STaR-VL: Capability-Aligned Reinforcement Learning for Visual-Temporal Memory

Anonymous authors
Paper under double-blind review

Abstract

Vision-language models can recognize objects in individual images yet struggle to retain and compare observations across a changing scene. We present **STaR-VL**, a staged post-training framework for visual-temporal memory on a frozen Qwen2.5-VL backbone. Three LoRA adapters learn per-frame perception, symbolic temporal memory through a recurrent memory module, and transfer to visual histories, respectively; each stage combines supervised fine-tuning with a capability-aligned reinforcement-learning reward. Our contributions are a capability-aligned training framework and a controlled evaluation separating its gains from in-domain joint training, persistent-memory augmentation, and inference-time prompting. On SpaMEM, STaR-VL-3B reaches a V-score of 47.60, compared with 21.85 for the untrained backbone, 40.80 for a joint rank-48 LoRA, and 38.90 for an adapted ReMEmbR system on the same backbone. A structured memory-table prompt raises the untrained model to 22.21. The same trained checkpoints transfer to held-out benchmarks: the 7B model improves VSI-Bench from 33.3 to 46.4, with smaller gains on EgoSchema and TempCompass. Visual-history and symbolic-history controls reveal complementary effects of frame coverage and accessible scene state. These results support staged capability-aligned training for object recall, event detection, and cumulative memory over explicitly ordered observations, while identifying fine-grained grounding and temporal localization as remaining challenges.

1 Introduction

Visual-temporal memory requires a model to retain what it observed, update that representation as a scene changes, and retrieve relevant evidence for a later query. Accurate recognition in a single image provides only part of this capability. An object may disappear from view without leaving the scene, a relation may change between observations, and a cumulative query may require evidence distributed across several steps. These demands arise in embodied question answering and video understanding, where an answer depends on the observation history rather than the final frame alone.

Our evaluation on SpaMEM (Liao et al., 2026) compares per-frame perception, visual-history reasoning, and symbolic-history reasoning. General-purpose VLMs spanning several model families exhibit substantial differences across these input regimes. For example, Qwen3-VL-235B (Qwen Team, 2025) reaches 60.7 with symbolic histories but 24.4 with visual histories. Such comparisons identify a practical gap between information available in a scene and information a model can use from visual observations. We refer to this operational difference as the *visual–symbolic performance gap*. Ground-truth symbolic histories also provide denoised state and different history coverage, so the comparison motivates training and diagnostic controls rather than identifying a single causal bottleneck.

We propose **STaR-VL**, a capability-aligned post-training framework on a frozen Qwen2.5-VL backbone (Bai et al., 2025). The framework learns three LoRA adapters in sequence. The perception adapter establishes per-frame grounding; the memory adapter learns to summarize symbolic histories through a recurrent memory module; and the visual adapter adapts this initialized system to multi-image observations. Each stage combines supervised fine-tuning with reinforcement learning under a reward matched to its output capabilities. This decomposition offers a concrete way to learn temporal-memory behavior from structured histories before optimizing it under visual conditioning.

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.

An important evaluation question is how much of the improvement comes from task-specific data, and how much remains after equipping the base model with alternative training or memory mechanisms. We therefore compare STaR-VL with joint LoRA training and an adapted ReMEmbR persistent-memory system (Anwar et al., 2024) on the same Qwen2.5-VL-3B backbone. The untrained backbone scores 21.85, joint rank-48 LoRA reaches 40.80, ReMEmbR with memory and frames reaches 38.90, and STaR-VL reaches 47.60. In-domain training and external memory both recover substantial performance; the complete staged framework adds 6.80 and 8.70 points over these respective controls. These are comparisons between systems, with the contribution of training order alone remaining distinct from the joint effects of the memory module, adapter composition, and optimization recipe.

Additional controls clarify what the learned capability depends on. A structured prompt that explicitly constructs a temporal memory table raises the untrained backbone only from 21.85 to 22.21. Increasing STaR-VL's visual budget from three to sixteen frames raises V-score from 47.60 to 52.40, showing that accessible history also matters. The method's gains concentrate on object recall, event detection, and cumulative memory; fine-grained grounding and temporal localization remain weaker. Transfer beyond SpaMEM is positive on three held-out multi-image/video benchmarks, with the largest gain on VSI-Bench.

Our contributions are twofold: **(i)** a staged, capability-aligned training framework that combines perceptual grounding, recurrent symbolic memory, and visual adaptation; and **(ii)** a controlled empirical assessment of that framework against joint training, persistent memory, structured prompting, and history-coverage alternatives. Together, these contributions establish a practical post-training approach to visual-temporal memory and identify which improvements persist beyond task exposure or inference-time scaffolding.

2 Related Works

STaR-VL sits at the intersection of three lines of work. *(i) Embodied question answering and long-horizon embodied reasoning*: from the original EQA formulation (Das et al., 2018; Yu et al., 2019) through OpenEQA (Majumdar et al., 2024) and recent benchmarks emphasizing exploration, noisy queries, and adverse sensing (Cheng et al., 2024; Wu et al., 2024; Jiang et al., 2025; Zhao et al., 2025b; Park et al., 2025; Zhang et al., 2026), the field has moved beyond navigation-plus-answering toward agents that must maintain coherent intermediate evidence over long horizons. The closest line of work — exemplified by TIDEE (Sarch et al., 2022) for long-horizon scene-state reasoning, FindingDory (Yadav et al., 2025) for long-horizon memory in embodied agents, and SpaMEM (Liao et al., 2026) for fine-grained visual-temporal memory diagnostics — shares our focus on multi-step temporal reasoning under dynamic scene evolution. *(ii) Persistent memory and world models for embodied agents*: a parallel line argues that embodied intelligence depends on state construction rather than single-step recognition, exemplified by world-model-based language adaptation (Xiang et al., 2023; 2024; Yang et al., 2024; Anwar et al., 2024) and methods that make memory an explicit architectural primitive (Zhai et al., 2025; Hu et al., 2025; Wang et al., 2025a; Dong et al., 2025; Wang et al., 2026b; Dorbala & Manocha, 2026). Training-free in-context aggregation approaches such as Embodied VideoAgent (Fan et al., 2025) pursue the same goal through retrieval over an external persistent memory, whereas STaR-VL learns the memory interface through reinforcement-learning supervision. *(iii) Reinforcement learning for multimodal reasoning*: building on PPO (Schulman et al., 2017) and the verifiable-reward paradigm of DeepSeek-R1 (DeepSeek-AI, 2025; Shao et al., 2024), recent works transfer R1-style training to vision-language (Huang et al., 2025; Zhang et al., 2025a) and embodied (Zhai et al., 2024a; Zhao et al., 2025a; Kim et al., 2025; Song et al., 2025; Ye et al., 2025) models, often with progressive or staged post-training (Bengio et al., 2009; Chen et al., 2025; Wang et al., 2026a; Lin & Zheng, 2026; Zhou et al., 2026a).

Relative to these lines, STaR-VL studies staged acquisition of perceptual grounding, symbolic temporal memory, and visual adaptation. Capability-specific rewards match each stage's structured outputs. The same-backbone joint-training and persistent-memory controls in Sec. 4.1 evaluate the complete framework against alternative ways of improving visual-temporal memory. Appendix A provides an extended discussion. Memory construction is central to this distinction: STaR-VL trains recurrent compression jointly with symbolic-history reasoning, then adapts the resulting system to visual observations. This connects persistent state maintenance with capability-specific post-training across the transition from ground-truth symbolic observations to multi-image visual input.

3 STAR-VL

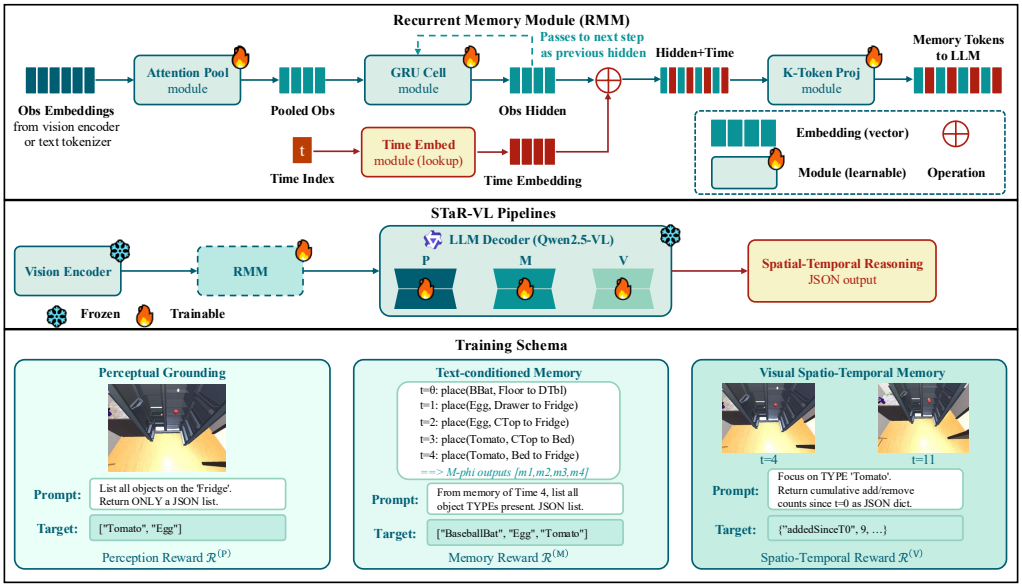

Figure 1: **STaR-VL overview.** Three LoRA adapters $\Delta_{\mathrm{P}}, \Delta_{\mathrm{M}}, \Delta_{\mathrm{V}}$ are acquired in dependency order on a frozen Qwen2.5-VL backbone, each trained with SFT followed by GRPO under a capability-aligned reward ($\mathcal{R}^{(\mathrm{P})}, \mathcal{R}^{(\mathrm{M})}, \mathcal{R}^{(\mathrm{V})}$). Top: the recurrent memory module (RMM) compresses observations into K memory tokens. Middle: inference pipeline. Bottom: per-stage input modality and target output.

We propose **STaR-VL**, a sequential adapter-acquisition framework on a frozen Qwen2.5-VL backbone. Three LoRA adapters $\Delta_{\mathrm{P}}, \Delta_{\mathrm{M}}, \Delta_{\mathrm{V}}$ are stacked on the language-model linear projections and acquired in dependency order, each isolating one capability: Δ_{P} targets per-frame perceptual grounding (localizing, identifying, quantifying objects), Δ_{M} targets temporal memory in the symbolic regime by replacing visual input with ground-truth symbolic observations routed through a recurrent memory module M_ϕ, and Δ_{V} transfers the symbolic temporal capability to multi-image RGB input under reference-anchored regularization. Joint optimization across capabilities can encounter gradient interference (Laitinen Imanov, 2026) and catastrophic forgetting once visual-temporal training begins (Kirkpatrick et al., 2017; Zhai et al., 2024b); sequential acquisition with reference anchoring is our approach to managing these effects. Each adapter is trained with SFT followed by GRPO over its own *capability-aligned* reward, defined below.

