exhaust the window before the budget.*

Proof. We provide experimental results in Sections 4.4 and proofs in Appendix B.1. □

3.2 Segments and Omni-Modal Hypergraph Memory

A saturated context must be closed and the episode continued afresh. As a summary loses its grounding once a frame becomes a sentence (Zerhoudi et al., 2026), we fix when a context closes, what it writes to memory and what the next context reads back from the memory to continue the episode.

Segments. A *segment* is the part of a rollout that owns one context. With N_t the calls made before turn t, segment k, opened at turn t_k, closes once its window nears saturation or it has spent C calls:

$$t_{k+1} = \min\Big\{\, t > t_k :\ \ell(\mathbf{s}_t) > \eta W \ \ \vee \ \ \big(N_t - N_{t_k} \geq C\big) \wedge \big(t - t_k \geq 2\big) \wedge \big(N_t < K\big) \Big\}, \quad (3)$$

with η=0.9, leaving room for the hand-over, and C=4; inside a segment, Eq. (1) holds turn by turn.

Evidence notes. Before a segment closes, the policy re-reads each of its media observations natively and writes an *evidence note*. With $\mathcal{O}_k$ the turns of segment k whose observation carries media and $\bar{x}$ the question stem without its options, the note on turn $t \in \mathcal{O}_k$ is

$$\mathcal{O}_k = \big\{t :\ t_k \leq t < t_{k+1},\ o_t \text{ carries media}\big\}, \qquad \tilde{z}_t \sim \pi_\theta(\cdot \mid o_t, \bar{x}), \qquad z_t = \mathrm{keep}(\tilde{z}_t), \quad (4)$$

where keep drops sentences that state a verdict or restate an option, so a note records what is shown rather than a guess; a segment writes at most six notes, which become the text of its memory units.

Omni-modal hypergraph memory. Each observation becomes a *unit* of the segment's hypergraph $\mathcal{G}_k = (\mathcal{V}_k, \mathcal{F}_k)$ with nodes $\mathcal{V}_k$ and hyperedges $\mathcal{F}_k$: a text z_t (the note, or a passage of a text search), the images and audio $\mathcal{M}_t$ of o_t as media nodes, and an *address*, the call a_t that produced it. A local extractor Ext lifts the entities of z_t, and the unit becomes one n-ary hyperedge

$$f_t = \mathrm{Ext}(z_t) \cup \mathcal{M}_t \in \mathcal{F}_k, \qquad \mathrm{addr}(f_t) = a_t, \qquad \mathcal{V}_k = \bigcup\nolimits_{f \in \mathcal{F}_k} f, \quad (5)$$

so a picture, its caption and a spoken line stay one fact. As $\mathcal{E}$ is deterministic, re-issuing an address returns the same pages, frames or clip, so the raw evidence of every unit remains only one call away.

Hand-over and read-back. A segment ends with the policy's *hand-over* h_k (what was seen and where, a best answer, what is missing), into which the loop writes the verbatim question. The successor opens from the prompt, h_k and a read-back $\mathbf{b}_k$ of the memory: with a unit-norm text encoder ϕ and query $\mathbf{q}_k = \phi(x \oplus h_k)$, a hyperedge f of an earlier hypergraph $\mathcal{G}_j$, $j \le k$, scores

$$\mathbf{s}_{t_{k+1}} = \mathbf{p}(x) \oplus h_k \oplus \mathbf{b}_k, \qquad \mathrm{sc}(f) = \nu \left\langle \phi(f), \mathbf{q}_k \right\rangle + (1-\nu) \operatorname*{mean}_{v \in f \cap \mathcal{Q}_j} \left\langle \phi(v), \mathbf{q}_k \right\rangle, \tag{6}$$

where $\mathcal{Q}_j \subseteq \mathcal{V}_j$ holds the κ_v entities nearest to $\mathbf{q}_k$, candidates are the κ_f hyperedges nearest to $\mathbf{q}_k$ and those incident to $\mathcal{Q}_j$, and ν weighs a hyperedge against its entities (the first term alone if $f \cap \mathcal{Q}_j = \emptyset$). $\mathbf{b}_k$ renders the κ_b best units as evidence lines ending in their addresses, which the successor cites directly or re-opens to recover the raw pages, frames or audio clip behind that line.

Proposition 2. *Addressed memory lets a compacted agent re-open raw evidence a summary drops.*

Proof. We provide experimental results in Sections 4.5 and proofs in Appendix B.2. □

3.3 Single-rollout RL over segments

Segments share no context, so a rollout trained as one concatenated sample would condition tokens on prefixes they were never generated under (Zhang et al., 2026c). We train every segment on its own context and replace the group baseline of GRPO (Shao et al., 2024) by a moving batch baseline.

Segment samples and reward. Each training prompt x_u yields one rollout u with segments $k = 1, \dots, S_u$. Segment k is one sample: its tokens $\mathbf{w}^{u,k}$ are the segment's final context, and its loss positions $\mathcal{I}_{u,k}$ are those of its own responses g_t, $t_k \le t < t_{k+1}$, so each trained token sees the prefix it was generated under. All segments of u share the terminal reward

$$R(u) = \mathbf{1}\big[N_{\mathcal{T}}(u) > 0\big] \cdot \Big(\mathrm{F1}\big(\hat{y}_u, \mathcal{Y}^\star_u\big) - \min\big(\lambda\, N_{\mathrm{rep}}(u),\ \bar{\lambda}\big)\Big), \tag{7}$$

where F1 is the token-level F1 of the answer $\hat{y}_u$ against its best-matching gold answer, the evaluation score of §4.1, $N_{\mathcal{T}}(u)$ counts the tool calls of u, and $N_{\mathrm{rep}}(u)$ its refused repeats (λ=0.02, $\bar{\lambda}$=0.1). The indicator is the *interaction floor*: a rollout that answers without looking scores zero yet stays in the update, and in training a first reply that only names an option is returned once, asking for a look.

Per-modality moving batch baseline. One rollout per prompt leaves no group, so the batch sets the baseline. Each batch holds one environment kind d (text, knowledge base, document or stream); with $\bar{R}$ and V the mean and variance of $R(u)$ over its rollouts, moving averages with decay γ=0.9 give every segment of u the advantage

$$\mu_d \leftarrow \gamma\, \mu_d + (1-\gamma)\, \bar{R}, \quad \sigma_d^2 \leftarrow \gamma\, \sigma_d^2 + (1-\gamma)\, V, \quad \hat{A}_u = \mathrm{clip}\Big(\frac{R(u) - \mu_d}{\sigma_d + \delta},\ -\bar{A},\ \bar{A}\Big), \tag{8}$$

with σ_d floored at 10^{-2}, δ=10^{-6}, bound $\bar{A}$=5, and (μ_d, σ_d^2) set by the first batch of kind d, so a batch that scores alike still moves the policy relative to the history of its own environment kind, and one baseline per kind keeps an easy text batch from setting the bar for a much harder batch of long audio-video streams, whose rewards lie on a different scale and would bias all of its advantages.

Objective. Each trained token $w_i = w_i^{u,k}$, $i \in \mathcal{I}_{u,k}$, takes $\hat{A}_u$ in a dual-clip PPO surrogate (Schulman et al., 2017) with a KL penalty to the untrained backbone $\pi_{\mathrm{ref}} = \pi_{\theta_0}$, averaged over the $n_{\mathcal{B}}$ trained tokens of a mini-batch $\mathcal{B}$ of segments:

$$\begin{aligned} \mathcal{J}(\theta) &= \frac{1}{n_{\mathcal{B}}} \sum_{(u,k)\in\mathcal{B}} \sum_{i \in \mathcal{I}_{u,k}} \Big[\psi_i - \beta\big(\exp(\Delta_i) - \Delta_i - 1\big)\Big], \qquad n_{\mathcal{B}} = \sum_{(u,k)\in\mathcal{B}} |\mathcal{I}_{u,k}|, \\ \psi_i &= \max\Big(\min\big(\rho_i \hat{A}_u,\ \mathrm{clip}(\rho_i, 1{-}\epsilon, 1{+}\epsilon)\, \hat{A}_u\big),\ \zeta \min(\hat{A}_u, 0)\Big), \end{aligned} \tag{9}$$

where ρ_i is the probability ratio of π_θ to the rollout policy $\pi_{\theta_{\mathrm{old}}}$ and Δ_i the log-ratio of π_{ref} to π_θ, both on w_i given $\mathbf{w}^{u,k}_{<i}$, with clip range ϵ=0.2, dual-clip bound ζ=3 and the KL weight β=0.01.

Proposition 3. *A moving batch baseline yields advantages even from one rollout per prompt.*

Proof. We provide experimental results in Sections 4.6 and proofs in Appendix B.3. □

Table 2: Main results: token F1, mean ± half-range over three evaluation runs, one backbone (Qwen2.5-Omni-3B) trained once on one training set; †: held out. Under each name: memory carrier and whether it compacts (✓/✗). Underlined: best baseline; green: gain over it; ‡: see §4.2.

	Base	**Training-free**		**Trained omni agents**					
Benchmark	Qwen2.5-Omni-3B no memory ✗	OmniAgent-AG raw video ✗	**OmniPlex** native replay ✓	OmniRAG-Agent frames + audio ✗	Orchestra-o1‡ sub-agent messages ✗	OmniAgent-RL textual memory ✓	Light-Omni script + latent ✓	Omni-Reasoner time spans ✗	**OmniPlex-RL** **native replay** ✓
BC-Plus	0.9 ±0.0	5.3 ±0.3	6.1 ±1.7	5.6 ±0.9	4.1 ±0.7	4.8 ±1.1	4.0 ±0.8	4.1 ±0.7	**7.1** ±1.1 (+1.00)
HotpotQA	6.7 ±0.0	26.6 ±0.2	27.7 ±4.1	20.2 ±1.3	24.2 ±3.1	27.1 ±3.2	27.2 ±0.2	18.7 ±0.8	**32.5** ±1.6 (+4.77)
MuSiQue†	1.1 ±0.0	7.8 ±0.2	8.5 ±0.6	4.8 ±0.4	3.0 ±0.4	5.4 ±0.8	5.7 ±1.1	5.7 ±0.5	**10.4** ±0.8 (+1.90)
E-VQA	3.7 ±0.2	10.1 ±0.4	2.5 ±1.4	5.3 ±1.2	2.8 ±0.4	9.9 ±1.2	1.5 ±1.1	13.3 ±3.5	**15.0** ±2.9 (+1.73)
InfoSeek†	0.5 ±0.0	17.8 ±0.1	14.5 ±3.4	10.5 ±0.4	18.7 ±1.4	24.1 ±1.0	12.4 ±0.7	19.3 ±2.6	**25.6** ±0.3 (+1.53)
LongDocURL	5.7 ±0.4	14.3 ±0.5	25.1 ±2.9	23.5 ±0.7	23.5 ±4.7	12.0 ±1.9	22.8 ±1.0	12.9 ±1.8	**25.9** ±0.7 (+0.80)
MMLB-Doc†	4.4 ±0.1	9.5 ±1.1	12.8 ±1.8	9.9 ±1.0	12.7 ±1.7	6.5 ±1.2	11.5 ±1.5	6.5 ±0.7	**15.4** ±1.3 (+2.60)
OVBench	25.3 ±0.8	26.5 ±2.0	27.3 ±2.3	27.4 ±1.9	28.1 ±1.2	28.1 ±0.8	24.8 ±2.0	18.9 ±2.1	**30.2** ±1.9 (+2.10)
Daily-Omni	35.2 ±0.0	47.7 ±3.9	52.3 ±1.2	55.5 ±1.6	52.1 ±2.3	50.3 ±2.3	51.6 ±4.2	47.2 ±3.2	**57.3** ±3.1 (+1.80)
FineVideo	2.7 ±0.0	21.6 ±0.8	22.6 ±2.2	26.3 ±0.6	16.7 ±1.5	21.3 ±1.6	10.1 ±1.3	22.1 ±1.1	**29.6** ±0.4 (+3.27)
LVOBench†	27.1 ±0.3	17.5 ±2.7	28.4 ±5.0	28.1 ±0.8	28.4 ±3.9	23.2 ±0.3	26.3 ±1.2	17.4 ±2.2	**31.5** ±2.8 (+3.13)
WorldSense†	24.2 ±0.0	25.5 ±2.0	42.2 ±0.8	38.5 ±2.3	39.8 ±3.2	40.1 ±3.1	38.5 ±1.1	30.3 ±1.6	**43.8** ±1.9 (+1.57)
Avg. Text	2.9 ±0.0	13.2 ±0.2	14.1 ±1.7	10.2 ±0.6	10.4 ±0.8	12.5 ±0.5	12.3 ±0.1	9.5 ±0.2	**16.7** ±0.3 (+2.56)
Avg. MM	3.6 ±0.1	12.9 ±0.1	13.7 ±1.3	12.3 ±0.6	14.4 ±1.0	13.1 ±0.4	12.0 ±0.5	13.0 ±2.0	**20.5** ±0.9 (+6.04)
Avg. Omni	22.9 ±0.2	27.8 ±0.2	34.6 ±0.8	35.2 ±0.3	33.0 ±1.7	32.6 ±0.8	30.3 ±0.8	27.2 ±1.4	**38.5** ±1.0 (+3.31)
Avg. All	11.5 ±0.1	19.2 ±0.1	22.5 ±0.7	21.3 ±0.3	21.2 ±1.1	21.1 ±0.2	19.7 ±0.5	18.0 ±0.3	**27.0** ±0.3 (+4.51)

4 EXPERIMENTS

We answer eight research questions. **RQ1:** Does OmniPlex-RL outperform existing omni-modal agents? **RQ2:** Does each of its components contribute? **RQ3–5:** Do native looks, the hypergraph memory and single-rollout RL behave as Propositions 1–3 state? **RQ6:** Are the gains robust? **RQ7:** What do they cost in interaction? **RQ8:** How does OmniPlex-RL answer where the baselines fail?

4.1 EXPERIMENTAL SETUP

Datasets and Evaluation Metrics. We evaluate on twelve benchmarks (Appendix D.1; †: held out of training): *Text* – BC-Plus (Chen et al., 2025), HotpotQA (Yang et al., 2018), MuSiQue† (Trivedi et al., 2022); *MM* – knowledge bases E-VQA (Mensink et al., 2023), InfoSeek† (Chen et al., 2023) and documents LongDocURL (Deng et al., 2025), MMLB-Doc† (Ma et al., 2024); *Omni* – audio-video streams OVBench (Li et al., 2026a), Daily-Omni (Zhou et al., 2026b), FineVideo (Farré et al., 2024), LVOBench† (Tao et al., 2026b), WorldSense† (Hong et al., 2026). The metric is token-level F1 (%) against the best-matching gold answer, which is also the reward's task term (Appendix D.2).

Baselines and Ablations. We compare the bare Qwen2.5-Omni-3B, the training-free OmniAgent-AG and OmniPlex (our loop, untrained), and five trained omni agents, OmniRAG-Agent, Orchestra-o1, OmniAgent-RL, Light-Omni and OmniReasoner, each ported and retrained on our data (Appendix E). Six ablations each remove one component of §3, three at inference and three in training.

Implementation Details. All columns share the Qwen2.5-Omni-3B thinker, W=8192, K=8, $T_{\max}$=16 and L=320 (§3.1); trained columns use LoRA (r=64, α=128) at learning rate 2×10^{-5} under one early-stopping rule, and OmniPlex-RL takes one rollout for each of 8 prompts per step, up to 80 steps, on four NVIDIA RTX 6000 Ada GPUs (Appendix E.4), evaluated as in Appendix D.2.

4.2 MAIN RESULTS (RQ1)

As shown in Table 2, OmniPlex-RL leads all twelve suites with 27.0 average F1, 4.5 above the strongest baseline and 5.7 above the strongest trained omni agent. **The margin is widest where a look is dearest:** +6.0 on MM, where one look costs 1.9K–2.5K tokens, against +2.6 on Text and +3.3 on Omni, and as large held out (+2.15) as in distribution (+2.21). **The native loop and its RL**

Table 3: Ablations: inference-time (the trained checkpoint with one component off) and training-time (a changed objective); rows and protocol as Table 2; brackets: difference from Full.

	Full	Inference-time: one component off			Training-time: a changed objective		
Benchmark	**OmniPlex-RL**	w/o compaction	w/o hyper-graph mem.	w/o coordinates	w/o segment credit	w/o batch baseline	w/o interact. floor
BC-Plus	**7.1** ±1.1	6.7 ±1.6 (−0.43)	5.8 ±0.0 (−1.33)	5.1 ±1.5 (−2.00)	5.2 ±1.0 (−1.87)	4.4 ±0.2 (−2.70)	1.7 ±0.1 (−5.37)
HotpotQA	**32.5** ±1.6	29.8 ±3.1 (−2.73)	29.8 ±1.4 (−2.73)	27.7 ±2.0 (−4.83)	16.9 ±3.3 (−15.63)	30.4 ±2.6 (−2.10)	7.3 ±0.1 (−25.17)
MuSiQue†	**10.4** ±0.8	7.9 ±0.8 (−2.43)	9.6 ±0.5 (−0.80)	7.3 ±0.2 (−3.10)	5.3 ±0.2 (−5.03)	4.2 ±0.7 (−6.20)	3.7 ±0.0 (−6.63)
E-VQA	**15.0** ±2.9	13.6 ±1.7 (−1.37)	14.0 ±2.5 (−1.03)	13.2 ±0.6 (−1.83)	13.6 ±1.4 (−1.43)	13.9 ±3.9 (−1.07)	15.4 ±0.4 (+0.43)
InfoSeek†	**25.6** ±0.3	23.6 ±2.4 (−2.07)	25.2 ±1.9 (−0.43)	25.1 ±3.2 (−0.57)	25.9 ±2.8 (+0.30)	17.1 ±0.6 (−8.57)	4.7 ±0.5 (−20.97)
LongDocURL	**25.9** ±0.7	16.2 ±2.5 (−9.67)	15.9 ±2.0 (−10.00)	25.4 ±2.1 (−0.50)	26.6 ±3.2 (+0.73)	28.0 ±2.1 (+2.13)	7.9 ±0.5 (−17.97)
MMLB-Doc†	**15.4** ±1.3	11.6 ±1.1 (−3.77)	9.1 ±2.2 (−6.33)	15.6 ±2.3 (+0.20)	10.1 ±0.9 (−5.27)	15.3 ±1.2 (−0.07)	1.8 ±0.1 (−13.57)
OVBench	**30.2** ±1.9	28.6 ±2.7 (−1.57)	29.7 ±1.9 (−0.53)	30.7 ±2.7 (+0.50)	30.5 ±2.3 (+0.27)	30.7 ±0.8 (+0.50)	26.9 ±1.9 (−3.30)
Daily-Omni	**57.3** ±3.1	56.3 ±1.1 (−1.00)	55.2 ±4.3 (−2.03)	56.5 ±1.9 (−0.77)	54.4 ±0.4 (−2.83)	54.4 ±1.2 (−2.83)	50.8 ±2.7 (−6.47)
FineVideo	**29.6** ±0.4	27.2 ±0.6 (−2.40)	28.5 ±0.7 (−1.03)	30.8 ±2.2 (+1.20)	23.1 ±1.3 (−6.47)	26.1 ±1.4 (−3.47)	17.6 ±0.3 (−12.00)
LVOBench†	**31.5** ±2.8	27.6 ±1.5 (−3.93)	28.9 ±0.8 (−2.63)	30.2 ±1.2 (−1.33)	29.7 ±2.3 (−1.87)	30.5 ±5.1 (−1.07)	29.4 ±7.0 (−2.10)
WorldSense†	**43.8** ±1.9	43.0 ±1.1 (−0.80)	42.2 ±3.2 (−1.60)	42.7 ±3.1 (−1.03)	40.3 ±0.4 (−3.43)	45.8 ±1.6 (+2.07)	41.7 ±2.8 (−2.07)
Avg. Text	**16.7** ±0.3	14.8 ±0.3 (−1.87)	15.0 ±0.6 (−1.62)	13.3 ±0.2 (−3.31)	9.1 ±1.4 (−7.51)	13.0 ±0.6 (−3.67)	4.3 ±0.1 (−12.39)
Avg. MM	**20.5** ±0.9	16.3 ±0.4 (−4.22)	16.0 ±1.2 (−4.45)	19.8 ±0.6 (−0.67)	19.1 ±1.9 (−1.42)	18.6 ±0.5 (−1.89)	7.5 ±0.2 (−13.02)
Avg. Omni	**38.5** ±1.0	36.5 ±0.7 (−1.94)	36.9 ±1.1 (−1.57)	38.2 ±1.0 (−0.29)	35.6 ±0.8 (−2.87)	37.5 ±0.8 (−0.96)	33.3 ±1.3 (−5.19)
Avg. All	**27.0** ±0.3	24.3 ±0.4 (−2.68)	24.5 ±0.4 (−2.54)	25.8 ±0.5 (−1.17)	23.5 ±1.1 (−3.54)	25.1 ±0.2 (−1.95)	17.4 ±0.5 (−9.60)

both contribute: untrained, OmniPlex already exceeds every trained omni agent (22.5 vs. ≤21.3), and RL adds 4.5. ‡Orchestra-o1 trains its text orchestrator against an LLM judge; our port scores its rubric by rule (Appendix E.2).

4.3 Ablation Study (RQ2)

As shown in Table 3, removing any component lowers Avg. All and every group average (Appendix E.3): compaction, the hypergraph memory and the coordinates cost 2.68, 2.54 and 1.17 F1 at inference, most on documents and on the questions that fold (§4.5), and the interaction floor, segment-level credit and the batch baseline 9.60, 3.54 and 1.95 in training.

4.4 Analysis of What One Look Costs (RQ3)

As shown in Table 4 and Figure 4, we compare the cost of one native look with Proposition 1's bound n_{fill} and count window hits over the 46,080 episodes run in our loop. **The window, not the budget, binds beyond text:** a knowledge-base, document or stream look costs 1.9–2.5K tokens, so $n_{\text{fill}} \leq 5 < K$, and the window is almost never reached below n_{fill} looks but in 27–63% of episodes from then on (a). **Segments keep OmniPlex-RL under it:** with 3.1 knowledge-base and 4.6 document calls per episode, only 3.0% and 0.8% of its episodes reach the window (b).

Table 4: The cost of one native look: median tokens, Proposition 1's bound n_{fill}, and window hits (%) below and from n_{fill} looks.

			Hits (%)	
Look	**Tok.**	n_{fill}	$< n_{\text{fill}}$	$\geq n_{\text{fill}}$
Text passage	819	11	0.0	never
KB entry + photo	2458	4	0.0	63.4
Three-page read	1904	5	0.6	27.2
A/V window, 180 s	1969	4	0.0	45.1

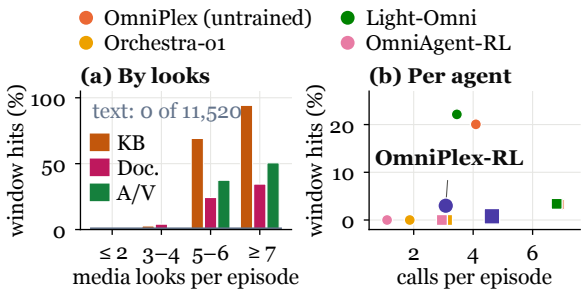


Figure 4: Window hits (a) by media looks and (b) per agent against its calls (circles: KB, squares: Doc.).

4.5 Analysis of What Survives Compaction (RQ4)

As shown in Figure 5, we split the questions by whether the full system folds on them (166 of 1,536, in at least two of three runs). **The lead is widest where a context is compacted:** +6.1 over the strongest baseline on folded questions against +4.3 elsewhere (a), where removing compaction, the memory or the coordinates costs 9.0, 9.8 and 4.0 F1, at most 1.9 elsewhere (b). **The hypergraph carries what a hand-over drops:** a short gold answer reaches the successor after 29.6% of folds with it and 9.7% without (c), each read-back line ending in its source call (d).

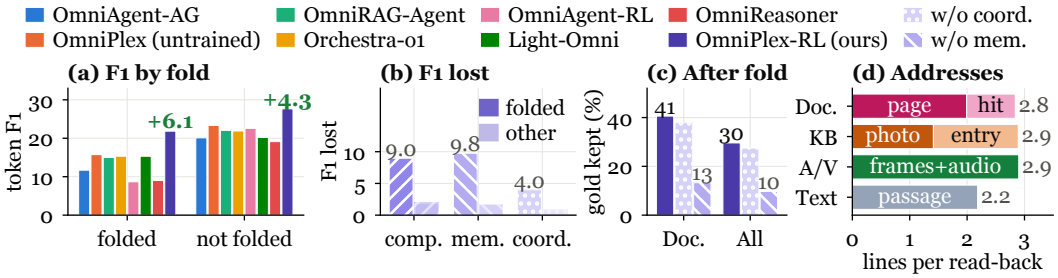


Figure 5: What survives compaction: (a) F1 by fold; (b) F1 lost; (c) gold kept; (d) read-back lines.

4.6 Analysis of Credit under a Single Rollout (RQ5)

As shown in Figure 6, from the training logs, **a per-kind moving baseline leaves no step without signal:** none of OmniPlex-RL's steps has all-zero advantages, while group baselines zero out 15.8–68.4% of the steps of four of the five trained baselines (a). **Per-segment samples train each token on the prefix it saw:** the train–rollout gap is 0.10 nats against 2.03 for one concatenated sample (b), and the policy keeps looking, while without the interaction floor it stops by step 27 (c).

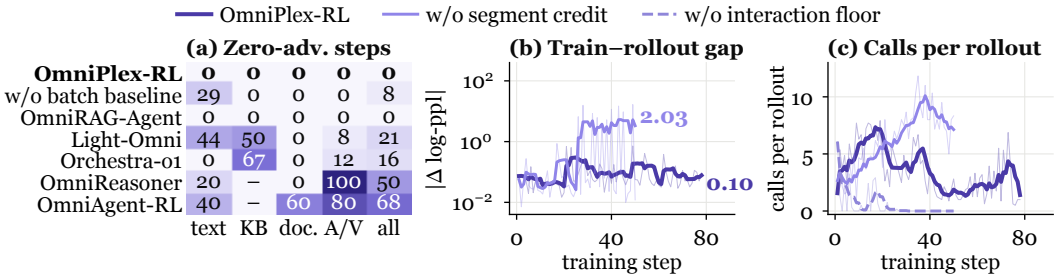


Figure 6: Credit under one rollout: (a) all-zero-advantage steps (%); (b) train–rollout gap; (c) calls.

4.7 Robustness and Significance of the Gains (RQ6)

As shown in Figure 7, following the length-split analysis of OmniAgent-RL (Xing et al., 2026), OmniPlex-RL is **ahead at every stream length**, from +5.6 under a minute to +1.3 over ten minutes (a), **significant against every baseline**, its 95% paired-bootstrap interval excluding zero against all seven (b), and **ahead on all twelve suites**, as much held out (+2.15) as in distribution (+2.21) (c).

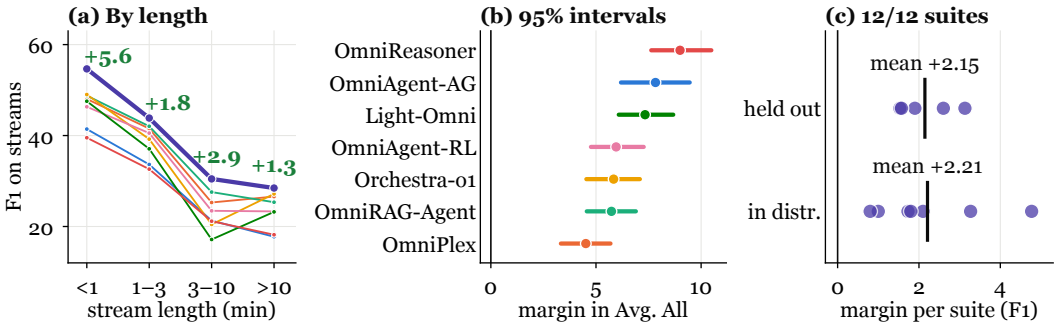
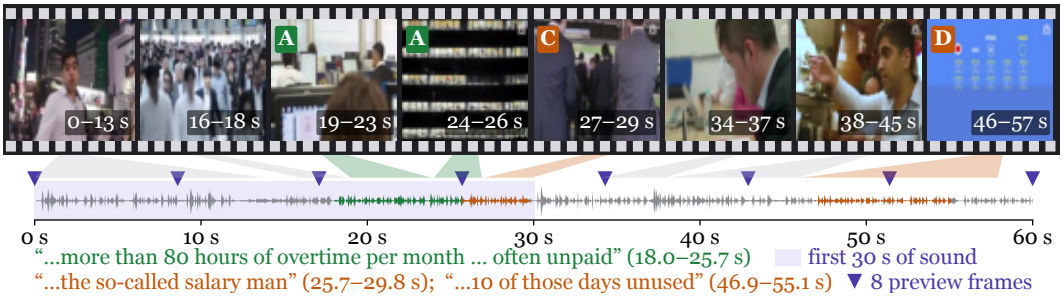
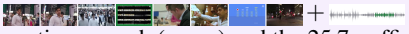


Figure 7: Robustness: (a) F1 by stream length; (b) 95% intervals of the margin; (c) margin per suite.