Further discussion of the dependency order is provided in Appendix E.

3.1 CAPABILITY-ALIGNED REWARDS

We instantiate three adapter-specific rewards $\mathcal{R}^{(\mathrm{P})}, \mathcal{R}^{(\mathrm{M})}, \mathcal{R}^{(\mathrm{V})}$, each scoring the capability targeted by its adapter. All three are verifiable, structure-aware rewards keyed by output type, drawn from a shared library: F1 over set/dict, IoU for grounding, $1-\mathrm{NED}$ for trajectories, exact match for counts, and normalized error for depth. $\mathcal{R}^{(\mathrm{V})}$ inherits the scoring branches of $\mathcal{R}^{(\mathrm{M})}$ so that visual adaptation retains the target structures learned in the symbolic stage. Task rewards have different numerical ranges and sparsity (e.g., set F1 versus bounding-box IoU). Stage-specific objectives and baselines accommodate these differences while retaining a shared structured scoring interface. A format reward $R_{\mathrm{fmt}} \in \{0, 1\}$ enforces structural validity (DeepSeek-AI, 2025), giving $R^{(s)}_{\mathrm{total}} = \mathcal{R}^{(s)} + \lambda_{\mathrm{fmt}} R_{\mathrm{fmt}}$ for $s \in \{\mathrm{P}, \mathrm{M}, \mathrm{V}\}$ (Appendix C.1, Eqs. (9)–(8)).

Concretely, dictionary outputs are scored over their key sets, trajectories by normalized edit distance, and counts by exact match. The symbolic and visual memory stages share the list, dictionary-key, and bounding-box scoring branches.

3.2 PER-ADAPTER RL OBJECTIVES

The three adapters share a common SFT-then-GRPO recipe but differ in advantage estimation and reference regularization, chosen to match each capability's variance profile (full forms in Appendix C.2). Advantages are normalized group-relatively for Δ_{P} where rewards are dense; Δ_{M} and Δ_{V} use a fixed-baseline (with clipping) and a per-task EMA baseline respectively, to handle the heavier-tailed reward distributions of long-horizon symbolic and multi-image tasks. Reference regularization uses the unbiased KL estimator $\mathbb{E}[\exp(d_t) - d_t - 1]$ for Δ_{P} and a simpler squared log-probability surrogate for Δ_{M} and Δ_{V}; both vanish at $\pi_\theta = \pi_{\mathrm{ref}}$ and avoid the numerical instability of $\exp$ on long generated sequences.

Δ_{P}: Perceptual Grounding. We adopt a *task-routing* strategy across six per-frame tasks (SOR, VGL, DPE, CC, IC, RSR): SFT supervises stable structured outputs (SOR/VGL/DPE), while GRPO covers the full set including counting and relational reasoning. Routing tasks by output stability stabilizes optimization on tasks with sparse or noisy reward signals (Lin & Zheng, 2026) and prevents early collapse to format-degenerate solutions. For GRPO, we sample K predictions per prompt, compute $\mathcal{R}^{(\mathrm{P})}$, and normalize them into group-relative advantages A_k:

$$\mathcal{L}_{\mathrm{RL}}^{(\mathrm{P})} = -\frac{1}{K}\sum_{k=1}^{K} A_k \log \pi_\theta(\hat{y}^{(k)} \mid x, q) + \lambda_{\mathrm{ref}} \mathcal{R}_{\mathrm{ref}}^{(\mathrm{P})}, \tag{1}$$

addressing the known drift of RL fine-tuning away from the pretrained distribution (Kotha et al., 2024).

Δ_{M}: Symbolic Temporal Memory. Each episode is converted into a flattened action-step timeline; every retained step is a symbolic observation string preserving receptacle identities and object types. Operating in text isolates compression from perception, enabling M_ϕ to learn a clean symbolic interface that visual fusion can later anchor to (Chen et al., 2024). Training-free in-context aggregation suffers from positional dilution and quadratic context costs as histories grow (Liu et al., 2025); following persistent memory interfaces for embodied reasoning (Liu et al., 2025), M_ϕ compresses the symbolic history $o_{\le t}$ into K virtual memory tokens $M_t \in \mathbb{R}^{K\times d}$. Concretely, M_ϕ first applies attention pooling over the encoded symbolic timeline conditioned on a learnable set of K query slots, then carries the resulting summary across time through a GRU update, and finally projects the recurrent state back into the language-model embedding space so the tokens can be consumed as a memory prefix by the backbone. We use $K = 64$ in all experiments; the three input-context prompting modes (LoRA-only, memory-only, hybrid) and full architectural details are in Appendix C.4. The GRPO objective is:

$$\mathcal{L}_{\mathrm{RL}}^{(\mathrm{M})} = -A_t \cdot \frac{1}{N_{\mathrm{prefix}}} \sum_{i=1}^{N_{\mathrm{prefix}}} \log \pi_\theta(\hat{y}_{t,i} \mid z_t, \hat{y}_{t,<i}) + \lambda_{\mathrm{ref}} \mathcal{R}_{\mathrm{ref}}^{(\mathrm{M})}, \tag{2}$$

with two Δ_{M}-specific stabilization heuristics (restricting policy gradients to the answer prefix and skipping updates on already-failed examples; Appendix C.5). The output is Δ_{M} together with M_ϕ, providing a persistent memory interface inherited by Δ_{V}.

Δ_{V}: Visual Spatio-Temporal Memory. Task definitions and target JSON structures are unchanged from Δ_{M}; only the input modality changes from symbolic trajectories to RGB frame sequences, with M_ϕ reused unchanged so the symbolic temporal interface is preserved across modalities. Direct fine-tuning on a new modality is known to induce severe catastrophic forgetting (Zhai et al., 2024b; Wang et al., 2025c), especially when the new modality requires different attention patterns (Laitinen Imanov, 2026), so Δ_{V} explicitly preserves prior capabilities through both adapter composition and reference regularization. We merge the prior adapters into a reference policy $\Theta_{\mathrm{ref}} = \mathrm{Merge}(\Theta_0, \Delta_{\mathrm{P}}, \Delta_{\mathrm{M}})$, which serves as both initialization and reference, and attach a new trainable adapter $\Theta^{(\mathrm{V})} = \Theta_{\mathrm{ref}} \oplus \Delta_{\mathrm{V}}$ (Prabhakar et al., 2024; Kesim et al., 2024; Zhang et al., 2025b). For each query, a bounded visual history $V_{\le t} = \{I_{s_1}, \ldots, I_{s_B}\}$ within image budget B is constructed by uniform subsampling of the episode timeline, with each frame paired with an explicit temporal tag (e.g., `[Time 0]`, `[Time 1]`, ...); the temporal tag explicitly identifies the sampled observation step. We evaluate memory over these ordered observations; removing or shuffling tags at test time is examined in Appendix F. The RL objective uses the same squared log-probability surrogate as Δ_{M}, but anchored to Θ_{ref} rather

Model	Recall		Locate		Track		Detect		Accumulate		Score
	SOR-M	LOR	VGL-M	TL	RSR-M	STT	SED+	SED-	CSR+	CSR-	
(A) Visual-temporal regime (V-score): multi-frame RGB input.											
Proprietary Models											
ChatGPT-5.4 (2025)	19.1	22.0	1.1	48.9	38.4	4.3	70.5	69.3	20.6	13.7	30.8
ChatGPT-4o (2024)	18.4	23.8	1.0	44.7	35.5	0.0	64.8	71.1	16.2	5.5	28.1
Gemini-2.5-flash (2025)	14.4	0.6	0.6	9.5	37.6	0.0	57.1	66.7	10.3	3.8	20.1
Open-source Models											
Qwen3-VL-235B-a22b-Instruct (2025)	17.0	7.7	1.2	26.3	37.7	0.0	65.4	69.1	14.6	5.0	24.4
Qwen3-VL-32B-Instruct (2025)	29.4	18.0	2.6	44.6	35.6	0.0	65.0	65.7	16.9	4.8	28.3
Qwen3-VL-8B-Instruct (2025)	1.7	0.1	0.1	3.5	2.2	0.0	6.5	6.4	1.8	4.8	2.7
Qwen2.5-VL-72B-Instruct (2025)	18.4	15.8	0.5	36.9	40.9	8.7	64.4	58.4	15.6	6.4	26.6
Qwen2.5-VL-32B-Instruct (2025)	19.2	9.6	1.3	37.4	38.4	**10.5**	63.5	65.2	14.5	8.9	26.9
Qwen2.5-Omni-7B (2025)	4.9	23.4	0.2	**53.2**	28.1	1.1	58.3	54.8	17.7	5.2	24.7
Qwen2.5-VL-7B-Instruct (2025)	2.0	1.9	0.0	4.0	0.4	0.0	6.4	6.5	2.6	4.9	2.9
Qwen2.5-VL-3B-Instruct (2025)	—	—	—	—	—	—	—	—	—	—	21.85
LLaVA-Interleave-Qwen-7B (2024b)	0.9	0.6	0.0	1.2	1.8	0.0	3.4	4.8	0.4	3.9	1.7
LLaVA-v1.6-Mistral-7B (2024a)	1.6	2.1	0.0	4.1	1.2	0.0	2.6	6.1	2.9	4.8	2.5
LLaVA-OneVision-Qwen2-7B (2024a)	0.3	1.8	0.2	2.5	1.8	0.0	2.6	5.4	1.7	4.0	2.0
InternVL3.5-8B (2025b)	14.0	21.3	0.5	37.9	6.4	0.0	44.0	65.3	16.0	49.3	25.5
InternVL3-8B (2025)	20.4	29.4	1.8	37.1	15.3	0.0	52.7	63.3	25.0	49.3	29.4
Ours											
STaR-VL-3B (Full)	98.5	55.2	1.7	45.2	54.3	2.5	74.0	66.2	29.7	48.3	47.6
STaR-VL-3B-pmv	89.4	52.0	3.9	41.6	61.6	2.0	34.1	32.7	20.0	32.9	37.0
STaR-VL-7B (Full)	**99.0**	**60.0**	**5.8**	50.2	58.1	5.8	**97.0**	**84.9**	**36.9**	**57.7**	**55.5**
STaR-VL-7B-pmv	95.4	53.8	3.5	**53.2**	**63.5**	4.7	84.7	70.8	25.0	35.9	49.0
(B) Symbolic-history regime (M-score): ground-truth observation tokens replace visual input.											
Proprietary Models											
ChatGPT-5.4 (2025)	**99.9**	22.1	9.0	68.3	93.0	8.0	80.4	82.5	21.7	18.3	50.3
ChatGPT-4o (2024)	78.8	54.7	4.3	68.0	93.0	30.4	88.3	79.8	35.5	29.6	56.2
Gemini-2.5-flash (2025)	98.9	8.2	1.5	18.3	93.0	2.2	59.8	68.5	10.9	5.6	36.7
Open-source Models											
Qwen3-VL-235B-a22b-Instruct (2025)	99.6	61.6	**23.5**	67.0	92.6	21.0	87.0	81.9	47.9	25.0	60.7
Qwen3-VL-32B-Instruct (2025)	99.7	73.9	3.5	68.4	92.6	30.4	88.7	84.7	**59.5**	33.2	**63.5**
Qwen3-VL-8B-Instruct (2025)	89.8	81.5	2.8	82.1	42.9	0.0	79.3	75.6	38.3	50.3	54.3
Qwen2.5-VL-72B-Instruct (2025)	99.6	49.7	18.0	67.7	93.0	**36.2**	86.4	81.2	37.0	25.6	59.4
Qwen2.5-VL-32B-Instruct (2025)	98.3	19.3	1.5	67.0	**94.9**	35.1	80.2	81.4	15.6	14.1	50.7
Qwen2.5-Omni-7B (2025)	87.3	69.2	3.5	61.7	90.9	2.2	79.7	58.0	53.8	28.1	53.4
Qwen2.5-VL-7B-Instruct (2025)	86.6	81.6	1.3	74.4	25.8	0.0	65.5	62.6	36.4	50.2	48.4
Qwen2.5-VL-3B-Instruct (2025)	86.7	67.0	0.2	54.2	35.0	5.0	58.3	51.4	29.7	38.5	42.6
LLaVA-Interleave-Qwen-7B (2024b)	47.0	10.9	0.0	27.6	38.4	7.5	24.7	24.9	8.8	8.0	19.8
LLaVA-v1.6-Mistral-7B (2024a)	80.9	55.5	0.2	59.4	56.3	4.2	37.3	54.8	30.3	26.0	40.5
LLaVA-OneVision-Qwen2-7B (2024a)	86.7	57.7	0.8	36.8	38.7	0.0	33.8	59.2	23.8	45.5	38.3
InternVL3.5-8B (2025b)	89.5	**89.6**	0.7	**82.5**	64.4	0.0	78.5	70.9	47.1	49.5	57.3
InternVL3-8B (2025)	81.2	48.8	3.5	71.8	19.6	0.0	73.7	71.1	36.4	**51.3**	45.7
Ours											
STaR-VL-3B-M	98.7	82.0	6.4	65.8	64.3	16.4	**90.6**	**85.6**	53.9	49.8	61.4
STaR-VL-3B-m	94.2	78.9	0.0	65.2	54.2	0.0	58.3	51.4	29.7	38.5	47.0