Table 5: Case study on Daily-Omni #805 (60 s clip): traces and three-run answers from the logs.

Question. Which visual element directly coincides with the audio discussion of unpaid overtime exceeding 80 hours per month? (A) Employees working at desks in a busy office (**gold**); (B) A crowded public transportation hub; (C) Salarymen walking on a platform in dark suits; (D) A blue screen displaying vacation day statistics.



Method	Trace	Runs	F1
Qwen2.5-Omni-3B	The question text alone; no media, no tools.	C C C	0.00
OmniAgent-AG	`[audio caption]` "00:17–00:24: Explanation of the unpaid overtime ..." → `[watch 0-60 s]` → `[memory]` "C. ... the visual element shows salarymen, which aligns with the context"	C C C	0.00
OmniReasoner	Preview frame times as text; answers without a zoom.	D C A	0.33
OmniRAG-Agent	Answers without a tool call.	A D C	0.33
OmniPlex		A D D	0.33
Orchestra-o1	The same opening view as OmniPlex-RL (the preview frames, the first 30 s	C D D	0.00
OmniAgent-RL	of sound, the question); each answers at once, without a further look.	D A D	0.33
Light-Omni		A D D	0.33
OmniPlex-RL **native replay**	**One native context**  + + question → **(A)**: it holds the overtime speech (green) and the 25.7 s office frame (outlined). In run 2 the option votes (App. D.2) turn its (D) into (A).	**A A A**	**1.00**

4.8 Accuracy against Interaction Cost (RQ7)

As shown in Figure 8, following OmniAgent-AG (Tao et al., 2026a), we place every column of Table 2 by its calls per episode against its accuracy. **More accuracy per look.** OmniPlex-RL tops the frontier (dashed) at 27.0 with 2.8 calls, while the two other agents that look over three times reach 22.5 and 19.7 with 3.7 and 3.6.

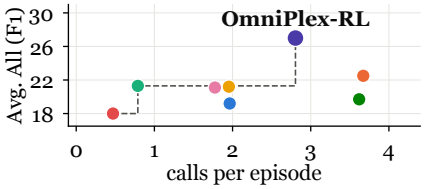


Figure 8: Accuracy against calls.

4.9 Case Study (RQ8)

As shown in Table 5, the answer lies in aligning sound with image. **Text carriers match words, not moments:** OmniAgent-AG's captions even date the overtime passage to 00:17–00:24, yet its memory picks the shot of the next sentence, the "salary man" (C), in all three runs. **A native context keeps sound and frames together:** OmniPlex-RL holds both and ends on (A) in all runs.

5 Conclusion

We presented OmniPlex-RL, a native omni-modal agent that reads text corpora, knowledge bases, long documents and audio-video streams as they are. As native looks fill the window before the call budget, it compacts its context into segments linked by an omni-modal hypergraph memory whose units keep the call that re-opens their raw media; as folding leaves no consistent sample, it trains each segment as a sample under single-rollout RL with a per-modality moving batch baseline. On twelve benchmarks OmniPlex-RL leads every suite, 4.5 F1 above the strongest baseline, and its gains hold on held-out suites, at every stream length and at fewer calls per point of accuracy.

AI USE STATEMENT

We used generative AI tools to assist with code implementation, the preparation of figures and tables, and language editing of the manuscript. All AI-assisted code, results and text were reviewed, verified and tested by the authors, who take full responsibility for the content of this work.

ETHICS STATEMENT

This work uses only publicly released benchmarks, corpora and media under their original terms (Appendix D), and involves no human subjects, crowdsourced annotation or collection of personal data. OmniPlex-RL answers questions solely from the provided sources through retrieval and viewing tools, and we release it for research use to support transparent study of omni-modal agents.

REPRODUCIBILITY STATEMENT

The training and evaluation code of OmniPlex-RL, with its launch scripts and configurations, is available at the anonymous repository linked in the abstract. Section 3 and Section 4.1 specify the method and setup, and the appendix gives every prompt verbatim (Appendix A), complete proofs of all propositions (Appendix B), the full algorithm (Appendix C), benchmark construction and evaluation (Appendix D), baseline implementations (Appendix E) and per-experiment details (Appendix F).

REFERENCES

Weihao Bo, Shan Zhang, Yanpeng Sun, Jingjing Wu, Qunyi Xie, Xiao Tan, Kunbin Chen, Wei He, Xiaofan Li, Na Zhao, Jingdong Wang, and Zechao Li. Agentic learner with grow-and-refine multimodal semantic memory, 2026. URL https://arxiv.org/abs/2511.21678.

Yang Chen, Hexiang Hu, Yi Luan, Haitian Sun, Soravit Changpinyo, Alan Ritter, and Ming-Wei Chang. Can pre-trained vision and language models answer visual information-seeking questions? In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 14948–14968, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.925. URL https://aclanthology.org/2023.emnlp-main.925/.

Yu Chen, Caorui Li, Ziyu Xiong, Yidong Wang, Mingqi Gao, Shuman Liu, Biao Liu, Chunfeng Yang, Anxiang Zeng, Haibo Zhang, and Chaofan Chen. Omnireasoner: Thinking with long audio-video via native tool use, 2026. URL https://arxiv.org/abs/2607.19339.

Zijian Chen, Xueguang Ma, Shengyao Zhuang, Ping Nie, Kai Zou, Andrew Liu, Joshua Green, Kshama Patel, Ruoxi Meng, Mingyi Su, Sahel Sharifymoghaddam, Yanxi Li, Haoran Hong, Xinyu Shi, Xuye Liu, Nandan Thakur, Crystina Zhang, Luyu Gao, Wenhu Chen, and Jimmy Lin. Browsecomp-plus: A more fair and transparent evaluation benchmark of deep-research agent, 2025. URL https://arxiv.org/abs/2508.06600.

Thang Dang, Yuma Ichikawa, Sakina Fatima, and Koichi Shirahata. Addressable recall compaction for long context-window control in ai agents, 2026. URL https://arxiv.org/abs/2607.25066.

Chao Deng, Jiale Yuan, Pi Bu, Peijie Wang, Zhong-Zhi Li, Jian Xu, Xiao-Hui Li, Yuan Gao, Jun Song, Bo Zheng, and Cheng-Lin Liu. LongDocURL: a comprehensive multimodal long document benchmark integrating understanding, reasoning, and locating. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1135–1159, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.57. URL https://aclanthology.org/2025.acl-long.57/.

Run-Ze Fan, Zihao Zhang, Simin Ma, Yebowen Hu, Shouju Wang, Kaiqiang Song, Fei Liu, Hamed Zamani, and Xiaoyang Wang. An empirical study of harness design for coding agents, 2026. URL https://arxiv.org/abs/2609.20804.

Miquel Farré, Andi Marafioti, Lewis Tunstall, Leandro Von Werra, and Thomas Wolf. Finevideo. https://huggingface.co/datasets/HuggingFaceFV/finevideo, 2024.

Xian Gao, Jinpeng Wang, Jiacheng Ruan, Guangyu Cao, Ting Liu, and Yuzhuo Fu. Memo: Multimodal evidence memory organization for long-horizon llm agents, 2026. URL https://arxiv.org/abs/2609.07471.

Xubin Hao, Hongjin Meng, Xin Yin, Jiawei Zhu, and Chenpeng Cao. Self-gc: Self-governing context for long-horizon llm agents, 2026. URL https://arxiv.org/abs/2607.00692.

Jack Hong, Shilin Yan, Jiayin Cai, Xiaolong Jiang, Yao Hu, and Weidi Xie. Worldsense: Evaluating real-world omnimodal understanding for multimodal LLMs. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=YxsfxAvJv4.

Zinco J, Xunjie Zhu, Shen Huang, Zhenyi Wang, Pengjun Xie, and Jieping Ye. Memorywalker: Stop training agents on contexts they never saw, 2026. URL https://arxiv.org/abs/2609.00865.

Dingyi Kang, Dongming Jiang, Yi Li, Guanpeng Li, and Bingzhe Li. V-mem: Modality-routed retrieval for long-term multimodal agentic memory, 2026a. URL https://arxiv.org/abs/2608.01543.

Minki Kang, Wei-Ning Chen, Dongge Han, Huseyin A Inan, Lukas Wutschitz, Yanzhi Chen, Robert Sim, and Saravan Rajmohan. ACON: Optimizing context compression for long-horizon LLM agents. In *Forty-third International Conference on Machine Learning*, 2026b. URL https://openreview.net/forum?id=5EmOOLtH5P.

Caorui Li, Yu Chen, Yiyan Ji, Jin Xu, Zhenyu Cui, Shihao Li, Yuanxing Zhang, Zhenghao Song, Dingling Zhang, Ying He, Haoxiang Liu, Yuxuan Wang, Qiufeng Wang, Jiafu Tang, Zhenhe Wu, Jiehui Luo, Zhiyu Pan, Weihao Xie, Chenchen Zhang, Zhaohui Wang, Jiayi Tian, Yanghai Wang, Zhe Cao, Minxin Dai, Ke Wang, Runzhe Wen, Yinghao Ma, Yaning Pan, Sungkyun Chang, Termeh Taheri, Haiwen Xia, Christos Plachouras, Emmanouil Benetos, Yizhi Li, Ge Zhang, Jian Yang, Tianhao Peng, Zili Wang, Minghao Liu, Junran Peng, Zhaoxiang Zhang, and Jiaheng Liu. Omnivideobench: Towards audio-visual understanding evaluation for omni MLLMs. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=ItRYEe8E61.

Geng Li, Yuhao Wang, Dong Li, Jianye Hao, and Yuxin Peng. Graphmemix: Query-aware evidence forests for long-term multimodal agent memory, 2026b. URL https://arxiv.org/abs/2608.26983.

Jiazheng Li, Chi-Hao Wu, Yunze Liu, Kaize Ding, Jundong Li, and Chuxu Zhang. Bridging modalities, spanning time: Structured memory for ultra-long agentic video reasoning, 2026c. URL https://arxiv.org/abs/2605.08271.

Tianjian Li, Jingyu Zhang, William Jurayj, Xi Wang, Chuanyang Jin, Mehrdad Farajtabar, Eric Nalisnick, and Daniel Khashabi. Self-compacting language model agents, 2026d. URL https://arxiv.org/abs/2606.23525.

Xiaochuan Li, Ryan Ming, Meng Chu, Shuai Shao, Rong Jin, and Chenyan Xiong. Acm: Agentic context management for long horizon tasks, 2026e. URL https://arxiv.org/abs/2607.23809.

Xiaoxi Li, Wenxiang Jiao, Jiarui Jin, Haoxuan Li, Hao Wang, Shijian Wang, Guanting Dong, Jiajie Jin, Yinuo Wang, Yuan Lu, Ji-Rong Wen, Zhicheng Dou, and Zhouchen Lin. Omnigaia: Towards native omni-modal ai agents, 2026f. URL https://arxiv.org/abs/2602.22897.

Yujiang Li, Zhenyu Hou, Yi Jing, Jie Tang, and Yuxiao Dong. Compactionrl: Reinforcement learning with context compaction for long-horizon agents, 2026g. URL https://arxiv.org/abs/2607.05378.

Ning Liao, Zihao Long, Xiaoxing Wang, Xue Yang, Yaoming Wang, Ziyuan Zhuang, Xunliang Cai, Rongxiang Weng, and Junchi Yan. Ace: Pluggable adaptive context elasticizer across agents, 2026. URL https://arxiv.org/abs/2606.31564.

Yin Lin, Elaine Ang, Erkang Zhu, Bolin Ding, and Jingren Zhou. Context as an environment: Programmatic context management for long-horizon agents, 2026. URL https://arxiv.org/abs/2608.21690.

Jiaqi Liu, Zipeng Ling, Shi Qiu, Yanqing Liu, Siwei Han, Peng Xia, Haoqin Tu, Zeyu Zheng, Cihang Xie, Charles Fleming, Mingyu Ding, and Huaxiu Yao. Omni-simplemem: Autoresearch-guided discovery of lifelong multimodal agent memory, 2026a. URL https://arxiv.org/abs/2604.01007.

Yu Liu, Wenxiao Zhang, Cheng Hu, Cong Cao, Fangfang Yuan, Xinyu Wang, Jin B. Hong, and Yanbing Liu. Mirage: How conversation state shapes historical evidence use in multimodal personal agents, 2026b. URL https://arxiv.org/abs/2609.19059.

Lin Long, Yichen He, Wentao Ye, Yiyuan Pan, Yuan Lin, Hang Li, Junbo Zhao, and Wei Li. Seeing, listening, remembering, and reasoning: A multimodal agent with long-term memory. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=PMz29A7Muq.

Miao Lu, Weiwei Sun, Weihua Du, Zhan Ling, Xuesong Yao, Kang Liu, and Jiecao Chen. Scaling llm multi-turn rl with end-to-end summarization-based context management, 2025. URL https://arxiv.org/abs/2510.06727.

Yijun Lu, Rui Ye, Yuwen Du, Jiajun Wang, Songhua Liu, and Siheng Chen. Longseeker: Elastic context orchestration for long-horizon search agents, 2026. URL https://arxiv.org/abs/2605.05191.

Fengji Ma, Yan Rong, Xu Li, Xuenan Xu, Chen Zhang, and Li Liu. Ace-cap: Active evidence acquisition via agentic co-evolution for long-paragraph fine-grained audio captioning, 2026a. URL https://arxiv.org/abs/2608.16162.

Ming Ma, Yi Zhu, Yiran Zhong, Feida Zhu, Weigao Sun, Junhan Shi, Lingrui Mei, Tianming Yang, and Steven Hoi. Omni-decision: A progressive evidence-state agent system for omni-modal qa, 2026b. URL https://arxiv.org/abs/2607.11433.

Yubo Ma, Yuhang Zang, Liangyu Chen, Meiqi Chen, Yizhu Jiao, Xinze Li, Xinyuan Lu, Ziyu Liu, Yan Ma, Xiaoyi Dong, Pan Zhang, Liangming Pan, Yu-Gang Jiang, Jiaqi Wang, Yixin Cao, and Aixin Sun. Mmlongbench-doc: Benchmarking long-context document understanding with visualizations. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 95963–96010. Curran Associates, Inc., 2024. doi: 10.52202/079017-3041. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/ae0e43289bffea0c1fa34633fc608e92-Paper-Datasets_and_Benchmarks_Track.pdf.

Thomas Mensink, Jasper Uijlings, Lluis Castrejon, Arushi Goel, Felipe Cadar, Howard Zhou, Fei Sha, André Araujo, and Vittorio Ferrari. Encyclopedic vqa: Visual questions about detailed properties of fine-grained categories. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3113–3124, October 2023.

Chang Nie, Jiaju Wei, Junlan Feng, Chaoyou Fu, and Caifeng Shan. Light-omni: Reflex over reasoning in agentic video understanding with long-term memory, 2026. URL https://arxiv.org/abs/2607.05511.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. Memgpt: Towards llms as operating systems, 2024. URL https://arxiv.org/abs/2310.08560.

Zhuoshi Pan, Qizhi Pei, Junru Lu, Honglin Lin, H. Vicky Zhao, Di Yin, and Xing Sun. Contextpilot: Teaching agents for proactive context management via fine-grained rl, 2026. URL https://arxiv.org/abs/2608.28476.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Jiaqi Su, Cong Pang, Jiawei Hong, Tiankuo Yao, Zixuan Chen, Xin Lou, and Lewei Lu. When history is multimodal: Rethinking context management for long-horizon agents, 2026. URL https://arxiv.org/abs/2608.29897.

Jingyu Sun, Yan Lin, Yuyang Xue, Yifan Wang, Zhengtao Yao, Rui Qian, Zefeng Xu, Jiachen Li, Xianyang Liu, Jiancheng Pan, Jingyuan Sun, Syed Murtuza Baker, and Hongpeng Zhou. Pmmc: Prospective multimodal memory compilation for long-term lvlm agents, 2026. URL https://arxiv.org/abs/2608.00962.

Weiwei Sun, Miao Lu, Zhan Ling, Kang Liu, Xuesong Yao, Yiming Yang, and Jiecao Chen. Scaling long-horizon llm agent via context-folding, 2025. URL https://arxiv.org/abs/2510.11967.

Keda Tao, Wenjie Du, Bohan Yu, Weiqiang Wang, Jian Liu, and Huan Wang. Active perception agent for omnimodal audio-video understanding, 2026a. URL https://arxiv.org/abs/2512.23646.

Keda Tao, Yuhua Zheng, Jia Xu, Wenjie Du, Kele Shao, Hesong Wang, Xueyi Chen, Xin Jin, Junhan Zhu, Bohan Yu, Weiqiang Wang, Jian Liu, Can Qin, Yulun Zhang, Ming-Hsuan Yang, and Huan Wang. Lvomnibench: Pioneering long audio-video understanding evaluation for omnimodal llms, 2026b. URL https://arxiv.org/abs/2603.19217.

Qwen Team. Qwen3.5-omni technical report, 2026. URL https://arxiv.org/abs/2604.15804.

Yizhou Tian, Zizhe Chen, Shiyuan Deng, Garry Yang, Zijie Dai, Luohao Pan, Hao Lin, Peiqi Yin, Xiao Yan, and James Cheng. Cam: Question answering on entity-centric videos with continuous extraction and adaptive querying, 2026. URL https://arxiv.org/abs/2609.06504.

Hayder Tirmazi, Sam Markelon, Allison Bishop, and Michael Mitzenmacher. Context compaction theory, 2026. URL https://arxiv.org/abs/2608.01326.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022. doi: 10.1162/tacl_a_00475. URL https://aclanthology.org/2022.tacl-1.31/.

Junxi Wang, Te Sun, Jiayi Zhu, Junxian Li, Haowen Xu, Zichen Wen, Xuming Hu, Zhiyu li, and Linfeng Zhang. StreamMeCo: Long-term agent memory compression for efficient streaming video understanding. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 13234–13251, San Diego, California, United States, July 2026a. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.647. URL https://aclanthology.org/2026.findings-acl.647/.

Junxi Wang, Te Sun, Jiayi Zhu, Chen Zhang, Siyuan Li, Xuyang Liu, Zichen Wen, Xiaobing Tu, Jinkui Ren, Xiantao Zhang, Ziqi Yuan, and Linfeng Zhang. Memforest: Efficient agent memory management via eventtree partitioning and progressive merging, 2026b. URL https://arxiv.org/abs/2609.08273.

Qiuchen Wang, Shihang Wang, Yu Zeng, Qiang Zhang, Fanrui Zhang, Zhuoning Guo, Bosi Zhang, Wenxuan Huang, Lin Chen, Zehui Chen, Pengjun Xie, and Ruixue Ding. Vimrag: Navigating massive visual context in retrieval-augmented generation via multimodal memory graph, 2026c. URL https://arxiv.org/abs/2602.12735.

Zhenting Wang, Huancheng Chen, Jiayun Wang, and Wei Wei. Memex(rl): Scaling long-horizon llm agents via indexed experience memory, 2026d. URL https://arxiv.org/abs/2603.04257.

Zhiqi Wang, Yichi Zhang, Dongwon Lee, and Yuchen Yang. Lost in compaction: Evaluating side-constraint loss under context compaction, 2026e. URL https://arxiv.org/abs/2608.11242.

Zijun Xie, Binbin Zheng, Enlei Gong, Jihua Liu, Yuyang You, Lingfeng Liu, Jiayao Tang, Guanqun Zhao, Aoqi Hu, and Zeyu Chen. Echo: Prune to act, trace to learn with selective turn memory in agentic rl, 2026. URL https://arxiv.org/abs/2606.31650.

Zhenghao Xing, Ruiyang Xu, Yuxuan Wang, Jinzheng He, Ziyang Ma, Qize Yang, Yunfei Chu, Jin Xu, Junyang Lin, Chi-Wing Fu, and Pheng-Ann Heng. Native active perception as reasoning for omni-modal understanding. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=3dLPVNhFZh.

Binyan Xu, Haitao Li, and Kehuan Zhang. Llm agents are latent context managers: Eliciting self-managed context via state proprioception, 2026a. URL https://arxiv.org/abs/2606.30005.

Buqiang Xu, Zirui Xue, Dianmou Chen, Chenyang Fu, Chiyu Wu, Caiying Huang, Chen Jiang, Jizhan Fang, Xinle Deng, Yijun Chen, Yunzhi Yao, Xuehai Wang, Jin Shang, Gong Yu, and Ningyu Zhang. Tokenpilot: Cache-efficient context management for llm agents, 2026b. URL https://arxiv.org/abs/2606.17016.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. Qwen2.5-omni technical report, 2025. URL https://arxiv.org/abs/2503.20215.

Ke Xu, Yuhao Wang, Ziyang Cheng, Hongcheng Liu, Yanfeng Wang, and Yu Wang. Agentic active omni-modal perception for multi-hop audio-visual reasoning, 2026c. URL https://arxiv.org/abs/2605.28192.

Ruiyao Xu, Tiankai Yang, and Wei-Chieh Huang. Hyperskill: Self-evolving llm agents via hypergraph-structured skill memory, 2026d. URL https://arxiv.org/abs/2608.16114.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2369–2380, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259/.

Hanshu Yao, Janfeng Zhong, Niu Lian, and Jinpeng Wang. Docmemo: Dynamic evidence discovery via probabilistic memory-guided retrieval for multi-modal document understanding, 2026a. URL https://arxiv.org/abs/2608.07067.

Yiqun Yao, Naitong Yu, Xiang Li, Xin Jiang, Xuezhi Fang, Wenjia Ma, Xuying Meng, Jing Li, Aixin Sun, and Yequan Wang. Egomem: Lifelong memory agent for full-duplex omnimodal models, 2026b. URL https://arxiv.org/abs/2509.11914.

Rui Ye, Zhongwang Zhang, Kuan Li, Huifeng Yin, Zhengwei Tao, Yida Zhao, Liangcai Su, Liwen Zhang, Zile Qiao, Xinyu Wang, Pengjun Xie, Fei Huang, Siheng Chen, Jingren Zhou, and Yong Jiang. Agentfold: Long-horizon web agents with proactive context management, 2025. URL https://arxiv.org/abs/2510.24699.

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

Lu Yi, Runlin Lei, Liuyi Yao, Yuexiang Xie, Yuyang Li, Wenhao Zhang, Zhewei Wei, Yaliang Li, and Jian-Yun Nie. Learning agent-compatible context management for long-horizon tasks, 2026. URL https://arxiv.org/abs/2605.30785.

Ziyun Zeng, Hang Hua, Bocheng Zou, Mu Cai, Rogerio Feris, and Jiebo Luo. Mementogui: Learning agentic multimodal memory control for long-horizon gui agents, 2026. URL https://arxiv.org/abs/2605.18652.

Saber Zerhoudi, Jelena Mitrovic, and Michael Granitzer. The compaction cliff in long-running ai agent memory, 2026. URL https://arxiv.org/abs/2608.22752.

Fan Zhang, Guangming Yao, Jinyang Wu, Hao Wu, Zheng Lian, Xinyu Geng, Jingdong Chen, Yi Yuan, and Pheng-Ann Heng. Videogaia: A benchmark for general ai assistants on agentic video understanding, 2026a. URL https://arxiv.org/abs/2608.14718.

Fan Zhang, Vireo Zhang, Shengju Qian, Haoxuan Li, Hao Wu, Jinyang Wu, Donghao Zhou, Zhihong Zhu, Zheng Lian, Xin Wang, and Pheng-Ann Heng. Orchestra-o1: Omnimodal agent orchestration, 2026b. URL https://arxiv.org/abs/2606.13707.

Guangzhe Zhang. Correct now, insufficient later: Auditing update sufficiency in context compression, 2026. URL https://arxiv.org/abs/2609.20045.

Yuxiang Zhang, Jiangming Shu, Ye Ma, Xueyuan Lin, Shangxi Wu, and Jitao Sang. Memory as action: Autonomous context curation for long-horizon agentic tasks, 2026c. URL https://arxiv.org/abs/2510.12635.

Zijian Zhou, Ao Qu, Zhaoxuan Wu, Sunghwan Kim, Alok Prakash, Daniela Rus, Bryan Kian Hsiang Low, and Paul Pu Liang. MEM1: Learning to synergize memory and reasoning for efficient long-horizon agents. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=XY8AaxDSLb.

Ziwei Zhou, Rui Wang, Zuxuan Wu, and Yu-Gang Jiang. Daily-omni: Towards audio-visual reasoning with temporal alignment across modalities, 2026b. URL https://arxiv.org/abs/2505.17862.

Yifan Zhu, Xinyu Mu, Tao Feng, Zhonghong Ou, Yuning Gong, and Haoran Luo. Omnirag-agent: Agentic omnimodal reasoning for low-resource long audio-video question answering, 2026. URL https://arxiv.org/abs/2602.03707.

APPENDIX

A PROMPTS USED IN OMNIPLEX-RL

All prompts are the released code's own strings, verbatim: a {slot} is filled at run time and a [bracket] names the questions a line is sent on. Each is answered by the policy π_θ itself, as no other language model is prompted (Ext is a local parser and ϕ a text encoder), and those of Appendices A.1–A.2 serve training and evaluation alike.

A.1 AGENTIC INTERACTION PROMPT

Figure 9 gives $\mathbf{p}(x)$ of Eq. (1): one system frame, filled with the tools of the question's environment kind, the user message, the policy's first look at a photograph, and the turn protocol of §3.1.

System prompt (every environment)

You answer a question by looking things up in the environment you are given. You cannot see the environment until you ask.
{tool blocks of the environment kind (coloured boxes)}
---- BEGIN FUNCTION #{n}: finish ----
Description: Return the final result when you have a definitive answer or cannot progress further. Provide a concise answer plus a brief, evidence-grounded explanation.
Parameters:
(1) answer (string, required): A succinct, final answer.
(2) explanation (string, required): A brief explanation for your final answer. For this section only, cite evidence documents inline by placing their docids in square brackets at the end of sentences (e.g., [20]). Do not include citations anywhere else.
(3) confidence (string, optional): Confidence: your confidence score between 0% and 100% for your answer
---- END FUNCTION #{n} ----
For each call, emit a json object inside <tool_call></tool_call> tags:
<tool_call>
{"name": <name>, "arguments": <args>}
</tool_call>
Submit with: <tool_call>
{"name": "finish", "arguments": {"answer": "..."}}
</tool_call>
You may make at most 8 tool calls, so make each one count; submit as soon as you can support an answer. Give the answer only, with no explanation; if the question lists options, answer with the letter.

User message

{the question's photograph, or the stream's preview: its frames and first 30 s of sound}
Question: {question}
Reply with the option letter only. [choice questions]
What the question's own media show (your first look): {first-look note} [a question with a photograph]

First-look prompt

System: You are looking at the material a question came with, before answering it. In at most four sentences, write what is shown or heard that could matter: the subject's identity if you recognise it (a place, a building, a species, a product, a person's role), any text on screen, names, numbers, objects, actions, what is said. Only what is actually shown; write 'unsure' rather than guess.
User: {photograph}
Question: {question}
What do you see and hear here that could matter for this question?