Table 1: **Visual-history (V) and symbolic-history (M) evaluations.** Metrics are scaled to $[0, 100]$. Regimes differ in input representation, available scene state, and history coverage. The corrected Qwen2.5-VL-3B V-score is 21.85; its corrected task-level breakdown is unavailable and shown as dashes. Bold and underlining retain the task-wise rankings among reported values. Dashed lines separate model families.

than the pretrained backbone:

$$\mathcal{R}_{\text{ref}}^{(\text{V})} = \frac{1}{N_t} \sum_{i=1}^{N_t} \left(\log \pi_{\Theta^{(\text{V})}}(\hat{y}_{t,i} \mid \cdot) - \log \pi_{\Theta_{\text{ref}}}(\hat{y}_{t,i} \mid \cdot) \right)^2, \tag{3}$$

acting analogously to elastic-weight anchoring in continual learning (Kirkpatrick et al., 2017). The full RL objective is:

$$\mathcal{L}_{\text{RL}}^{(\text{V})} = -A_t \cdot \frac{1}{N_t} \sum_{i=1}^{N_t} \log \pi_{\Theta^{(\text{V})}}(\hat{y}_{t,i} \mid V_{\leq t}, q_t, \hat{y}_{t,<i}) + \lambda_{\text{ref}} \mathcal{R}_{\text{ref}}^{(\text{V})}. \tag{4}$$

The teacher-forced SFT counterpart is in Appendix C.6.

4 Experiment

Evaluation suite. We organize our experiments around three macro scores that align with the three adapters of STaR-VL. The *V-score* (visual-temporal) is the macro average over ten task-level

Model	VSI-Bench (2025)				MMBench-Spatial (2024b)					Ego. (2023)	TempCompass (2024c)			OOD
	Ord.	Cnt.	Dist.	Avg.	Loc.	Prop.	Rel.	Spat.	Avg.	QA	Act.	Ord.	Avg.	
Open-source Models														
Qwen2.5-VL-3B-Instruct (2025)	24.0	17.3	38.0	26.4	66.0	84.0	36.0	54.0	60.0	21.5	62.5	42.9	52.7	40.2
Qwen2.5-VL-7B-Instruct (2025)	36.0	16.0	48.0	33.3	**70.0**	82.0	46.0	60.0	**64.5**	23.0	67.5	42.9	55.2	44.0
Ours														
STaR-VL-3B (Full)	42.0	**20.0**	56.0	39.3	66.0	72.0	**50.0**	52.0	60.0	25.0	63.0	43.0	53.0	44.3
STaR-VL-7B (Full)	**51.0**	17.1	**71.0**	**46.4**	62.0	**86.0**	42.0	**66.0**	64.0	**27.0**	**69.0**	**45.7**	**57.4**	**48.7**

Table 2: Out-of-distribution benchmark results. All metrics are scaled to $[0, 100]$. OOD denotes the macro average over benchmark-level averages. Best result in each column is in bold and second-best is underlined.

metrics on the multi-image visual evaluation suite covering RECALL, LOCATE, TRACK, DETECT, and ACCUMULATE (Sec. 4.2), and is our primary metric. The *M-score* (symbolic memory) uses the same five capabilities on a symbolic-history split where visual input is replaced by ground-truth observations, isolating temporal compression from perception (Sec. 4.3). The *P-score* (per-frame perception) measures per-frame grounding ability and is reported in Appendix D.2. Variants follow a naming convention over $\{P, M, V\}$ where the string denotes which adapters are stacked, uppercase indicates SFT+RL, and lowercase indicates SFT only; the full mapping and the capability-to-task mapping across SpaMEM (Liao et al., 2026) and OOD benchmarks are in Appendix D.1 (Tables 5 and 4).

Baselines. We compare against three families: (i) closed-source models (*ChatGPT-5.4* (OpenAI, 2025), *ChatGPT-4o* (OpenAI, 2024), *Gemini-2.5-Flash* (Google DeepMind, 2025)); (ii) open-source VLMs (Qwen2.5-VL (Bai et al., 2025), Qwen2.5-Omni (Xu et al., 2025) and Qwen3-VL (Qwen Team, 2025), 3B–235B); and (iii) multimodal baselines built on alternative backbones (LLaVA-Interleave (Li et al., 2024b), LLaVA-OneVision (Li et al., 2024a), LLaVA-NeXT (Liu et al., 2024a), InternVL3 (Zhu et al., 2025) and InternVL3.5 (Wang et al., 2025b)). Both 3B and 7B STaR-VL backbones are evaluated to examine scaling behavior, and we always include the SFT-only counterpart (**pmv**) and intermediate variants (**PMv**, **PM**) to isolate the contribution of each component.

Generalization beyond SpaMEM. To test whether STaR-VL's gains generalize beyond the training distribution, we evaluate it zero-shot on four widely-used external benchmarks chosen to span the four primary axes of visual-temporal reasoning: VSI-Bench (multi-image spatial intelligence), MMBench-Spatial (single-image spatial reasoning), EgoSchema (long-horizon video QA), and TempCompass (short-horizon temporal ordering). All four use the original benchmark's multiple-choice or short-answer protocol under shared inference settings across all evaluated models; details are in Sec. 4.4 and Appendix D.4.

Ablations. Two ablation studies isolate the contribution of model design and inference-time prompting on the same trained PMV (Full) checkpoint. The *adapter ablation* evaluates seven configurations (**PMV**, **PM**, **MV**, **PV**, **P**, **M**, **V**) on V-score to quantify the marginal contribution of each adapter (Sec. 4.5); detailed per-capability breakdowns are in Appendix D.5. The *prompt-format ablation* fixes the trained checkpoint and varies only the inference-time prompt wrapper across six variants (Train-aligned, +CoT, +JSON, +R1, +NoReason, +Spatial), testing whether STaR-VL's behavior is brittle to prompt phrasing (right panel of Fig. 3 and Sec. 4.5); full details are in Appendix D.6.

Data and training. SpaMEM uses a house-disjoint split: training comprises 10 houses, $6{,}011$ episode queries, and $22{,}592$ images; held-out evaluation comprises 5 houses, $2{,}742$ queries, and $10{,}304$ images. Within each stage, SFT and RL use the same underlying examples; there is no RL-only dataset. Stage-specific representations and task routing are described in Sec. 3. The main 3B and 7B visual runs use budgets of three and two frames, respectively. Same-backbone controls and an explicit history-budget sweep provide the principal comparisons.

4.1 SAME-BACKBONE TRAINING AND MEMORY CONTROLS

Beyond in-domain joint training. Table 3 separates the benefit of task exposure from the additional gain of the complete STaR-VL framework. A joint rank-16 adapter reaches 38.50, and increasing its rank to 48 raises performance to 40.80. STaR-VL reaches 47.60, improving over the higher-capacity joint adapter by 6.80 points. The joint controls use the same in-domain task data without the staged

adapter cascade. The comparison supports the combined framework; it does not attribute the full residual to stage order independently of recurrent memory, adapter composition, or optimization differences.

Beyond persistent-memory augmentation. We adapt ReMEmbR (Anwar et al., 2024) to the SpaMEM evaluation interface. On the same Qwen2.5-VL-3B backbone, self-generated memory alone reaches 34.50, and retaining the visual frames alongside memory reaches 38.90. This 17.05-point gain over the raw backbone establishes persistent memory as a consequential baseline. STaR-VL adds a further 8.70 points. An additional InternVL3-8B comparison, reported in Appendix F, contrasts self-generated memory with ground-truth state under a common evaluation protocol.

Qwen2.5-VL-3B system	V-score	Gain over base
Untrained backbone	21.85	—
Joint LoRA, rank 16	38.50	+16.65
Joint LoRA, rank 48	40.80	+18.95
ReMEmbR-adapted, memory only	34.50	+12.65
ReMEmbR-adapted, memory + frames	38.90	+17.05
STaR-VL, staged P→M→V	**47.60**	**+25.75**

Table 3: **Same-backbone training and memory controls.** Joint LoRA uses the in-domain task data without the staged adapter cascade. ReMEmbR is adapted to the SpaMEM interface. These comparisons evaluate complete systems; they do not isolate training order from architecture and optimization.

Structured prompting. The memory-table prompt asks the model to inspect each frame, list objects and locations, compare adjacent observations, maintain a state table, and then answer. It changes the untrained backbone's V-score from 21.85 to 22.21, and STaR-VL's from 47.60 to 48.30. Under this prompt STaR-VL remains 26.09 points ahead. Across six additional prompt wrappers, the untrained model scores 20.92–22.60. Thus the tested prompting procedures do not reproduce the learned system's gains; full results and serialization controls appear in Appendix F.

Memory quality on a second backbone. The InternVL3-8B control complements the same-backbone comparison: self-generated memory with frames reaches 43.60, compared with 29.40 from raw frames and 50.51 with oracle scene state plus frames (Appendix F). Retaining images also improves over memory-only inputs, from 39.80 to 43.60 for self-generated memory and from 45.70 to 50.51 for oracle memory. Thus both accessible scene state and retained visual evidence contribute in this setting. The remaining gap between self-generated and oracle memory identifies a concrete target for better state construction and retrieval.

4.2 Visual-Temporal Memory Results

Aggregate and task-level performance. Table 1(a) reports visual-history evaluation. STaR-VL reaches V-score 47.6 at 3B and 55.5 at 7B. The corrected 3B backbone aggregate is 21.85, giving a 25.75-point gain; the same-backbone controls in Sec. 4.1 establish the more informative comparison with alternative post-training and memory systems. Comparisons with general-purpose zero-shot VLMs provide context for task-specific adaptation rather than a controlled test of parameter scaling.

STaR-VL-3B scores 98.5 on object recall (SOR-M), 74.0 and 66.2 on event addition/removal, and 29.7 and 48.3 on cumulative addition/removal. Its gain over the SFT-only configuration (37.0) is 10.6 points. The reported per-task scores show a heterogeneous pattern: visual grounding (VGL-M) remains at 1.7 and spatial-temporal tracking (STT) at 2.5, while relational recall (RSR-M) is lower than the SFT-only configuration. The aggregate improvement therefore reflects a redistribution of task performance, with particularly strong object recall and event detection.

History coverage and accessible state. On STaR-VL-3B, increasing the visual budget from $B = 3$ to $B = 16$ raises V-score from 47.60 to 52.40. With the symbolic score held at 61.40, the visual–symbolic difference shrinks from 13.80 to 9.00. Coverage accounts for 4.80 points, or 34.78% of the initial difference. Because symbolic inputs still provide denoised scene state and full history, the remaining difference combines perception, state extraction, memory access, and downstream reasoning. Appendix F reports the full sweep.

Diagnostic interpretation. Figure 2 places visual-history performance alongside differences from symbolic history and per-frame perception. These plots summarize the supplied evaluations across

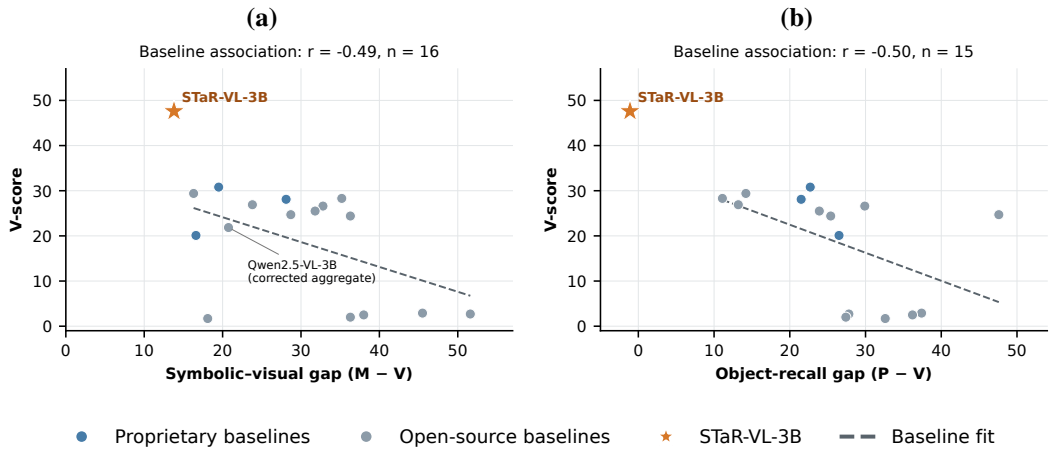

Figure 2: **Descriptive performance-gap comparisons.** (a) Reported M-score minus V-score, using the corrected Qwen2.5-VL-3B aggregate. (b) Per-frame SOR minus visual-history SOR-M; Qwen2.5-VL-3B is excluded because its corrected task-level breakdown is unavailable. Fits use baseline models only. The M−V horizontal coordinate contains V-score itself, so its correlation with V-score is partly mathematically coupled and does not identify a causal mechanism.

model families. STaR-VL combines high recall with a small recall gap (97.4 in the perception configuration versus 98.5 in the full configuration), while other capabilities remain uneven. For STaR-VL, the compared regimes also use different adapter configurations. We therefore interpret these differences as operational diagnostics and use the direct training, memory, prompting, and coverage controls to assess the framework.

4.3 Symbolic-History Memory Results

Table 1(b) evaluates temporal queries from symbolic histories. STaR-VL-3B-M reaches an M-score of 61.4, compared with 42.6 for the reported backbone configuration and 47.0 for the SFT-only memory configuration. This shows that the symbolic-memory stage learns useful task behavior before visual adaptation. The resulting M−V difference of 13.8 compares the symbolic-memory configuration with the full visual model. It is a diagnostic reference, rather than a formal upper bound: history coverage, perception noise, adapter configuration, and state representation differ between regimes. The history-budget control in Sec. 4.2 quantifies one of these differences directly.

4.4 Transfer to Held-Out Benchmarks

Table 2 compares each trained checkpoint with its matched Qwen2.5-VL backbone on four benchmarks not used for training. STaR-VL-3B improves VSI-Bench from 26.4 to 39.3, EgoSchema from 21.5 to 25.0, and TempCompass from 52.7 to 53.0; MMBench-Spatial remains at 60.0. At 7B, VSI-Bench improves from 33.3 to 46.4, EgoSchema from 23.0 to 27.0, and TempCompass from 55.2 to 57.4, while MMBench-Spatial changes from 64.5 to 64.0.

These results support transfer across scene distributions and answer protocols, with the largest gain on VSI-Bench and smaller improvements on the video benchmarks. The single-image spatial benchmark does not improve in aggregate. The OOD comparison is restricted to matched backbones, and its task-dependent pattern complements the heterogeneous in-domain results rather than establishing uniform gains across spatial and temporal reasoning.

4.5 Ablation Study

Adapter configurations. Figure 3a compares the reported configurations. MV and PV score 42.1 and 41.1, V alone scores 37.9, P scores 32.9, PM scores 24.5, and M scores 17.9. The full configuration has the highest reported performance (47.7 in the adapter-ablation summary, compared with 47.6 in the main evaluation). Because the adapters were acquired sequentially,

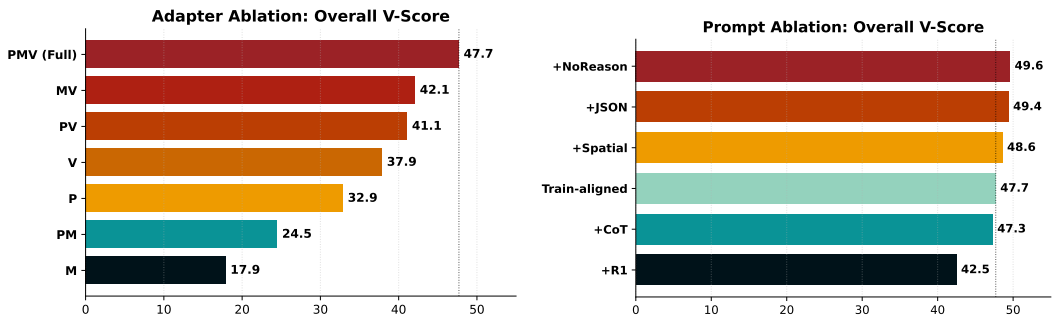

Figure 3: **V-score ablations on STaR-VL-3B (PMV).** (a) Adapter ablation: reported component configurations. (b) Prompt ablation: five of six variants within a 2.3-point band. Breakdowns in Appendix D.5, D.6.

removing components from the trained system changes its input to downstream components and is not equivalent to independently retraining each architecture. We therefore interpret this ablation as dependence on the learned composition, complementary to the trained controls in Table 3.

Prompt-format robustness. Five of the six prompt-wrapper scores on the trained checkpoint lie between 47.3 and 49.6; the R1-style wrapper scores 42.5. The additional untrained-backbone and structured-prompt controls in Appendix F address a different question: whether prompting alone reproduces the gains. Figure 3b and Appendix D.6.2 describe the checkpoint's wrapper sensitivity.

4.6 Temporal Ordering and Output Robustness

Temporal-tag and serialization controls distinguish two ways the task interface can affect performance. Removing time tags changes object recall from 98.50 to 96.20, while relational recall falls from 54.30 to 25.80 and event-addition detection from 74.00 to 24.60. Shuffling tags produces a similar contrast between stable object recall and weaker order-sensitive behavior. These outcomes separate retaining the observed object set from using temporal position to compare observations. Since the checkpoint was trained with correct tags, the perturbations measure sensitivity under a changed input distribution. They support an explicitly ordered-history interpretation of the evaluated capability.

Changing the answer schema provides a complementary test. Object recall remains at 91.20 with a dictionary, 88.60 with plain lines, and 93.80 with a wrapped object, compared with 98.50 using the trained JSON list. Each alternative parser passes a ground-truth round-trip test. The retained performance shows that object recall extends beyond the exact training serialization, while the decreases quantify practical format sensitivity. Together, these controls suggest evaluating visual-temporal systems along separate axes: retained content, access to temporal position, and answer extraction. Appendix F reports the complete control tables.

5 Conclusion

STaR-VL learns visual-temporal memory through sequential perception, symbolic-memory, and visual-adaptation stages with capability-aligned rewards. On Qwen2.5-VL-3B, its V-score of 47.60 exceeds the corrected backbone score of 21.85, joint rank-48 LoRA by 6.80 points, and adapted persistent memory with frames by 8.70 points. Structured prompting does not reproduce these gains, and held-out evaluations show transfer beyond SpaMEM. Increasing visual history from three to sixteen frames raises V-score from 47.60 to 52.40, recovering 4.80 points, or 34.78% of the initial 13.80-point visual–symbolic difference. Coverage and symbolic-state controls therefore help separate history access from the quality of the available scene representation. The resulting contribution is a practical staged post-training framework with controlled evidence for stronger recall, event detection, and cumulative memory; fine-grained grounding and temporal localization remain priorities for further development, alongside improved construction and retrieval of persistent scene state.

AI USE STATEMENT

Generative AI tools were used to draft and revise manuscript sections, for language editing, feedback on manuscript claims and experimental controls, interpretation of reported results, and LaTeX formatting. Models used in the method and evaluations are described in the paper. Ground-truth training labels are derived from procedural scene state. The authors are responsible for the final content of this work.

ETHICS STATEMENT

The training environment and supervision are object-centric. Stronger temporal memory may also facilitate persistent surveillance and profiling; deployment in person-centric settings requires attention to consent, privacy, and inherited model biases. Appendix B.2 discusses these risks and possible mitigations.