Corpus · BC-Plus, HotpotQA, MuSiQue

---- BEGIN FUNCTION #1: search ----
Description: Search the corpus and get back the top passages with their identifiers. Query with the fact you are missing rather than the whole question; a multi-hop question usually needs one search per hop, using what the previous hit told you. Each call consumes context.
Parameters:
(1) query (string, required): What to look for.
(2) topk (integer, optional): How many results to return (default 5).
---- END FUNCTION #1 ----

Knowledge base · E-VQA, InfoSeek

---- BEGIN FUNCTION #1: search_by_image ----
Description: Look up the image attached to the question in the knowledge base, and get back the entries whose subject looks like it, each with its text and its own picture. Start here when the question points at something in the image ('this lake', 'this bird') rather than naming it -- a text search cannot find an entity the question never names. Once an entry tells you the name, `search` can follow up on it.
Parameters:
(1) topk (integer, optional): How many entries to return (default 5).
---- END FUNCTION #1 ----
---- BEGIN FUNCTION #2: search ----
Description: Search the knowledge base for entries about an entity. Each hit comes back as the entry's text together with its image, because the base is multimodal and the picture often carries what the text does not. Query with what you are trying to identify -- a species, a landmark, a model -- not with the question itself. Each call consumes context, so prefer a sharper query over a second call.
Parameters:
(1) query (string, required): What to look for.
(2) topk (integer, optional): How many results to return (default 5).
---- END FUNCTION #2 ----

Document · LongDocURL, MMLB-Doc

---- BEGIN FUNCTION #1: search_pages ----
Description: Find which of this document's {n} pages mention something. Returns page numbers with a text snippet, ranked. The snippet is extracted text only -- figures, charts and table layout are not in it, so follow a hit with read_page to actually look at the page.
Parameters:
(1) query (string, required): Words expected to appear on the page.
(2) topk (integer, optional): How many pages to return (default 5).
---- END FUNCTION #1 ----
---- BEGIN FUNCTION #2: read_page ----
Description: Look at pages of this document as images. It has {n} pages, numbered 1 to {n}. Pass the page numbers you want, at most 3 per call. Use this once you know roughly where to look -- the page image is the only way to read a chart, a table, or anything the layout carries.
Parameters:
(1) pages (array, required): Page numbers to look at, between 1 and {n}.
---- END FUNCTION #2 ----

Stream · OVBench, Daily-Omni, FineVideo, LVOBench, WorldSense

---- BEGIN FUNCTION #1: watch ----
Description: Re-watch a time range of the video at higher temporal and spatial detail, and re-hear its audio. The opening view of the video is a coarse preview: long videos are sampled to a handful of frames, so brief events fall between them. Call watch with the interval you need (seconds from the start) and the segment comes back as real frames and sound, not a description. Prefer short intervals -- a tight window returns finer detail than a wide one for the same budget, and each call consumes context.
Parameters:
(1) start (number, required): Start of the interval, in seconds from the beginning of the video.
(2) end (number, required): End of the interval, in seconds. Keep the span under a few minutes.
(3) modality (string, optional): What to return: 'both' (default), 'video' for frames only, or 'audio' for sound only.
---- END FUNCTION #1 ----

Turn protocol

a reply without a call, once → No tool call found. Emit one inside <tool_call></tool_call>, or submit with the `finish` tool.
after every observation → [{k} tool call(s) left]
a call already made, bar a media re-open in a later segment → You already called {tool} with these exact arguments and got the result above. Repeating it returns the same thing. Use different arguments, open one of the results, or finish.
an episode that ends without `finish` → [No tool calls left. Answer now with the `finish` tool, using what you have seen.] Reply with the option letter only. [choice questions]

Figure 9: Prompt $\mathbf{p}(x)$ for the policy π_θ in Eq. (1), with the first-look prompt and the turn protocol.

A.2 Compaction and Hand-over Prompts

Three prompts serve a segment that closes (Eq. (3); Figure 10): the hand-over h_k, in four fixed parts, which on a choice question also copies the options and asks for an option letter; an evidence note for each media observation, whose reply after keep is z_t in Eq. (4); and the successor prompt, holding h_k and the read-back $\mathbf{b}_k$ of Eq. (6), each evidence line of which ends in the call that re-opens it.

Hand-over prompt (free-form answers)

Your context is full and will be cleared. Write a short hand-over for yourself, inside <summary></summary>, with exactly these four parts:
1. Question: copy the question exactly as it was asked.
2. Seen: what you actually observed that bears on the question, one line each, each with where it came from (a page number, a time range in seconds, an entry name or docid). Only what you saw; nothing guessed.
3. Best answer so far: only if nothing is still missing (part 4 says "nothing"), a short answer (a name, a number, a phrase); otherwise write "undecided".
4. Still missing: what one more look would have to show, if anything.
Shape:
<summary>
Question: ...
Seen: - ... (where)
  - ... (where)
Best answer so far: a short answer, or undecided
Still missing: ..., or nothing
</summary>
Be brief and literal. Put nothing outside the tags.

Evidence-note prompt

System: You are recording evidence for a question. Look at what a tool returned and write, in at most three sentences, what in it bears on the question: quote the exact words, numbers, names, captions or timestamps you see. If nothing bears on it, write 'nothing relevant'. Do not answer the question and do not name an option or a letter: record what is shown, nothing more; do not guess.
User: {for a knowledge-base result, the question's photograph} {the observation's images, or its frames and sound}
The first image is the question's own photograph; the rest are what the tool returned. [knowledge-base results]
Question: {question stem, without options}
Tool result ({the call that produced it}): {"the images above", "the frames and sound above", or the first entry's text, up to 500 characters}
What here bears on the question?

Successor prompt

You are continuing this task in a fresh context. A previous session of yours made the progress below; read it, then act.
<previous_progress>
{hand-over}
</previous_progress>
<memory>
{read-back}
</memory>
Do not write another summary. Your reply must be exactly one tool call inside <tool_call></tool_call> -- a tool from the list to look further, or `finish` with the answer.
[{k} tool call(s) left. Use them if the summary is not enough to answer; otherwise submit with `finish`.]

Read-back of MMLB-Doc question 502, as the successor received it

<prior_findings>
[Evidence]
- The image shows that Cubans and South Americans in the U.S. are more likely than some other Hispanic origin groups to say they have a positive impression of capitalism. Additionally, Cubans and South Americans are more likely to express dissatisfaction with the nation's direction. (from read_page(pages=[95, 103, 104]))
- The text states that Cubans, more than other Hispanic origin groups, are more likely than others to say that the Republican Party represents people like them somewhat or very well. (from read_page(pages=[4, 29]))
[Re-open (re-issue any of these calls to see the original evidence)]
  - read_page(pages=[95, 103, 104])
  - read_page(pages=[4, 29])
</prior_findings>

Figure 10: Prompts for compaction in Eqs. (4) and (6), with the read-back $\mathbf{b}_1$ of the case study.

A.3 Evaluation-time Prompts

Figure 11 gives the prompts of the evaluation-time rules of Appendix D.2.

Look-up before the first turn

knowledge base → Before your first call, the knowledge base was searched by the question's own photograph; the best-matching entries are below (this counted as one call). Identify the subject from them, look further if none of them is it, then submit with `finish`.
corpus → Before your first call, the corpus was searched with the question itself; the passages are below (this counted as one call). Search once more for the fact they do not settle, then submit with `finish`.

Grounded finish

document, pages surfaced → Not submitted: the answer must rest on a page this episode has looked at. The pages your search ranked highest are opened below (read_page {pages}); read them, then submit with `finish`, or look further.
document, none surfaced → Not submitted: nothing in this episode has looked at the document. Its pages were searched with the question below; open the likely ones with `read_page`, then finish.
corpus → Not submitted: one search rarely settles this kind of question. To check your proposed answer '{answer}', the corpus was searched for it; the passages are below. Confirm it, revise it, or search once more for the fact still missing, then finish.
no look to take → Not submitted: nothing in this episode has looked at the evidence yet. Use {tool} first, then submit with `finish`.
[{k} tool call(s) left]
Not submitted: one look is rarely enough for this question. Use {tool} once more for the fact your answer still relies on, then submit with `finish`.
[{k} tool call(s) left]

Short answer (knowledge base)

appended to the question and the final request → Answer with a word or a short phrase, not a sentence; if the question offers alternatives, answer with one of them verbatim.

Reading note (corpus, except BC-Plus)

System: You are reading search results for a question. In at most two sentences, say what in them bears on the question -- quote the exact names, dates or numbers -- or say that nothing does. Do not answer the question; do not choose. Then, in one more sentence starting with 'Still to look up:', name the single fact or entity that must still be searched for before the question can be answered, or write 'Still to look up: nothing'.
User: Question: {question}
Results:
{the passages, first 2,500 characters}
appended to the observation → [What bears on the question: {note}]

Choice re-ask (WorldSense, Daily-Omni, OVBench)

System: You answer a multiple-choice question about the material shown and heard. Reply with the option letter only.
User: {the question's video and pictures, and a stretch of its sound}
Question: {question stem}
{the options, one line each as "A. . . . ", reversed or rotated}
Reply with the option letter only.

Direct answer (E-VQA)

System: You answer a question about a photograph from what you can see and what you know. Answer with a word or a short phrase, not a sentence.
User: {the photograph}
A knowledge-base entry that may be the photograph's subject (it may also be wrong):
[{title}] {entry text, first 600 characters}
Question: {question}

Figure 11: Prompts of the evaluation-time rules of Appendix D.2.

B THEORETICAL PROOF

This appendix proves the three propositions of §3; notation follows that section.

B.1 PROOF OF PROPOSITION 1

Proposition 1. *Beyond text, a few full-size native looks exhaust the window before the budget.*

Proof. Fix an environment $\mathcal{E}$ and a question x, and consider one context, opened with $\mathbf{p}(x)$ and grown by Eq. (1). A context holds at most one audio track, which a `watch` replaces, and everything else in it is appended and never removed. We split the length of Eq. (2) accordingly,

$$\ell(\mathbf{s}) = \ell^{\circ}(\mathbf{s}) + \tau\big(A(\mathbf{s})\big), \qquad \ell^{\circ}(\mathbf{s} \oplus g \oplus o) = \ell^{\circ}(\mathbf{s}) + \ell^{\circ}(g) + \ell^{\circ}(o) \;\geq\; \ell^{\circ}(\mathbf{s}), \tag{10}$$

where $A(\mathbf{s})$ is the audio track of $\mathbf{s}$ (empty off a stream) and ℓ° counts everything else; ℓ° is additive over $\oplus$ and hence non-decreasing from turn to turn. Write $\Delta\ell^{\circ}_t = \ell^{\circ}(\mathbf{s}_{t+1}) - \ell^{\circ}(\mathbf{s}_t) \geq 0$. Off a stream $\Delta\ell^{\circ}_t = c(a_t)$; on a stream, $\underline{c}_{\mathcal{E}}$ is the least $\Delta\ell^{\circ}_t$ of a full-size `watch`, i.e. its cost with the replacement of the track already applied. Every track, the preview's included, lasts at most 30 s, so $\tau(A(\mathbf{s})) \leq 750$, with equality right after a full-size `watch`, whose clip is the full 30 s. Let $t_1 < \cdots < t_n$ be the turns of the full-size looks made in the context. Telescoping ℓ° and dropping the other turns, whose $\Delta\ell^{\circ}_t$ are non-negative, gives right after the n-th look

$$\begin{aligned} \ell(\mathbf{s}_{t_n+1}) \;&=\; \ell^{\circ}\big(\mathbf{p}(x)\big) + \sum\nolimits_{t \leq t_n} \Delta\ell^{\circ}_t + \tau\big(A(\mathbf{s}_{t_n+1})\big) \\ &\geq\; \ell^{\circ}\big(\mathbf{p}(x)\big) + \sum\nolimits_{j=1}^{n} \Delta\ell^{\circ}_{t_j} + \tau\big(A(\mathbf{p}(x))\big) \;\geq\; \ell\big(\mathbf{p}(x)\big) + n\,\underline{c}_{\mathcal{E}}, \end{aligned} \tag{11}$$

where the middle step uses $\tau(A(\mathbf{s}_{t_n+1})) = 750 \geq \tau(A(\mathbf{p}(x)))$ on a stream and $0 = 0$ elsewhere. The model reads a context of at most W tokens, so $\ell(\mathbf{s}_{t_n+1}) \leq W$ and

$$n \;\leq\; \frac{W - \ell\big(\mathbf{p}(x)\big)}{\underline{c}_{\mathcal{E}}} \quad\Longrightarrow\quad n \;\leq\; n_{\text{fill}}(x), \tag{12}$$

as n is an integer. Whenever $n_{\text{fill}}(x) < K$, i.e. $K\,\underline{c}_{\mathcal{E}} > W - \ell(\mathbf{p}(x))$, an episode that keeps looking at full size must stop, or compact, with $K - n_{\text{fill}}(x) \geq 1$ calls unspent. The bound carries over to every later context: by Eq. (6) a successor opens with $\mathbf{p}(x) \oplus h_k \oplus \mathbf{b}_k$, whose length is at least $\ell(\mathbf{p}(x))$, so it too holds at most $n_{\text{fill}}(x)$ full-size looks, and an episode that spends all K calls on them spans $S \geq \lceil K/n_{\text{fill}}(x) \rceil$ contexts. Evaluating both sides on the observations behind Table 4 – the smallest full-size cost of each environment kind against the shortest prompt of that kind – gives $\underline{c}_{\mathcal{E}} = 1{,}786$ tokens for a knowledge-base search, $1{,}470$ for a three-page read and $1{,}219$ for a 180 s watch (its $1{,}969$ tokens less the 750-token clip that replaces the preview's), against shortest prompts of 928, 401 and $2{,}266$ tokens, hence $n_{\text{fill}} = 4$, 5 and 4, all below $K = 8$: one context spends at most $5/8$ of the budget, at least three calls stay unspent, and $S \geq 2$. A text search (700 tokens, prompt 388) gives $n_{\text{fill}} = 11 \geq K$, which is why the proposition excludes text. In summary, beyond text the window, not the budget, is the binding constraint of a native agent, and compaction is what lets it spend the budget it is given. □

B.2 PROOF OF PROPOSITION 2

Proposition 2. *Addressed memory lets a compacted agent re-open raw evidence a summary drops.*

Proof. Let Y be the answer, a discrete random variable jointly distributed with the episode, and let segment k close with context $\mathbf{s}$. A summary-only compaction hands the successor $h_k = \chi(\mathbf{s})$ for some, possibly stochastic, map χ. Since h_k depends on Y only through $\mathbf{s}$, $Y \to \mathbf{s} \to h_k$ is a Markov chain, $I(Y; \mathbf{s}, h_k) = I(Y; \mathbf{s})$, and the chain rule gives

$$I(Y; \mathbf{s}) - I(Y; h_k) \;=\; I(Y; \mathbf{s} \mid h_k) \;\geq\; 0, \tag{13}$$

the data-processing inequality, whose gap $I(Y; \mathbf{s} \mid h_k)$ is exactly what the summary dropped about the answer. Now let $o = \mathcal{E}(a)$ be a media observation of $\mathbf{s}$ whose unit f is rendered in $\mathbf{b}_k$, so that $\text{addr}(f) = a$ is in the successor's context. The address records the call's arguments (every `watch`

and `read_page` in full, a search query up to 120 characters), and the environment is deterministic and serves an identical media call again in a later segment, so

$$\mathcal{E}\big(\,\mathrm{addr}(f)\big) \;=\; \mathcal{E}(a) \;=\; o. \tag{14}$$