REPRODUCIBILITY STATEMENT

Section 3 and Appendix C.1 onward specify the training objectives, memory module, and adapter construction. Appendix D.1 describes the data split and evaluation setup, Appendix D.6 provides prompt templates, and Appendix F details the additional controls. Table 1 explicitly distinguishes the corrected aggregate from the unavailable corrected task-level breakdown.

REFERENCES

Abrar Anwar, John Welsh, Joydeep Biswas, Soha Pouya, and Yan Chang. Remembr: Building and reasoning over long-horizon spatio-temporal memory for robot navigation. *arXiv preprint arXiv:2409.13682*, 2024.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 41–48, 2009. doi: 10.1145/1553374.1553380.

Boyuan Chen et al. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. *arXiv preprint arXiv:2401.12168*, 2024.

Hanyang Chen, Mark Zhao, Rui Yang, Qinwei Ma, Ke Yang, Jiarui Yao, Kangrui Wang, Hao Bai, Zhenhailong Wang, Rui Pan, Mengchao Zhang, Jose Barreiros, Aykut Onol, ChengXiang Zhai, Heng Ji, Manling Li, Huan Zhang, and Tong Zhang. Era: Transforming vlms into embodied agents via embodied prior learning and online reinforcement learning. *arXiv preprint arXiv:2510.12693*, 2025.

Kai Cheng, Zhengyuan Li, Xingpeng Sun, Byung-Cheol Min, Amrit Singh Bedi, and Aniket Bera. Efficienteqa: An efficient approach for open vocabulary embodied question answering. *arXiv preprint arXiv:2410.20263*, 2024.

Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Yifei Dong, Fengyi Wu, Guangyu Chen, Lingdong Kong, Xu Zhu, Qiyu Hu, Yuxuan Zhou, Jingdong Sun, Jun-Yan He, Qi Dai, Alexander G. Hauptmann, and Zhi-Qi Cheng. Towards unified world models for visual navigation via memory-augmented planning and foresight. *arXiv preprint arXiv:2510.08713*, 2025.

Vishnu Sashank Dorbala and Dinesh Manocha. Memctrl: Using mllms as active memory controllers on embodied agents. *arXiv preprint arXiv:2601.20831*, 2026.

Vishnu Sashank Dorbala, Prasoon Goyal, Robinson Piramuthu, Michael Johnston, Reza Ghanadan, and Dinesh Manocha. S-eqa: Tackling situational queries in embodied question answering. *arXiv preprint arXiv:2405.04732*, 2024.

Yue Fan, Xiaojian Ma, Rongpeng Su, Jun Guo, Rujie Wu, Xi Chen, and Qing Li. Embodied VideoAgent: Persistent memory from egocentric videos and embodied sensors enables dynamic scene understanding. In *International Conference on Computer Vision (ICCV)*, 2025.

Google DeepMind. Gemini 2.5: Our most intelligent ai model, 2025. `https://blog.google/technology/google-deepmind/gemini-model-thinking-updates-march-2025/`.

Wenbo Hu, Yining Hong, Yanjun Wang, Leison Gao, Zibu Wei, Xingcheng Yao, Nanyun Peng, Yonatan Bitton, Idan Szpektor, and Kai-Wei Chang. 3dllm-mem: Long-term spatial-temporal memory for embodied 3d large language model. *arXiv preprint arXiv:2505.22657*, 2025.

Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models. *arXiv preprint arXiv:2503.06749*, 2025.

Kaixuan Jiang, Yang Liu, Weixing Chen, Jingzhou Luo, Ziliang Chen, Ling Pan, Guanbin Li, and Liang Lin. Beyond the destination: A novel benchmark for exploration-aware embodied question answering. *arXiv preprint arXiv:2503.11117*, 2025.

Ege Kesim et al. Multi lora meets vision: Merging multiple adapters to create a multi-task model. *arXiv preprint arXiv:2411.14064*, 2024.

Dongyoung Kim, Sumin Park, Huiwon Jang, Jinwoo Shin, Jaehyung Kim, and Younggyo Seo. Robot-r1: Reinforcement learning for enhanced embodied reasoning in robotics. *arXiv preprint arXiv:2506.00070*, 2025.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114 (13):3521–3526, 2017.

Suhas Kotha, Jacob Mitchell Springer, and Aditi Raghunathan. Understanding catastrophic forgetting in language models via implicit inference. In *International Conference on Learning Representations*, 2024.

Olaf Yunus Laitinen Imanov. Mechanistic analysis of catastrophic forgetting in large language models during continual fine-tuning. *arXiv preprint arXiv:2601.18699*, 2026.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024a.

Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv preprint arXiv:2407.07895*, 2024b.

Chih-Ting Liao, Xi Xiao, Chunlei Meng, Zhangquan Chen, Yitong Qiao, Weilin Zhou, Tianyang Wang, Xu Zheng, and Xin Cao. Spamem: Benchmarking dynamic spatial reasoning via perception-memory integration in embodied environments. *arXiv preprint arXiv:2604.22409*, 2026.

Zekai Lin and Xu Zheng. Panoenv: Exploring 3d spatial intelligence in panoramic environments with reinforcement learning. *arXiv preprint arXiv:2602.21992*, 2026.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge. https://llava-vl.github.io/blog/2024-01-30-llava-next/, 2024a.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European Conference on Computer Vision*, pp. 216–233. Springer, 2024b.

Yuanxin Liu, Shicheng Li, Yi Liu, Yuxiang Wang, Shuhuai Ren, Lei Li, Sishuo Chen, Xu Sun, and Lu Hou. Tempcompass: Do video llms really understand videos? In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 8731–8772. Association for Computational Linguistics, 2024c.

Zuntao Liu, Yi Du, Taimeng Fu, Shaoshu Su, Cherie Ho, and Chen Wang. Vision-language memory for spatial reasoning. *arXiv preprint arXiv:2511.20644*, 2025.

Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul McVay, Oleksandr Maksymets, Sergio Arnaud, Karmesh Yadav, Qiyang Li, Ben Newman, Mohit Sharma, Vincent-Pierre Berges, Shiqi Zhang, Pulkit Agrawal, Yonatan Bisk, Dhruv Batra, Mrinal Kalakrishnan, Franziska Meier, Chris Paxton, Saurabh Gupta, and Aravind Rajeswaran. Openeqa: Embodied question answering in the era of foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

OpenAI. Hello gpt-4o, 2024. https://openai.com/index/hello-gpt-4o/.

OpenAI. Introducing gpt-5, 2025. https://openai.com/index/introducing-gpt-5/.

Yohan Park, Hyunwoo Ha, Wonjun Jo, and Tae-Hyun Oh. Darkeqa: Benchmarking vision-language models for embodied question answering in low-light indoor environments. *arXiv preprint arXiv:2512.24985*, 2025.

Akshara Prabhakar, Yuanzhi Li, Karthik Narasimhan, Sham Kakade, Eran Malach, and Samy Jelassi. Lora soups: Merging loras for practical skill composition tasks. *arXiv preprint arXiv:2410.13025*, 2024.

Qwen Team. Qwen3-vl technical report. *arXiv preprint*, 2025. https://github.com/QwenLM/Qwen3-VL.

Gabriel Sarch, Zhaoyuan Fang, Adam W. Harley, Paul Schydlo, Michael J. Tarr, Saurabh Gupta, and Katerina Fragkiadaki. TIDEE: Tidying up novel rooms using visuo-semantic commonsense priors. In *European Conference on Computer Vision (ECCV)*, 2022.

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, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Zirui Song, Guangxian Ouyang, Mingzhe Li, Yuheng Ji, Chenxi Wang, Zixiang Xu, Zeyu Zhang, Xiaoqing Zhang, Qian Jiang, Zhenhao Chen, Zhongzhi Li, Rui Yan, and Xiuying Chen. Maniplvm-r1: Reinforcement learning for reasoning in embodied manipulation with large vision-language models. *arXiv preprint arXiv:2505.16517*, 2025.

Chaoyang Wang, Wenrui Bao, Sicheng Gao, Bingxin Xu, Yu Tian, Yogesh S. Rawat, Yunhao Ge, and Yuzhang Shang. Vla-thinker: Boosting vision-language-action models through thinking-with-image reasoning. *arXiv preprint arXiv:2603.14523*, 2026a.

Qineng Wang, Wenlong Huang, Yu Zhou, Hang Yin, Tianwei Bao, Jianwen Lyu, Weiyu Liu, Ruohan Zhang, Jiajun Wu, Li Fei-Fei, and Manling Li. Enact: Evaluating embodied cognition with world modeling of egocentric interaction. *arXiv preprint arXiv:2511.20937*, 2025a.

Sen Wang, Bangwei Liu, Zhenkun Gao, Lizhuang Ma, Xuhong Wang, Yuan Xie, and Xin Tan. Explore with long-term memory: A benchmark and multimodal llm-based reinforcement learning framework for embodied exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026b. arXiv:2601.10744.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Yuhang Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025b.

Ziqi Wang et al. Smolora: Exploring and defying dual catastrophic forgetting in continual visual instruction tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025c.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Tao Wu, Chuhao Zhou, Yen Heng Wong, Lin Gu, and Jianfei Yang. Noisyeqa: Benchmarking embodied question answering against noisy queries. *arXiv preprint arXiv:2412.10726*, 2024.

Jiannan Xiang, Tianhua Tao, Yi Gu, Tianmin Shu, Zirui Wang, Zichao Yang, and Zhiting Hu. Language models meet world models: Embodied experiences enhance language models. In *Advances in Neural Information Processing Systems*, 2023.

Jiannan Xiang, Guangyi Liu, Yi Gu, Qiyue Gao, Yuting Ning, Yuheng Zha, Zeyu Feng, Tianhua Tao, Shibo Hao, Yemin Shi, Zhengzhong Liu, Eric P. Xing, and Zhiting Hu. Pandora: Towards general world model with natural language actions and video states. *arXiv preprint arXiv:2406.09455*, 2024.

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. *arXiv preprint arXiv:2503.20215*, 2025.

Karmesh Yadav, Yusuf Ali, Gunshi Gupta, Yarin Gal, and Zsolt Kira. FindingDory: A benchmark to evaluate memory in embodied agents. *arXiv preprint arXiv:2506.15635*, 2025.

Jihan Yang, Shusheng Yang, Anjali W. Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. *arXiv preprint arXiv:2412.14171*, 2024.

Jihan Yang, Shusheng Yang, Anjali W. Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. *arXiv preprint arXiv:2412.14171*, 2025.

Angen Ye, Zeyu Zhang, Boyuan Wang, Xiaofeng Wang, Dapeng Zhang, and Zheng Zhu. Vla-r1: Enhancing reasoning in vision-language-action models. *arXiv preprint arXiv:2510.01623*, 2025.

Licheng Yu, Xinlei Chen, Georgia Gkioxari, Mohit Bansal, Tamara L. Berg, and Dhruv Batra. Multi-target embodied question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

Mingliang Zhai, Zhi Gao, Yuwei Wu, and Yunde Jia. Memory-centric embodied question answering. *arXiv preprint arXiv:2505.13948*, 2025.

Yuexiang Zhai, Hao Bai, Zipeng Lin, Jiayi Pan, Shengbang Tong, Yifei Zhou, Alane Suhr, Saining Xie, Yann LeCun, Yi Ma, and Sergey Levine. Fine-tuning large vision-language models as decision-making agents via reinforcement learning. In *Advances in Neural Information Processing Systems*, 2024a.

Yuexiang Zhai, Shengbang Tong, Xiao Li, Mu Cai, Qing Qu, Yong Jae Lee, and Yi Ma. Investigating the catastrophic forgetting in multimodal large language models. In *Proceedings of Machine Learning Research*, 2024b.

Haochen Zhang, Nirav Savaliya, Faizan Siddiqui, and Enna Sachdeva. Fast-eqa: Efficient embodied question answering with global and local region relevancy. *arXiv preprint arXiv:2602.15813*, 2026.

Jingyi Zhang, Jiaxing Huang, Huanjin Yao, Shunyu Liu, Xikun Zhang, Shijian Lu, and Dacheng Tao. R1-vl: Learning to reason with multimodal large language models via step-wise group relative policy optimization. *arXiv preprint arXiv:2503.12937*, 2025a.

Juzheng Zhang, Jiacheng You, Ashwinee Panda, and Tom Goldstein. Lori: Reducing cross-task interference in multi-task low-rank adaptation. *arXiv preprint arXiv:2504.07448*, 2025b.

Baining Zhao, Ziyou Wang, Jianjie Fang, Chen Gao, Fanhang Man, Jinqiang Cui, Xin Wang, Xinlei Chen, Yong Li, and Wenwu Zhu. Embodied-r: Collaborative framework for activating embodied spatial reasoning in foundation models via reinforcement learning. *arXiv preprint arXiv:2504.12680*, 2025a.

Yong Zhao, Kai Xu, Zhengqiu Zhu, Yue Hu, Zhiheng Zheng, Yingfeng Chen, Yatai Ji, Chen Gao, Yong Li, and Jincai Huang. Cityeqa: A hierarchical llm agent on embodied question answering benchmark in city space. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 2025b.

Heng Zhou, Li Kang, Yiran Qin, Xiufeng Song, Ao Yu, Zilu Zhang, Haoming Song, Kaixin Xu, Yuchen Fan, Dongzhan Zhou, Xiaohong Liu, Ruimao Zhang, Philip Torr, Lei Bai, and Zhenfei Yin. Ego to world: Collaborative spatial reasoning in embodied systems via reinforcement learning. *arXiv preprint arXiv:2603.14811*, 2026a.

Hongyi Zhou, Kai Ye, Erhan Xu, Jin Zhu, Ying Yang, Shijin Gong, and Chengchun Shi. Demystifying group relative policy optimization: Its policy gradient is a u-statistic. *arXiv preprint arXiv:2603.01162*, 2026b.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

A Extended Related Work

This appendix expands the condensed Related Works section in the main paper (Sec. 2) with detailed discussion of each of the three lines of work that STaR-VL builds on.

Embodied question answering and long-horizon embodied reasoning. Embodied question answering (EQA) studies agents that must actively gather visual evidence from an environment before producing a grounded answer. Early work established the original EQA formulation and later extended it to more compositional multi-target settings, showing that embodied QA already requires nontrivial integration of navigation, perception, and memory (Das et al., 2018; Yu et al., 2019). More recent benchmarks have substantially broadened this scope. OpenEQA reframes embodied QA in the era of foundation models with open-vocabulary questions and realistic answering settings (Majumdar et al., 2024). Subsequent work further stresses richer and more realistic conditions, including situational queries that depend on consensus over multiple object states (Dorbala et al., 2024), efficient open-vocabulary exploration (Cheng et al., 2024), robustness to noisy questions (Wu et al., 2024), exploration-aware evaluation (Jiang et al., 2025), urban-scale embodied QA (Zhao et al., 2025b), and robustness under adverse sensing conditions such as low light (Park et al., 2025). FAST-EQA further highlights the importance of compact memory and efficient exploration for embodied question answering (Zhang et al., 2026). These works collectively show that embodied QA is no longer merely a navigation-plus-answering problem. Instead, modern EQA increasingly tests whether an agent can maintain coherent intermediate evidence over long horizons, decide what to observe next, and answer under imperfect or ambiguous conditions. Our work is closely related to this line, but differs in emphasis: rather than centering on exploration strategy or answer generation alone, we focus on the gap between strong spatial reasoning from immediate observations and weak temporal memory under dynamic scene evolution.

Spatial memory, belief revision, and world models for embodied agents. A parallel line of work argues that embodied intelligence fundamentally depends on persistent state construction rather than single-step recognition. Language Models Meet World Models showed that embodied experiences can be injected into language models through world-model interaction and parameter-efficient adaptation (Xiang et al., 2023). Pandora further broadens this view by treating multimodal world modeling as action-conditioned prediction over video states and natural-language actions (Xiang et al., 2024). More directly related to our setting, Thinking in Space studies whether multimodal language models can see, remember, and recall spaces over time (Yang et al., 2024), while ReMEmbR demonstrates retrieval-based long-horizon spatio-temporal memory for embodied navigation (Anwar et al., 2024). Recent work has made memory an explicit architectural primitive rather than a passive context buffer. MemoryEQA re-centers embodied QA around memory retrieval and aggregation (Zhai et al., 2025), and 3DLLM-Mem combines working-memory and episodic-memory mechanisms for embodied 3D reasoning (Hu et al., 2025). ENACT frames embodied cognition as world modeling over egocentric interaction sequences (Wang et al., 2025a), while UniWM integrates planning, foresight, and memory-augmented state modeling within a unified navigation framework (Dong et al., 2025). Long-term memory has also begun to appear as a first-class optimization target: LMEE studies embodied exploration with explicit long-term memory utilization (Wang et al., 2026b), and MemCtrl treats memory retention and pruning as an active control problem for embodied agents (Dorbala & Manocha, 2026). Taken together, these works suggest that failures in embodied spatial intelligence are often failures of state maintenance, memory compression, and belief revision rather than failures of local perception alone. Our work is directly motivated by this perspective. Instead of assuming that stronger visual encoders automatically yield stronger temporal reasoning, we explicitly study how to bridge grounded perception, symbolic temporal memory, and visual temporal memory through a progressive post-training pipeline.

Reinforcement learning for multimodal and embodied reasoning. Recent work increasingly uses reinforcement learning not only to optimize control policies, but also to improve reasoning quality in multimodal and embodied models. Building on the foundation of policy-gradient methods such as PPO (Schulman et al., 2017) and the verifiable-reward training paradigm popularized by DeepSeek-R1 (DeepSeek-AI, 2025; Shao et al., 2024), several works have transferred R1-style training to vision-language models for general multimodal reasoning, including Vision-R1 (Huang et al., 2025) and R1-VL (Zhang et al., 2025a). An important early example in the embodied direction is the RL fine-tuning framework of Zhai et al., which shows that large vision-language models can be directly optimized as decision-making agents from interaction rewards (Zhai et al., 2024a). More recent

embodied reasoning systems make reward design itself more explicit and structured. Embodied-R applies reinforcement learning to activate embodied spatial reasoning with consistency-aware rewards (Zhao et al., 2025a). Robot-R1 and ManipLVM-R1 extend R1-style or RLVR-style training to robotics and embodied manipulation (Kim et al., 2025; Song et al., 2025). VLA-R1 further demonstrates that verifiable rewards can be defined over region alignment, trajectory consistency, and output formatting for vision-language-action models (Ye et al., 2025). Another common pattern is progressive or staged post-training, a strategy with deep roots in curriculum learning (Bengio et al., 2009). ERA combines embodied prior learning with online reinforcement learning (Chen et al., 2025), and VLA-Thinker uses a cold-start reasoning phase followed by GRPO-style alignment for long-horizon multimodal reasoning (Wang et al., 2026a). Particularly relevant to our work are methods that apply RL to embodied spatial reasoning with structured rewards. PanoEnv introduces geometry-aware reward design and curriculum-based GRPO for panoramic spatial intelligence (Lin & Zheng, 2026), while Ego-to-World studies cross-view embodied reasoning with dense spatial rewards (Zhou et al., 2026a). Our method follows this emerging RL-for-multimodal-reasoning direction, but targets a different bottleneck. Rather than focusing on static geometric correctness or downstream action success alone, we study reinforcement learning as a mechanism for aligning grounded perception, temporal state tracking, and long-horizon memory consistency under dynamic scene evolution.

B Extended Discussion: Limitations, Broader Impact, and Future Work

This appendix extends the brief conclusion in the main paper (Sec. 5) with detailed limitations, broader societal impact, and future-work discussion. Our discussion is organized into three subsections: (a) limitations that scope our claims, (b) broader societal impacts including positive efficiency-driven access and dual-use risks, and (c) concrete future research directions.

B.1 Limitations

The empirical conclusions have the following scope.

Bounded visual context in the visual-temporal stage. The visual-temporal stage operates on a bounded multi-image context per query rather than the full episode, since high-resolution VLM frames produce a large number of visual tokens and dense sampling of long episodes is impractical under a fixed inference budget. This is a property the STaR-VL pipeline shares with other multi-image VLM evaluations, and the curriculum is compatible with relaxations of this constraint — token compression, hierarchical visual encoding, or learned frame selection — which we leave to future work.

Evaluation scope. SpaMEM (Liao et al., 2026) is a procedurally-generated indoor environment based on dynamic ProcTHOR; its visual distribution is bounded to indoor object arrangements. Our four-benchmark out-of-distribution evaluation (VSI-Bench, MMBench-Spatial, EgoSchema, TempCompass) provides initial evidence that the gains transfer beyond the training distribution, but extending STaR-VL to outdoor navigation, manipulation, and large-scale real-world video remains an open empirical question.

Single backbone family. All STaR-VL variants in this paper are built on Qwen2.5-VL. We chose this family because it provides matched 3B/7B variants, native multi-image support, and a well-understood instruction-tuning base. Whether the curriculum produces comparable gains on other backbone families (e.g., InternVL3.5, LLaVA-OneVision) is a natural follow-up; the procedure itself is backbone-agnostic by construction.

Temporal metadata and control resolution. The evaluated observations include explicit time tags. Test-time tag perturbations conflate lost temporal information with input distribution shift; training without tags is not evaluated. Joint-LoRA comparisons use common task data and backbone, but do not separately isolate every architectural and optimization choice. The reported controls are point estimates, and do not establish seed-averaged statistical significance. The corrected 3B aggregate is reported without reconstructing an unavailable task-level vector.

B.2 Broader Societal Impact

STaR-VL improves a small (3B-parameter) vision-language model's ability to perceive, remember, and reason about objects and events as a scene evolves over time. We see two principal categories of broader impact.