While a call remains ($N_t < K$), re-issuing the address therefore appends o itself, not a description of it. The successor then holds $(h_k, \mathbf{b}_k, o)$, and by the chain rule and the non-negativity of mutual information,

$$I(Y; h_k, \mathbf{b}_k, o) \;\geq\; I(Y; h_k, o) \;=\; I(Y; h_k) + I(Y; o \mid h_k) \;\geq\; \max\big\{I(Y; h_k),\, I(Y; o)\big\}. \tag{15}$$

The addressed successor thus recovers the raw observation, and with it the information $I(Y; o \mid h_k)$ that the summary dropped about o; the inequality over $I(Y; h_k)$ is strict exactly when $I(Y; o \mid h_k) > 0$, i.e. when the summary lost something about Y that o shows. The guarantee does not depend on χ: whatever the hand-over keeps, the successor knows at least what the raw observation tells. The gain in information is a gain in accuracy. By Fano's inequality, any answer $\hat{y}$ computed from a context $\mathbf{c}$ errs with probability

$$P_e(\mathbf{c}) \;\geq\; \frac{H(Y \mid \mathbf{c}) - 1}{\log |\mathcal{Y}|} \;=\; \frac{H(Y) - I(Y; \mathbf{c}) - 1}{\log |\mathcal{Y}|}, \tag{16}$$

with $\mathcal{Y}$ the answer's range, so a summary-only successor stays above the floor set by $I(Y; h_k)$, while re-opening o lowers that floor by $I(Y; o \mid h_k)/\log|\mathcal{Y}|$. A text unit needs no re-issue: its passage is stored verbatim and rendered up to 360 characters a line. In summary, an address turns what compaction would lose into one call away. □

B.3 Proof of Proposition 3

Proposition 3. *A moving batch baseline yields advantages even from one rollout per prompt.*

Proof. At $\theta = \theta_{\mathrm{old}}$ every ratio is $\rho_i = 1$, so neither clip in Eq. (9) is active (for $\hat{A}_u < 0$, $\zeta \hat{A}_u < \hat{A}_u$), and the policy-gradient part of the update is

$$\nabla_\theta \mathcal{J}\big|_{\theta_{\mathrm{old}}} = \frac{1}{n_{\mathcal{B}}} \sum_{(u,k)\in\mathcal{B}} \hat{A}_u \sum_{i \in \mathcal{I}_{u,k}} \nabla_\theta \log \pi_\theta\big(w_i^{u,k} \mid \mathbf{w}_{<i}^{u,k}\big) \;-\; \beta\, \nabla_\theta \mathrm{KL}, \tag{17}$$

which carries a learning signal only through non-zero advantages. A group baseline normalises each rollout against the rollouts $G(u)$ of its prompt,

$$\hat{A}_u^{\mathrm{G}} = \frac{R(u) - \mathrm{mean}_{u' \in G(u)}\, R(u')}{\mathrm{std}_{u' \in G(u)}\, R(u') + \delta} \;=\; 0 \qquad \text{whenever } |G(u)| = 1, \tag{18}$$

since with one rollout the group mean is $R(u)$ itself; the group-relative advantage then vanishes for every u at every step, and an implementation that falls back to the raw reward $R(u)$ has no baseline at all: a batch that fails uniformly ($r = 0$) again gives $\hat{A}_u = 0$, and one that succeeds uniformly is pushed up without contrast. Under Eq. (8), the baseline of kind d after its m-th batch is

$$\mu_d \;=\; \gamma^{m-1} \bar{R}_1 + (1-\gamma) \sum_{j=2}^{m} \gamma^{m-j} \bar{R}_j, \tag{19}$$

with $\bar{R}_j$ the mean reward of that kind's j-th batch: a convex combination, as the weights sum to one, of the history of batches drawn from the same environment. Consider a batch of kind d after the first one of that kind, with running statistics (μ_d^-, v_d^-) before it. If the rewards differ within the batch, $\hat{A}_u$ is, before clipping, a strictly increasing affine function of $R(u)$, and the clip, being monotone, keeps the order: rollouts are ranked and pushed apart. If every rollout scores alike, $R(u) = r$, then $\bar{R} = r$, $V = 0$, and

$$R(u) - \mu_d \;=\; r - \big(\gamma \mu_d^- + (1-\gamma) r\big) \;=\; \gamma\,\big(r - \mu_d^-\big), \tag{20}$$

so $\hat{A}_u \neq 0$ whenever the batch departs from the running mean of its kind, and it points toward that history: a uniformly failing batch is pushed down, a uniformly successful one up. Without the moving average ($\gamma = 0$) the same batch would again give $\hat{A}_u = 0$. Keying (μ_d, σ_d^2) by kind makes μ_d^- the history of batches drawn from the same environment, so the sign of $r - \mu_d^-$ compares like with like. In summary, the moving batch baseline supplies the contrast that a group of one cannot, and every batch that departs from its kind's history moves the policy. □

C OMNIPLEX-RL ALGORITHM DETAILS

We present the full workflow of OmniPlex-RL in Algorithm 1 (notation of §3), in three phases: **(1) Native Interaction.** Given question x, the policy π_θ issues one call per turn to the environment $\mathcal{E}$ and reads each observation natively, as frames, pages, pictures and sound, within a window W and budgets $T_{\max}$, K (Eq. (1)). **(2) Compaction into Hypergraph Memory.** When a segment's context fills or its calls reach C (Eq. (3)), the policy re-reads each media observation into a note, every observation becomes an addressed hyperedge of $\mathcal{G}_k$, and the successor opens from the prompt, the hand-over and the read-back $\mathbf{b}_k$ (Eqs. (4)–(6)). **(3) Single-rollout RL.** Each prompt of a single-kind batch yields one rollout; its terminal reward is compared with the moving baseline of the kind, and every segment is trained on its own context under the dual-clip objective (Eqs. (7)–(9)).

Algorithm 1 OmniPlex-RL: Native Agent, Hypergraph Compaction and Single-rollout RL

Require: Question x, environment $\mathcal{E}$ with tools $\mathcal{T}$, policy π_θ, window W, budgets $T_{\max}, K, C$, threshold η
Ensure: Answer $\hat{y}$ (inference); updated parameters θ (training)

// 1: Native Interaction within a Segment
Initialize $\mathbf{s}_1 \leftarrow \mathbf{p}(x)$, $k \leftarrow 1$, $t_1 \leftarrow 1$, $N \leftarrow 0$, $N_{\text{rep}} \leftarrow 0$
for $t = 1$ to $T_{\max}$ **do**
  Generate response and call: $g_t \sim \pi_\theta(\cdot \mid \mathbf{s}_t)$, $a_t \leftarrow \text{call}(g_t)$
  if $a_t =$ `finish` **then**
    return $\hat{y} \leftarrow$ the answer submitted by a_t
  else if $a_t \notin \mathcal{T}$ **then**
    Refuse without charge: $o_t \leftarrow$ a notice
  else if a_t repeats an earlier call and is not a media re-open after a hand-over **then**
    Refuse and charge: $o_t \leftarrow$ a notice, $N \leftarrow N + 1$, $N_{\text{rep}} \leftarrow N_{\text{rep}} + 1$
  else
    Observe natively: $o_t \leftarrow \mathcal{E}(a_t)$, $N \leftarrow N + 1$ ▷ N_t in Eq. (3)
  end if
  Append: $\mathbf{s}_{t+1} \leftarrow \mathbf{s}_t \oplus g_t \oplus o_t$ ▷ cost $c(a_t) = \ell(\mathbf{s}_{t+1}) - \ell(\mathbf{s}_t)$
  // 2: Compaction into Hypergraph Memory
  if segment k closes by Eq. (3) **then**
    for each $t' \in \mathcal{O}_k$ (at most six) **do**
      Look again and note: $\tilde{z}_{t'} \sim \pi_\theta(\cdot \mid o_{t'}, \bar{x})$, $z_{t'} \leftarrow \text{keep}(\tilde{z}_{t'})$
    end for
    Add hyperedges $f_{t'} \leftarrow \text{Ext}(z_{t'}) \cup \mathcal{M}_{t'}$, $\text{addr}(f_{t'}) \leftarrow a_{t'}$ to $\mathcal{F}_k$ for all observations of segment k
    Hand over: $h_k \sim \pi_\theta(\cdot \mid \mathbf{s}_{t+1})$, sanitized by the loop; $\mathbf{q}_k \leftarrow \phi(x \oplus h_k)$
    Read back: $\mathbf{b}_k \leftarrow$ the κ_b best units of $\mathcal{G}_1, \ldots, \mathcal{G}_k$ under sc, with addresses
    Open segment $k{+}1$: $\mathbf{s}_{t+1} \leftarrow \mathbf{p}(x) \oplus h_k \oplus \mathbf{b}_k$, $t_{k+1} \leftarrow t + 1$, $k \leftarrow k + 1$
  end if
end for
Ask once for the answer and **return** $\hat{y}$ ▷ the episode ended without `finish`
// 3: Single-rollout RL over Compacted Segments
for each training step **do**
  Draw a batch $\mathcal{B}_x$ of prompts of one kind d; roll out *once* per prompt by phases 1–2
  Reward each rollout u: $R(u) = \mathbf{1}[N_{\mathcal{T}}(u) > 0]\,\big(\text{F1}(\hat{y}_u, \mathcal{Y}^\star_u) - \min(\lambda N_{\text{rep}}(u), \bar{\lambda})\big)$
  Update kind statistics: $\mu_d \leftarrow \gamma\mu_d + (1-\gamma)\bar{R}$, $\sigma_d^2 \leftarrow \gamma\sigma_d^2 + (1-\gamma)V$
  Advantage: $\hat{A}_u = \text{clip}\big((R(u) - \mu_d)/(\sigma_d + \delta), -\bar{A}, \bar{A}\big)$ on $\mathcal{I}_{u,k}$ of every segment $\mathbf{w}^{u,k}$ of u
  Update policy: ascend $\mathcal{J}(\theta)$ of Eq. (9) over mini-batches $\mathcal{B}$ of segments, where
    $\psi_i = \max\big(\min(\rho_i \hat{A}_u, \text{clip}(\rho_i, 1{-}\epsilon, 1{+}\epsilon)\hat{A}_u),\, \zeta \min(\hat{A}_u, 0)\big)$, $\rho_i = \frac{\pi_\theta(w_i \mid \mathbf{w}^{u,k}_{<i})}{\pi_{\theta_{\text{old}}}(w_i \mid \mathbf{w}^{u,k}_{<i})}$
end for

Complexity Analysis. First, every context is bounded by the window, $\ell(\mathbf{s}_t) \le W$, so a turn costs at most one forward pass over W tokens and an episode $O(T_{\max} W)$, however much media it inspects; with $S_u \le T_{\max}$ segments per rollout, context never grows with episode length. Second, compaction issues at most six note calls per segment, each over one observation and $\bar{x}$; the extractor is local, and read-back scores the entities and hyperedges of each earlier graph with d_ϕ-dimensional vectors, $O(\sum_{j \le k}(|\mathcal{V}_j| + |\mathcal{F}_j|)\, d_\phi)$ per hand-over. Third, one rollout per prompt instead of a group of G cuts rollout cost by a factor of G, and a step trains $\sum_u \sum_k |\mathbf{w}^{u,k}| \le |\mathcal{B}_x|\, T_{\max} W$ tokens; rollouts of a batch run in parallel.

D BENCHMARK DETAIL

This appendix expands §4.1 into a self-contained reference. We propose no new benchmark: all twelve suites are public and reused with their own questions and answers. Throughout, x is a question, $\mathcal{Y}^\star$ its gold answers (aliases included) and $\hat{y}$ the submitted answer.

D.1 DATASET DETAILS

Table 6 lists the suites: 128 evaluation items each (1,536 in total), with the 2,400 training prompts drawn from the seven in-distribution suites only. **Text** suites query a corpus with `search`; **knowledge-base** suites query a CLIP ViT-L/14 index whose image-backed entries each return an article *and* its picture; the question's own photograph is part of $\mathbf{p}(x)$ and is never retrieved. **Document** suites expose `search_pages`, which ranks pages by BM25 and returns page numbers only, and `read_page`, which returns rendered pages; **omni** suites expose `watch`. Splits are drawn by container – LongDocURL by document, the omni suites by whole video – so no document or stream is shared between training and evaluation; MMLB-Doc keeps its answerable questions whose gold pages exist. Each E-VQA question shows an iNaturalist photograph of its taxon.

Table 6: The twelve suites. *Environment*: corpus, knowledge base, or the distinct documents or streams of the evaluation items with their median length. †: held out of training.