Potential benefits. Improved visual-temporal memory in a small backbone may support assistive vision and embodied systems operating under constrained inference resources. The present evaluation establishes benchmark improvements rather than deployment reliability or measured energy savings. Practical applications require task-specific validation, including the grounding and temporal-localization failures identified here.

Risks and mitigations. Vision-language models with stronger temporal memory are dual-use technology. The same capability that enables an assistive robot to remember which medication has been moved can also be applied to persistent visual surveillance, behavioral profiling, and tracking individuals across time and locations from passive video — applications where the privacy externalities are substantial and disproportionately fall on populations with the least ability to opt out. We note three properties of STaR-VL that bear on this risk. First, our training environment (procedurally-generated indoor scenes) and our supervision are object-centric, not person-centric: the model is supervised to track *objects* (cups, books, fridges) through time, not faces or people. Second, the released artifacts are LoRA adapters on top of public Qwen2.5-VL weights, which means anyone deploying STaR-VL inherits the same person-perception failure modes (face-recognition errors, demographic bias) as the underlying backbone — improvements at the temporal layer do not correct underlying biases at the visual-recognition layer, and downstream users should not assume they do. Third, the effectiveness of the curriculum on smaller models reduces the cost of *any* downstream deployment, which by the same logic also reduces the cost of misuse. Standard mitigations apply: deployment in person-tracking contexts should require explicit consent and purpose limitation, downstream users should evaluate fairness on the specific population and tasks of interest rather than relying on aggregate benchmark scores, and the released models should be accompanied by clear documentation of their evaluation scope (object-centric indoor reasoning) so they are not silently repurposed for higher-risk settings.

We do not see a category of harm that is unique to STaR-VL relative to existing open VLMs of comparable capability; the capability-aligned RL recipe primarily redistributes who can access this capability rather than introducing new capability classes.

B.3 Future Work

Several directions naturally extend this work: **(i)** relaxing the bounded-visual-context constraint through learned frame selection or token compression compatible with the STaR-VL pipeline; **(ii)** verifying backbone-agnosticism by re-running the curriculum on InternVL3.5, LLaVA-OneVision, and other families; **(iii)** extending to multi-turn embodied settings where temporal memory must be maintained across active exploration rather than at the end of a fixed observation sequence; **(iv)** systematic evaluation of fairness and demographic robustness when STaR-VL is applied to person-centric video, before any production deployment in such settings.

The next methodological priority is to separate the effects of memory architecture, stage order, and optimization under matched training budgets, alongside stronger evaluation of fine-grained grounding and temporal localization.

C Details of STaR-VL

C.1 Per-Adapter Capability-Aligned Reward Formulation

We provide the full specification of the three adapter-specific capability-aligned rewards used in STaR-VL, each designed to score the capability targeted by its adapter: the *Perception Reward* $\mathcal{R}^{(\mathrm{P})}$ scores per-frame visual grounding; the *Memory Reward* $\mathcal{R}^{(\mathrm{M})}$ scores temporal memory in isolation from perception (with symbolic ground-truth observations); and the *Spatio-Temporal Reward* $\mathcal{R}^{(\mathrm{V})}$ scores cross-time spatial reasoning over visual sequences. Each reward is keyed by output structure type τ and selects a verifiable scoring function from a shared library of capability-aligned branches.

Shared scoring library. All three adapters draw from the same six-branch library, each producing a scalar in $[0, 1]$:

$$B(\hat{y}, y^\star \mid \tau) = \begin{cases} \mathrm{F1}(\hat{S}, S^\star) & \tau \in \{\text{set}, \text{list}\} \\ \mathrm{F1}(\mathrm{keys}(\hat{D}), \mathrm{keys}(D^\star)) & \tau = \text{dict_keys} \\ \dfrac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \mathrm{IoU}(\hat{b}_c, b_c^\star) & \tau = \text{bbox} \\ 1 - \mathrm{NED}(\hat{p}, p^\star) & \tau = \text{trajectory} \\ \mathbf{1}[\hat{n} = n^\star] & \tau = \text{count} \\ \max\!\big(0,\ 1 - |\hat{d} - d^\star|/d^\star\big) & \tau = \text{depth} \end{cases} \tag{5}$$

where $\hat{S}, S^\star$ are predicted and reference sets; $\hat{D}, D^\star$ are dictionaries evaluated via key sets; $\hat{b}_c, b_c^\star$ are bounding boxes for category $c \in \mathcal{C}$; $\hat{p}, p^\star$ are symbolic trajectories evaluated by normalized edit distance (NED); $\hat{n}, n^\star$ are counts; and $\hat{d}, d^\star$ are scalar depth values.

Per-adapter rewards. Each adapter targets a different capability and accordingly invokes a different subset of the scoring library:

$$\mathcal{R}^{(\mathrm{P})}(\hat{y}, y^\star \mid \tau) = B(\hat{y}, y^\star \mid \tau), \quad \tau \in \{\text{list}, \text{bbox}, \text{depth}, \text{count}\}, \tag{6}$$

$$\mathcal{R}^{(\mathrm{M})}(\hat{y}, y^\star \mid \tau) = B(\hat{y}, y^\star \mid \tau), \quad \tau \in \{\text{list}, \text{dict_keys}, \text{bbox}\}, \tag{7}$$

$$\mathcal{R}^{(\mathrm{V})}(\hat{y}, y^\star \mid \tau) = B(\hat{y}, y^\star \mid \tau), \quad \tau \in \{\text{list}, \text{dict_keys}, \text{bbox}\}. \tag{8}$$

The *Perception Reward* $\mathcal{R}^{(\mathrm{P})}$ scores per-frame visual grounding via the perception suite (SOR, VGL, DPE, CC, IC, RSR), invoking the list, bbox, depth, and count branches. The *Memory Reward* $\mathcal{R}^{(\mathrm{M})}$ scores temporal memory via symbolic-input tasks (SOR-M, VGL-M, RSR-M, QA-dict), invoking the list, dict_keys, and bbox branches; the symbolic input regime decouples the reward from perception noise so it measures memory capability in isolation. The *Spatio-Temporal Reward* $\mathcal{R}^{(\mathrm{V})}$ scores cross-time spatial reasoning via multi-image versions of the same temporal tasks, intentionally invoking the same three branches as $\mathcal{R}^{(\mathrm{M})}$ so that the $\Delta_{\mathrm{M}} \to \Delta_{\mathrm{V}}$ transfer learns only modality bridging rather than re-learning task formats.

Format reward. A format reward $R_{\mathrm{fmt}} \in \{0, 1\}$ is applied to every adapter to enforce syntactic validity of structured outputs. The final reward at stage $s \in \{\mathrm{P}, \mathrm{M}, \mathrm{V}\}$ is:

$$R_{\mathrm{total}}^{(s)} = \mathcal{R}^{(s)} + \lambda_{\mathrm{fmt}} R_{\mathrm{fmt}}. \tag{9}$$

C.2 Per-Adapter Advantage and Reference Anchoring

This section consolidates the advantage estimators and reference-anchored regularizers used by all three adapters, organized to make their cross-stage differences explicit.

Advantage estimators. The three adapters use closely related but distinct advantage forms, chosen to match the variance and noise profile of each capability suite:

$$A_k^{(\mathrm{P})} = \frac{R^{(k)} - \mu_R}{\sigma_R + \epsilon}, \tag{10}$$

$$A_t^{(\mathrm{M})} = \mathrm{clip}\big(R^{(\mathrm{M})}(\hat{y}_t, y_t^\star \mid \tau(t)) - 0.5,\ -0.3,\ +0.5\big), \tag{11}$$

$$A_t^{(\mathrm{V})} = R^{(\mathrm{V})}(\hat{y}_t, y_t^\star \mid \tau(t)) - \bar{R}_{\tau(t)}^{\mathrm{EMA}}, \tag{12}$$

where μ_R, σ_R are the mean and standard deviation of rewards within the K-sized sampled group at stage P, and $\bar{R}_{\tau(t)}^{\mathrm{EMA}}$ is a per-task exponential moving average baseline updated at every step at stage V. The clipping range $[-0.3, +0.5]$ at stage M caps the negative tail to prevent gradient collapse on early-training failures while preserving strong positive signals; the fixed baseline 0.5 avoids the EMA-zeroing pathology observed when high-reward tasks dominate the running average. Per-task baselines further reduce gradient variance when reward magnitudes differ across capabilities, addressing a known limitation of group-only baselines in heterogeneous reward settings (Zhou et al., 2026b).

Reference-anchored regularizers. Let $d_t = \log \pi_{\text{ref}}(\hat{y}_t \mid \cdot) - \log \pi_\theta(\hat{y}_t \mid \cdot)$ denote the per-token log-probability gap between the reference and policy. The three adapters apply different anchoring forms:

$$\mathcal{R}_{\text{ref}}^{(\text{P})} = \mathbb{E}_t\big[\exp(d_t) - d_t - 1\big], \tag{13}$$

$$\mathcal{R}_{\text{ref}}^{(\text{M})} = \mathbb{E}_t\big[d_t^2\big], \tag{14}$$

$$\mathcal{R}_{\text{ref}}^{(\text{V})} = \mathbb{E}_t\big[d_t^2\big] \;\text{ with } \pi_{\text{ref}} = \pi_{\Theta_{\text{ref}}}. \tag{15}$$

Eq. (13) is the unbiased KL estimator, $\mathbb{E}\big[\exp(d) - d - 1\big] = \mathrm{KL}(\pi_{\text{ref}} \,\|\, \pi_\theta)$. Eqs. (14)–(15) use the squared log-probability surrogate, which (i) vanishes at $\pi_\theta = \pi_{\text{ref}}$, (ii) produces gradients that pull the policy back toward the reference, and (iii) is computationally simpler and numerically more stable than the $\exp(d)$ form on long generated sequences. We adopt the simpler L2 surrogate at stages M and V because their LoRA-constrained, longer-output settings make the asymptotic difference negligible while avoiding $\exp$ overflow on heavy-tailed log-ratio samples. At stage V, the reference $\pi_{\Theta_{\text{ref}}}$ is the merged adapter policy of Eq. (26), not the pretrained backbone.

C.3 Perceptual Grounding: Training Objective Details

This section provides the full SFT and RL objective details for the Perceptual Grounding adapter Δ_{P} summarized in Sec. 3.2.

SFT objective. SFT for Δ_{P} uses the autoregressive negative log-likelihood:

$$\mathcal{L}_{\text{SFT}}^{(\text{P})} = -\sum_{i=1}^{|y^\star|} \log \pi_\theta(y_i^\star \mid x, q, y_{<i}^\star). \tag{16}$$

Group-relative advantages. For GRPO, we sample K predictions per prompt, compute capability-aligned rewards $R^{(k)} = \mathcal{R}^{(\text{P})}(\hat{y}^{(k)}, y^\star \mid \tau)$ via Eq. (6), and normalize them into group-relative advantages A_k as in Eq. (10). These advantages are plugged into Eq. (1) of the main paper.