Group	Suite	Train	Eval	Environment	Answer
Text	BC-Plus	225	128	corpus of 100,195 web documents	span
	HotpotQA	225	128	Wikipedia-2018 passage corpus	span
	MuSiQue†	–	128	Wikipedia-2018 passage corpus	span
MM	E-VQA	450	128	knowledge base of 7,605 entries	span
	InfoSeek†	–	128	knowledge base of 6,589 entries	span
	LongDocURL	600	128	39 documents, median 84 pages	span / number / list
	MMLB-Doc†	–	128	79 documents, median 33 pages	number / list / span
Omni	OVBench	166	128	110 streams, median 5.1 min	4-way choice
	Daily-Omni	215	128	109 streams, median 30 s	4-way choice
	FineVideo	519	128	114 streams, median 5.2 min	span
	LVOBench†	–	128	97 streams, median 32 min	4-way choice
	WorldSense†	–	128	125 streams, median 1.6 min	3/4-way choice

D.2 EVALUATION DETAILS

Scores use no LLM judge. With $\mathrm{tok}(\cdot)$ lower-casing, folding to ASCII, dropping thousands separators, quotes and punctuation, and splitting on white space, the score of an answer is

$$\mathrm{F1}(\hat{y}, \mathcal{Y}^\star) = \max_{y \in \mathcal{Y}^\star} \frac{2\,|\,\mathrm{tok}(\hat{y}) \cap \mathrm{tok}(y)\,|}{|\,\mathrm{tok}(\hat{y})\,| + |\,\mathrm{tok}(y)\,|}, \tag{21}$$

with $\cap$ the multiset intersection; on a choice question the option letter is recovered first, making Eq. (21) exact match. It is also the task term of the reward (§3.3). A table cell is the mean over 128 items, averaged over three evaluation runs with half the range as $\pm$; a group average is the mean of its suites. The rules below run at evaluation only, for OmniPlex-RL in Table 2 and every column of Table 3; their prompts are in Figure 11. *Look-up* (knowledge base, corpus): before the first turn the knowledge base is searched by the photograph and a corpus by the question, five results each, given as the first observation and charged as one call. *Grounded finish* (document, corpus): while calls remain, an answer is declined, at most three times, until two `read_page` calls or two searches (look-up included); the loop then reads up to three surfaced unread pages (or searches pages by the question), or on a corpus searches the proposed answer. *Reading note* (corpus except BC-Plus): after each search the policy notes, in at most 150 tokens, what the first $2{,}500$ characters bear on the question and what remains; the note joins the passages in the memory. *Short answer* (knowledge base): a word or a short phrase is asked for. *Option votes* (WorldSense, Daily-Omni, OVBench): the choice is re-asked in fresh contexts holding the question's media, the sound last heard and re-ordered options – reversed and rotated once on WorldSense, reversed and every rotation otherwise, OVBench voters hearing three evenly spaced 30 s pieces; the most frequent mapped-back answer with two or more votes replaces the episode's (ties to the earliest vote, the episode's first). *Photograph answer* (E-VQA): the policy answers from the photograph in at most 32 tokens, with the first retrieved entry (title, first 600 characters) as a hint.

E BASELINE DETAIL

This appendix expands the columns of Tables 2 and 3 into a per-method specification: for the reference backbone, the two training-free agents, the five trained omni agents, the six ablations and OmniPlex-RL itself we report (i) *Overview* (the published mechanism and its carrier), (ii) *Port* (how it runs in the shared environment and what training changes) and (iii) *Hyperparameters*; Table 7 summarises all fifteen columns. With π_0 the untrained Qwen2.5-Omni-3B thinker (Xu et al., 2025) and F1 Eq. (21), every column shares π_0, the suites of Appendix D.1, the tools $\mathcal{T}$ and their per-call limits, W=8192 (media included), $T_{\max}$=16, K=8, L=320 (§3.1), the answer-format line and the scorer; a method's own tool is mapped onto $\mathcal{T}$ (a zoom, clip or retrieval becomes `watch` on a stream, `read_page` on a document and `search` on a corpus). Every trained column is a LoRA (r=64, α=128) on the text decoder of π_0 (vision and audio towers frozen), trained on the same 2,400 prompts and sampler at learning rate 2×10^{-5}, rollout temperature 0.5, with the interaction floor and repeat penalty of Eq. (7) and the early-stopping rule of §4.1, and evaluated in the loop it was trained in.

Table 7: Design comparison of every column of Tables 2 and 3. *Carrier*: what a method keeps across its context, with ✓/✗ for whether it compacts within a rollout. *Loop*: the scaffold it is evaluated (and trained) in. *Update*: what is optimised and its advantage; *Reward*: the training signal. "–" marks a column that is not trained.

Method	Carrier	Loop	Update	Reward
Qwen2.5-Omni-3B	none ✗	one turn, no call	–	–
OmniAgent-AG	raw video ✗	text planner, perception self-calls	–	–
OmniPlex	native replay ✓	OmniPlex loop, π_0	–	–
OmniRAG-Agent	frames + audio ✗	own: think, one retrieval a round	batch baseline	F1
Orchestra-o1	sub-agent messages ✗	shared linear loop	DA-GRPO	rubric 0.1/0.1/0.2/0.6
OmniAgent-RL	textual memory ✓	shared linear loop	TAURA, clip-higher	F1
Light-Omni	script + latent ✓	shared linear loop	reward-weighted	F1
OmniReasoner	time spans ✗	own: answer or one zoom	batch baseline	F1
w/o compaction	none ✗	OmniPlex loop, one segment	(full checkpoint)	–
w/o hypergraph mem.	hand-over only ✓	OmniPlex loop, no read-back	(full checkpoint)	–
w/o coordinates	unaddressed graph ✓	OmniPlex loop, no addresses	(full checkpoint)	–
w/o segment credit	native replay ✓	OmniPlex loop	one concatenated sample	Eq. (7)
w/o batch baseline	native replay ✓	OmniPlex loop	raw reward	Eq. (7)
w/o interact. floor	native replay ✓	OmniPlex loop	batch baseline	Eq. (7), no $\mathbf{1}[\cdot]$
OmniPlex-RL	**native replay ✓**	**OmniPlex loop**	**per-kind batch baseline**	**Eq. (7)**

E.1 REFERENCE AND TRAINING-FREE COLUMNS

These columns fix the weights at π_0, separating what a loop contributes from what training contributes.

Qwen2.5-Omni-3B (Xu et al., 2025). (i) *Overview.* The bare backbone, the floor of the paradigm. Carrier: none. (ii) *Port.* One chat-template turn from the question text alone under an answer-only system prompt; no call is made, so it bounds what parametric knowledge answers without looking. (iii) *Hyperparameters.* π_0, W=8192, L=320, zero calls.

OmniAgent-AG (Tao et al., 2026a). (i) *Overview.* Audio-guided active perception: a text-only "brain" plans in a Think–Act–Observe–Reflect loop, listens first and then looks, and never receives pixels or sound itself. Carrier: raw video, no compaction. (ii) *Port.* A self-calling port on π_0 that transcribes the paper's system, tool and reflection prompts: a coarse first pass captions the preview (audio caption, event list, visual summary; or a description of the question's photograph), the planner issues exactly one call per step, a perception self-call of the same backbone examines each result in one of seven audio or video modes (from `asr` to `video_clip`) and returns text, and a reflection follows every step. (iii) *Hyperparameters.* π_0; one call per step within K=8; self-calls share L=320.

OmniPlex (untrained). (i) *Overview.* Our loop with the omni-modal hypergraph memory of §3.2, run on the frozen backbone. Carrier: native replay, compacting. (ii) *Port.* Identical to OmniPlex-RL (§E.4) except that θ=θ_0, with the look-up, grounded-finish, reading-note and short-answer rules of Appendix D.2; the gap to OmniPlex-RL is what training adds. (iii) *Hyperparameters.* §E.4.

E.2 TRAINED OMNI AGENTS

Each agent is ported by its defining mechanism: OmniRAG-Agent and OmniReasoner define a scaffold, trained in it under the single-rollout recipe of §E.4; Orchestra-o1, OmniAgent-RL and Light-Omni define a learning recipe, trained in the shared linear loop, where every observation stays in the context and the loop compacts nothing.

OmniRAG-Agent (Zhu et al., 2026). (i) *Overview.* End-to-end agentic omni-modal RAG for long audio-video: the agent reasons inside `<think>`, issues one retrieval per round, and keeps every retrieved frame, clip, page and passage. Carrier: frames + audio, no compaction. (ii) *Port.* Its round protocol joins the shared system prompt: think about what is shown and missing, issue one call, never fetch twice, and summarise the evidence with its sources before `finish`. (iii) *Hyperparameters.* One rollout per prompt, 8 prompts per step, reward F1 with the batch baseline.

Orchestra-o1 (Zhang et al., 2026b). (i) *Overview.* A text orchestrator dispatches rounds of independent perception sub-tasks to fresh-context sub-agents and reads their messages; only its decisions are trained, by DA-GRPO against an LLM judge. Carrier: sub-agent messages, no compaction. (ii) *Port.* Its DA-GRPO objective, a group advantage over rollouts of one prompt, with the paper's rubric scored by rule instead of a judge: $R = 0.1\,\mathbb{I}[\text{submitted}] + 0.1\,\mathbb{I}[\text{called}] + 0.2\,n_{\text{ok}}/(n_{\text{ok}}{+}n_{\text{bad}}) + 0.6\,\text{F1}$, where n_{bad} counts tool errors and refused repeats. (iii) *Hyperparameters.* Clip 0.2, KL 0.01, token-mean loss.

OmniAgent-RL (Xing et al., 2026). (i) *Overview.* Native active perception trained by TAURA, which rescales each turn's advantage by its token entropy so pivotal, uncertain turns carry more credit. Carrier: textual memory, compacting. (ii) *Port.* With $\bar{H}_{u,j}$ the mean token entropy of turn j, GRPO advantages become $\hat{A}_{u,j} = \hat{A}_u\,\bar{H}_{u,j}/\,\text{mean}_{u',j'}\,\bar{H}_{u',j'}$ over a prompt's group, under a DAPO-style token-level loss. (iii) *Hyperparameters.* Clip $[0.8, 1.28]$, no KL term, reward F1.

Light-Omni (Nie et al., 2026). (i) *Overview.* Memorises the stream in advance into semantic facts and an episodic script consolidated into a global state, answering from that state; the paper trains it by supervised fine-tuning. Carrier: script + latent, compacting. (ii) *Port.* Supervision on the policy's own rollouts, each weighted by its reward: one rollout per prompt with $\hat{A}_u = R(u)$. (iii) *Hyperparameters.* Clip 0.2, KL 0.01, reward F1.

OmniReasoner (Chen et al., 2026). (i) *Overview.* Thinking with long audio-video through native tool use: from a global preview the model either answers or zooms into one interval, with plain-text time anchors. Carrier: time spans, no compaction. (ii) *Port.* The paper's tag protocol: `<think>` followed by exactly one of `<answer>` or a zoom, the zoom being `<interval>[s,e]` (one `watch`, $\leq$180 s, anchors every 2 s, at most 16), `<pages>` (one `read_page`) or `<query>` (one `search`); the zoomed observation returns with the paper's answer prompt. (iii) *Hyperparameters.* One zoom; single-rollout recipe of §E.4.

E.3 ABLATIONS

Inference-time ablations evaluate the OmniPlex-RL checkpoint with one switch changed in the loop: **w/o compaction** sets $C{=}\infty$ and removes the hand-over and memory, so one segment spans the rollout; **w/o hypergraph memory** keeps the segments and the hand-over h_k but drops $\mathbf{b}_k$ from Eq. (6); **w/o coordinates** keeps $\mathcal{G}_k$ and $\mathbf{b}_k$ but renders units without $\text{addr}(f)$. *Training-time* ablations are separate runs from π_0 on the same data, loop and call budget, changing one term of §3.3: **w/o segment-level credit** trains the folded rollout as one concatenated sequence; **w/o batch baseline** sets $\hat{A}_u = R(u)$; **w/o interaction floor** drops $\mathbf{1}[N_{\mathcal{T}}(u) > 0]$.

E.4 OMNIPLEX-RL

(i) *Overview.* The native agent of §3.1, compacted into the addressed omni-modal hypergraph of §3.2 and trained by single-rollout RL over segments (§3.3; Algorithm 1). (ii) *Port.* None; evaluation rules are in Appendix D.2. (iii) *Hyperparameters.* Segments: $C{=}4$, $\eta{=}0.9$, at most six notes per fold. Memory: Ext is spaCy `en_core_web_sm`, ϕ is Qwen3-Embedding-0.6B, $\kappa_v{=}6$, $\kappa_f{=}4$, $\nu{=}0.6$, $\kappa_b{=}30$ ($\leq$1,200 characters, four addresses). RL: one rollout per prompt, 8 prompts per step, up to 80 steps, $\gamma{=}0.9$, $\bar{A}{=}5$, $\delta{=}10^{-6}$, $\epsilon{=}0.2$, $\zeta{=}3$, $\beta{=}0.01$, $\lambda{=}0.02$, $\bar{\lambda}{=}0.1$.

F EXPERIMENT DETAIL

This appendix details the setting and results of each research question of §4.2–§4.9. Evaluation numbers come from the three runs of each column behind Tables 2 and 3 on the 1,536 questions (12 suites of 128), scored as in Appendix D.2; training numbers come from each trained column's logs up to its evaluated checkpoint.

F.1 MAIN RESULTS AND ABLATIONS (RQ1, RQ2)

Setting. Each column is evaluated three times on the same items; a cell of Tables 2 and 3 is the mean of the three suite scores with half their range as $\pm$, and a group average is the mean of its suites, so every suite weighs the same whatever its environment. All columns share the backbone, the window W, the call budget K and the turn limit $T_{\max}$ of §4.1; the trained columns share the training set.

Main results. OmniPlex-RL is first on all twelve suites, while the strongest baseline changes with the suite: the untrained OmniPlex on seven (BC-Plus, HotpotQA, MuSiQue, LongDocURL, MMLB-Doc, LVOBench, WorldSense), OmniAgent-RL on InfoSeek and OVBench, OmniRAG-Agent on Daily-Omni and FineVideo, and OmniReasoner on E-VQA. The margin is largest on HotpotQA ($+4.77$), FineVideo ($+3.27$), LVOBench ($+3.13$) and MMLB-Doc ($+2.60$), and it is $+2.56$ on Text, $+6.04$ on MM, $+3.31$ on Omni and $+4.51$ on All. Held out it averages $+2.15$ over five suites, in distribution $+2.21$ over seven. Untrained, OmniPlex (22.5) already exceeds every trained omni agent (OmniRAG-Agent 21.3, Orchestra-o1 21.2, OmniAgent-RL 21.1, Light-Omni 19.7, OmniReasoner 18.0); single-rollout RL then adds $+2.6$ on Text, $+6.8$ on MM and $+3.9$ on Omni.

Ablations. Each of the six ablations lowers Avg. All and every group average. At inference, removing compaction costs most on LongDocURL (-9.67), LVOBench (-3.93) and MMLB-Doc (-3.77); removing the hypergraph memory on LongDocURL (-10.00) and MMLB-Doc (-6.33); removing the coordinates on HotpotQA (-4.83) and MuSiQue (-3.10). In training, dropping the interaction floor costs -9.60 on average, up to -25.17 on HotpotQA, -20.97 on InfoSeek and -17.97 on LongDocURL; dropping segment-level credit costs up to -15.63 on HotpotQA and -6.47 on FineVideo; dropping the batch baseline up to -8.57 on InfoSeek and -6.20 on MuSiQue.

F.2 WHAT ONE LOOK COSTS (RQ3)

Setting. A *media look* is a call that returns page images, a knowledge-base entry with its picture, or frames with sound; an episode *hits the window* when its context reaches $W{=}8{,}192$ tokens at some turn. Our loop logs both, so the measurement pools the ten columns it runs – OmniPlex, Orchestra-o1, OmniAgent-RL, Light-Omni, OmniPlex-RL and five of its ablations – over three runs: 46,080 episodes. Table 8 (left) evaluates Proposition 1 on the constants of Appendix B.1; (right) gives the measured hit rate by media looks.

Table 8: Proposition 1 per environment kind (left) and the measured share of episodes that reach the window, by media looks taken, with the number of episodes (right).

Kind	$\ell(\mathbf{p})$	$\underline{c}_{\mathcal{E}}$	n_{fill}	$\ell{+}K\underline{c}_{\mathcal{E}}$
Text	388	700	11	5,988
KB	928	1,786	4	15,216
Doc.	401	1,470	5	12,161
A/V	2,266	1,219	4	12,018

Kind	$\leq$2 looks	3–4 looks	5–6 looks	$\geq$7 looks
Text	0.0 (all 11,520 episodes)			
KB	0.0 (2,754)	2.7 (3,622)	69.1 (628)	94.2 (676)
Doc.	0.0 (6,145)	4.1 (933)	24.3 (428)	34.5 (174)
A/V	0.0 (17,408)	0.5 (217)	37.4 (492)	50.6 (1,083)

Results. Beyond text the bound $\ell(\mathbf{p})+K\underline{c}_{\mathcal{E}}$ exceeds W by 47–86%, while eight text searches stay at 73% of it. The measurement follows: with at most two media looks one episode in 26,307 reaches the window, and the rate climbs from bin to bin to 34.5–94.2% from seven on; no text episode ever reaches it. OmniPlex-RL takes 3.08 knowledge-base, 4.63 document and 1.34 stream calls per episode (the answer included), yet only 3.0%, 0.8% and 0.4% of its episodes reach the window: a segment closes after $C{=}4$ calls.

F.3 What Survives Compaction (RQ4)

Setting. A question counts as *folded* when the full system folds on it in at least two of its three runs (166 questions; 1,370 others); a column's F1 on a side is its three-run mean over those questions (Table 9, left). *Gold kept* is measured on every folded episode whose gold answer, or an alias, has at most six words and is not an option letter: it counts when the normalised gold string appears in the successor's opening context, split into the hand-over h_k and the read-back $\mathbf{b}_k$.

Table 9: Per column: F1 on the 166 folded and 1,370 other questions (RQ4), F1 on the stream questions by video length (136, 162, 174, 168 questions; RQ6) and OmniPlex-RL's paired-bootstrap margin in Avg. All (RQ6); underlined: best baseline.

	Fold (RQ4)		**Stream length, min (RQ6)**				**Margin (RQ6)**
Column	folded	other	<1	1–3	3–10	>10	Δ [95% CI]
OmniAgent-AG	11.70	20.10	41.42	33.64	21.29	17.75	+7.83 [6.19, 9.46]
OmniPlex	<u>15.74</u>	<u>23.33</u>	48.04	41.53	25.28	26.58	+4.52 [3.33, 5.69]
OmniRAG-Agent	14.99	22.05	48.77	<u>42.04</u>	<u>27.57</u>	25.34	+5.74 [4.56, 6.90]
Orchestra-o1	15.33	21.90	<u>49.02</u>	39.21	20.45	<u>27.17</u>	+5.84 [4.54, 7.09]
OmniAgent-RL	8.67	22.57	46.32	40.54	23.45	23.32	+5.96 [4.76, 7.27]
Light-Omni	15.32	20.22	47.55	37.08	17.14	23.21	+7.33 [6.08, 8.68]
OmniReasoner	9.02	19.12	39.52	32.61	21.17	18.19	+9.00 [7.63, 10.49]
w/o compaction	12.88	25.73	–	–	–	–	–
w/o memory	12.09	25.98	–	–	–	–	–
w/o coordinates	17.85	26.81	–	–	–	–	–
OmniPlex-RL	**21.85**	**27.65**	**54.66**	**43.85**	**30.47**	**28.46**	–

Results. OmniPlex-RL leads every column on both sides, by +6.11 on the folded questions and +4.32 on the others. The six other carriers of Table 2 stay within 8.67–15.33 on the folded questions, and the ablations show where the lead comes from: removing compaction, the memory or the coordinates costs 8.96, 9.75 and 4.00 there, against 1.92, 1.67 and 0.84 elsewhere. Over its 463 folded episodes, the gold answer reaches OmniPlex-RL's successor in 29.6% – 8.9% through the hand-over and 26.3% through the read-back – against 9.7% over 473 episodes without the memory and 27.5% over 466 without the coordinates; on documents (298 episodes) the rates are 40.6, 13.3 and 37.9%, the read-back alone carrying 35.6% against 13.1% for the hand-over. The read-backs are addressed throughout: 399 document read-backs carry 2.8 lines each, 70% of them notes on rendered pages (`read_page`) and the rest page-search hits; 46 knowledge-base read-backs carry 2.9 lines, from photograph and text searches in equal parts; 60 stream read-backs carry 2.9 lines, each a watched window of frames and sound; 104 text ones carry 2.2. Every line ends in the call that produced it, so the successor can re-open any of them.

F.4 Credit under a Single Rollout (RQ5)

Setting. A training step's environment kind is read from its call counters (knowledge base if any `search_by_image`, document if any `search_pages` or `read_page`, stream if any `watch` or returned frames or sound, text if only `search`); a step carries *no signal* when the largest and the smallest advantage of its batch are both zero. The train–rollout gap is the per-step mean of $|\text{log-ppl}_{\text{train}} - \text{log-ppl}_{\text{rollout}}|$ over the trained tokens, and ρ the Pearson correlation of their token probabilities under the two.

Results. OmniPlex-RL has no step without signal in any kind, as the per-kind moving baseline turns even a uniformly failing batch into a push away from that kind's history (Proposition 3); group baselines lose whole steps whenever every group scores alike, up to all of OmniReasoner's stream steps. Training each segment on its own context keeps the gap at 0.103 nats with ρ=0.824, against 2.027 nats and ρ=0.691 when the folded rollout is concatenated into one sample. The policy keeps looking throughout training (3.51 calls per rollout on average), whereas without the interaction floor it stops by step 27.

Table 10: Optimizer steps without signal (%), by the environment kind of the step's batch (–: no batch of that kind).

Column	Contrast from	Text	KB	Doc.	Stream	All
OmniReasoner	GRPO group	20.0	–	0.0	100.0	50.0
OmniAgent-RL	TAURA group	40.0	–	60.0	80.0	68.4
Light-Omni	GRPO group	44.4	50.0	0.0	7.7	21.1
Orchestra-o1	GRPO group	0.0	66.7	0.0	12.5	15.8
OmniRAG-Agent	GRPO group	0.0	0.0	0.0	0.0	0.0
w/o batch baseline	raw reward	28.6	0.0	0.0	0.0	7.8
OmniPlex-RL	**per-kind moving batch**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**

F.5 Robustness, Significance and Interaction Cost (RQ6, RQ7)

Setting. The 640 stream questions of the five omni suites are split by video length into four buckets (Table 9). The paired bootstrap draws, 1,000 times, each suite's 128 questions with replacement, recomputes Avg. All for OmniPlex-RL and the baseline on the same draw, and reports the 2.5th and 97.5th percentiles of the margin. For RQ7 each column is placed by its environment calls per episode, averaged over the twelve suites.

Results. OmniPlex-RL is first in every length bucket, by $+5.64$, $+1.81$, $+2.90$ and $+1.29$ over the best baseline of the bucket, and the best baseline changes from bucket to bucket (Orchestra-o1, OmniRAG-Agent, OmniRAG-Agent, Orchestra-o1). Every bootstrap interval lies above zero, the lowest bound being $+3.33$ against the untrained OmniPlex. The accuracy–calls frontier is formed by OmniReasoner (18.0 at 0.47 calls), OmniRAG-Agent (21.3 at 0.79) and OmniPlex-RL (27.0 at 2.80); every other column is dominated by one of them, and on knowledge bases and documents OmniPlex-RL reaches 20.5 with 3.85 calls where the untrained OmniPlex and Light-Omni reach 13.7 and 12.0 with 5.48 and 5.13.

F.6 Case Study (RQ8)

Setting. Daily-Omni #805 is a question on which OmniPlex-RL's answer is right in all three runs while no other column is right in more than one. Its opening view holds eight frames sampled at 0.0, 8.6, 17.1, 25.7, 34.3, 42.9, 51.4 and 60.0 s and the first 30 s of sound, the length the audio loader keeps. Table 11 aligns the clip's shots with its speech (faster-whisper large-v3 timestamps).

Table 11: Daily-Omni #805: shots and speech of the 60 s clip; options (A), (C), (D) mark the shots they describe.

Shot	On screen	Speech
0–13 s	Reporter on a Tokyo street	"Working long hours is a way of life in Japan … Karoshi is a term that means death by overwork." (0.0–8.5 s)
16–18 s	Commuter crowd	"Japan has some of the longest working hours in the world." (14.8–17.7 s)
19–23 s	Employees at desks (A)	"Nearly a quarter of Japanese companies have their employees work more than 80 hours of overtime per month." (18.0–23.5 s)
24–26 s	Lit office floors at night (A)	"Those extra hours are often unpaid." (23.6–25.7 s)
27–29 s	Men in dark suits on a platform (C)	"Today, Japan's corporate culture is driven by the so-called salary man." (25.7–29.8 s)
46–57 s	Screen of vacation days (D)	"… despite being entitled to 20 days off per year, Japanese workers left 10 of those days unused." (46.9–55.1 s)

Results. The question is decided by aligning two modalities: the overtime figures are spoken over the desks and the lit office floors (A), and the very next sentence names the "salary man" over the commuters on a platform (C). OmniPlex-RL holds both the speech and the 25.7 s frame of the office floors in one native context and ends on (A) in all three runs. The captioning pipeline of OmniAgent-AG dates the overtime passage correctly ("00:17–00:24") but answers (C) in all three runs, matching the word "salary" rather than the moment; the bare backbone, answering from the question text alone, also picks (C) in all three.

G Extended Related Work

G.1 Omnimodal Agents

Qwen2.5-Omni (Xu et al., 2025) and Qwen3.5-Omni (Team, 2026) provide native multimodal backbones, while OmniGAIA (Li et al., 2026f) and VideoGAIA (Zhang et al., 2026a) evaluate tool-mediated multimodal reasoning. Selective perception, developed through OmniAgent-AG (Tao et al., 2026a), OmniRAG-Agent (Zhu et al., 2026), OmniAgent-RL (Xing et al., 2026), and OmniReasoner (Chen et al., 2026), makes evidence acquisition adaptive; Orchestra-o1 (Zhang et al., 2026b) and ACE-Cap (Ma et al., 2026a) extend coordination across agents and perceptual roles. Sustaining this interaction requires persistent state: M3-Agent (Long et al., 2026) and AOP-Agent (Xu et al., 2026c) organize accumulated experience, Light-Omni (Nie et al., 2026) maintains script and latent states, Omni-Decision (Ma et al., 2026b) tracks evidence sufficiency, and CAM (Tian et al., 2026) combines graph retrieval with renewed audiovisual inspection. OmniPlex-RL contributes a unified native agent that sustains textual, visual, document, and audiovisual reasoning across context boundaries by preserving evidence relationships and access to source media.

G.2 Agent Context Management

Context management must reconcile limited capacity with future evidence needs. Context Compaction Theory (Tirmazi et al., 2026) formalizes retention limits; Knowledge Triage (Zerhoudi et al., 2026), COMPINT (Wang et al., 2026e), MIRAGE (Liu et al., 2026b), and *Correct Now, Insufficient Later* (Zhang, 2026) expose losses in constraints, grounding, or information needed for later updates, while *Harness Design* (Fan et al., 2026) tests the practical value of recovery. MemGPT (Packer et al., 2024) organizes hierarchical memory; ACON (Kang et al., 2026b) and SelfCompact (Li et al., 2026d) compress history, while Context-Folding (Sun et al., 2025) and AgentFold (Ye et al., 2025) fold trajectories. AdaCoM (Yi et al., 2026) adapts management to the agent, and TokenPilot (Xu et al., 2026b) accounts for caching costs. Recoverability further separates active context from stored history: MemexRL (Wang et al., 2026d), ARC (Dang et al., 2026), VISTA (Xu et al., 2026a), and Self-GC (Hao et al., 2026) preserve addressable content, while ACE (Liao et al., 2026), ACM (Li et al., 2026e), LongSeeker (Lu et al., 2026), and Scroll (Lin et al., 2026) make access and reconstruction adaptive. Their common premise is that a compact working context need not imply a lossy evidence store.

For multimodal histories, EgoMem (Yao et al., 2026b), Omni-SimpleMem (Liu et al., 2026a), and ViLoMem (Bo et al., 2026) model persistent multimodal experience; WorldMM (Yeo et al., 2026) and MementoGUI (Zeng et al., 2026) retain visual evidence. MAGIC-Video (Li et al., 2026c) structures temporal experience, whereas StreamMeCo (Wang et al., 2026a) and MemForest (Wang et al., 2026b) compress accumulated memory. PMMC (Sun et al., 2026) compiles prospective memory, V-Mem (Kang et al., 2026a) routes retrieval by modality, and DocMemo (Yao et al., 2026a) tracks document evidence. VimRAG (Wang et al., 2026c) and GraphMemix (Li et al., 2026b) structure visual and query-relevant evidence, whereas HyperSkill (Xu et al., 2026d) connects reusable skills. The reader-facing representation matters too: VERA (Su et al., 2026) retains native visual observations, while MEMO (Gao et al., 2026) allocates textual and visual carriers. These designs differ in whether memory preserves observations, derived summaries, or reusable procedures.

Learning must also respect changing contexts: MEM1 (Zhou et al., 2026a), MemAct (Zhang et al., 2026c), SUPO (Lu et al., 2025), and CompactionRL (Li et al., 2026g) train memory or compaction behaviors; ECHO (Xie et al., 2026) and ContextPilot (Pan et al., 2026) refine credit assignment, while MemoryWalker (J et al., 2026) addresses conditioning inconsistency. Our contribution couples hypergraphs binding observations to cross-modal evidence and executable media addresses with single-rollout RL that trains each segment on its rollout context using modality-specific reward baselines.