C.4 Recurrent Memory Module M_ϕ and Prompting Modes

This section provides the architectural details of the recurrent memory module M_ϕ summarized in Sec. 3.2, including the attention pooling, GRU recurrence, token projection, and the three prompting modes used at training and inference time.

Attention pooling and recurrent update. Let $x_t \in \mathbb{R}^{L_t \times d}$ denote the token embeddings of the symbolic observation at time t, and $h_{t-1} \in \mathbb{R}^{d_h}$ the previous recurrent state. We first apply attention pooling with a learnable query $q \in \mathbb{R}^d$:

$$\alpha_t = \mathrm{softmax}\left(\frac{q^\top x_t^\top}{\sqrt{d}}\right), \tag{17}$$

$$\tilde{x}_t = \sum_{i=1}^{L_t} \alpha_{t,i} x_{t,i}. \tag{18}$$

The pooled representation updates the recurrent state via GRU and is augmented with a learnable temporal embedding τ_t:

$$\hat{h}_t = \mathrm{GRU}(\tilde{x}_t, h_{t-1}) + \tau_t. \tag{19}$$

Memory-token projection. The recurrent state is projected into K virtual memory tokens via a per-slot linear projection:

$$M_t = \Big[W_k\, \phi(\hat{h}_t) + b_k\Big]_{k=1}^{K} \in \mathbb{R}^{K \times d}, \tag{20}$$

where $\phi(\cdot)$ is a non-linear activation and $\{W_k, b_k\}_{k=1}^K$ are learnable per-slot parameters.

Prompting modes. Text-conditioned Memory supports three prompting modes for the input context z_t of the LoRA-adapted policy, depending on whether the model conditions on the raw observation

history, the compressed memory tokens, or both:

$$z_t^{\text{lora}} = [\text{OBS-HISTORY}] \oplus q_t, \tag{21}$$

$$z_t^{\text{mm}} = [M_t] \oplus q_t, \tag{22}$$

$$z_t^{\text{hyb}} = [M_t] \oplus [\text{OBS-HISTORY}] \oplus q_t. \tag{23}$$

The hybrid mode is the default configuration during Δ_{M} training.

Choice of K. We fix $K = 64$ throughout, following persistent memory interfaces for embodied reasoning (Liu et al., 2025) where K in the 32–128 range is standard. The choice trades off representational capacity against the prefix-token cost incurred at every multi-image query during Δ_{V} inference; a systematic sweep over K is left to future work.

C.5 Text-conditioned Memory: Training Objective Details

This section provides the SFT and RL objective details for Δ_{M} summarized in Sec. 3.2.

SFT objective. The SFT loss for Δ_{M} takes the same autoregressive form as Eq. (16), with the input x substituted by z_t from one of the three prompting modes (Eqs. (21)–(23)).

Clipped fixed-baseline advantage. Stage M uses a fixed-baseline advantage with clipping (Eq. (11)). The fixed baseline 0.5 avoids the EMA-zeroing pathology observed when high-reward tasks dominate the running average, and the clip range $[-0.3, +0.5]$ caps the negative tail to prevent gradient collapse on early-training failures while preserving strong positive signals.

Δ_{M}-specific stabilization. Two heuristics stabilize Δ_{M} training. **(i) Answer-prefix policy gradient.** Policy-gradient terms are restricted to the first $N_{\text{prefix}} = 30$ generated tokens (the JSON answer prefix), avoiding noisy long-tail token gradients on whitespace and trailing structure tokens that are unrewarded:

$$\mathcal{L}_{\text{P-loss}}^{(\text{M})} = -A_t \cdot \frac{1}{N_{\text{prefix}}} \sum_{i=1}^{N_{\text{prefix}}} \log \pi_\theta(\hat{y}_{t,i} \mid z_t, \hat{y}_{t,<i}). \tag{24}$$

(ii) Skip-on-failure update. Updates are skipped when the squared log-probability gap $\mathcal{R}_{\text{ref}}^{(\text{M})}$ exceeds a threshold τ_{KL} *and* reward is zero, preventing further drift on already-failed examples whose gradients are dominated by reference-anchoring noise rather than capability signal:

$$\text{skip_update} = \mathbb{1}\left[\mathcal{R}_{\text{ref}}^{(\text{M})} > \tau_{\text{KL}} \ \wedge \ R^{(\text{M})} = 0\right]. \tag{25}$$

Reference penalty. The squared log-probability reference penalty $\mathcal{R}_{\text{ref}}^{(\text{M})}$ is given by Eq. (14). Combining Eqs. (24) and (14) recovers Eq. (2) of the main paper.

C.6 Visual Spatio-Temporal Memory: Construction Details

This section provides the adapter-composition, visual-context construction, and SFT objective details for Δ_{V} summarized in Sec. 3.2.

Adapter merging and reference policy. Let Θ_0 denote the pretrained backbone, Δ_{P} the Perceptual Grounding adapter, and Δ_{M} the Text-conditioned Memory adapter. We merge the previously learned adapters into the base model:

$$\Theta_{\text{ref}} = \text{Merge}(\Theta_0, \Delta_{\text{P}}, \Delta_{\text{M}}), \tag{26}$$

which serves as both the initialization and the reference policy for Δ_{V}. We then attach a new trainable adapter Δ_{V} and optimize only this adapter:

$$\Theta^{(\text{V})} = \Theta_{\text{ref}} \oplus \Delta_{\text{V}}. \tag{27}$$

Per-task EMA advantage. Stage V uses a per-task EMA-baseline advantage (Eq. (12)). The EMA baseline $\bar{R}_{\tau(t)}^{\text{EMA}}$ is updated at every step with smoothing factor α_{EMA}:

$$\bar{R}_{\tau(t)}^{\text{EMA}} \leftarrow \alpha_{\text{EMA}} \cdot \bar{R}_{\tau(t)}^{\text{EMA}} + (1 - \alpha_{\text{EMA}}) \cdot R^{(\text{V})}. \tag{28}$$

Per-task baselines reduce gradient variance when reward magnitudes differ across the four V-stage tasks, addressing a known limitation of group-only baselines in heterogeneous reward settings (Zhou et al., 2026b).

Visual temporal context. For each query at time step t, we construct a bounded visual history:

$$V_{\leq t} = \{I_{s_1}, I_{s_2}, \ldots, I_{s_B}\}, \tag{29}$$

where I_{s_j} are frames sampled from the episode history within an image budget B. Each frame is paired with an explicit temporal tag (e.g., `[Time 0]`, `[Time t]`) to preserve temporal ordering in the multi-image prompt.

SFT objective. Under teacher forcing, Δ_{V} uses the same structured-generation objective as Text-conditioned Memory:

$$\mathcal{L}_{\mathrm{SFT}}^{(\mathrm{V})} = -\sum_{i=1}^{|y_t^\star|} \log \pi_{\Theta^{(\mathrm{V})}}(y_{t,i}^\star \mid V_{\leq t}, q_t, y_{t,<i}^\star). \tag{30}$$

The corresponding RL objective in the main paper (Eq. (4)) uses the reference penalty $\mathcal{R}_{\mathrm{ref}}^{(\mathrm{V})}$ defined in Eq. (15), anchored to the merged Θ_{ref}.

D Additional Experiments

D.1 Evaluation Setup Reference

Table 4 maps capability groups to SpaMEM tasks and reward branches; Table 5 describes adapter configurations.

Table 4: Capability-to-task mapping across SpaMEM and OOD benchmarks.

Cap.	SpaMEM	Ego	MMB	VSI	Temp
Recall	SOR, LOR	QA			
Locate	VGL, TL		Loc.; Rel.		
Track	RSR, STT			Ord.	Ord.
Detect	SED$^{+/-}$		Rel.	Cnt.	
Accumulate	CSR$^{+/-}$			Dist.	Act.

Datasets: SpaMEM (Liao et al., 2026), EgoSchema (Mangalam et al., 2023), MMBench (Liu et al., 2024b), VSI-Bench (Yang et al., 2025), TempCompass (Liu et al., 2024c). Tasks: QA (question answering), Loc. (localization), Rel. (relation), Ord. (ordering), Cnt. (counting), Dist. (distance), Act. (action).

Table 5: STaR-VL naming scheme.

Name	Meaning
PMV	full: $\Delta_{\mathrm{P}}, \Delta_{\mathrm{M}}, \Delta_{\mathrm{V}}$ all RL
PMv	$\Delta_{\mathrm{P}}, \Delta_{\mathrm{M}}$ RL; Δ_{V} SFT
pmv	all SFT only
pm	$\Delta_{\mathrm{P}}, \Delta_{\mathrm{M}}$ SFT only
P/M/V	single RL adapter

Strings over $\{P, M, V\}$ denote adapters $\{\Delta_{\mathrm{P}}, \Delta_{\mathrm{M}}, \Delta_{\mathrm{V}}\}$; uppercase = SFT+RL, lowercase = SFT only.

Data split and objectives. The training split contains 10 houses, 6,011 episode queries, and 22,592 images. The held-out split contains 5 different houses, 2,742 queries, and 10,304 images. Within each stage, SFT and RL draw from the same underlying examples; stage-specific routing and representation changes follow the method specification. Stage P trains the perception adapter, stage M trains the memory adapter and recurrent memory module, and stage V reuses the memory module unchanged while training the new visual adapter.

Visual and symbolic histories. The main STaR-VL-3B and 7B visual runs use $B = 3$ and $B = 2$ frames, respectively. Visual history is uniformly subsampled and paired with explicit temporal tags. Symbolic history is a ground-truth scene-state representation and may cover a longer history. Accordingly, cross-regime comparisons do not enforce identical information content. The $B = 3, 5, 8, 16$ visual sweep and a full-history versus five-step symbolic control are reported in Appendix F.

Inference and controls. The reported evaluations use greedy decoding and structured-answer parsing, with parsing failures scored as zero. The Train-aligned prompt is the reference condition. The structured memory-table prompt is evaluated on both the untrained and trained 3B model; six additional wrappers test the untrained backbone as well. The memory-system comparisons retain the same episodes, prompts, history budget, parsers, and aggregation procedure within their matched conditions. Broad cross-family results remain descriptive, because input representation and model-specific interfaces are not fully matched interventions.

D.2 PER-FRAME PERCEPTION RESULTS

We evaluate STaR-VL on the five per-frame perception tasks aggregated into the P-score — SOR (object identification), VGL (visual grounding / localization), RSR (relative spatial relation), CC (category count), and IC (instance count) — and report the unweighted mean. Table 6 compares STaR-VL-3B against three proprietary models, eight open-source VLMs spanning 3B to 235B parameters, and five multimodal baselines built on alternative backbones. We report two STaR-VL variants: **STaR-VL-3B-P** (Δ_{P} trained with SFT followed by GRPO) and **STaR-VL-3B-p** (SFT only, no RL).

STaR-VL-3B-P leads every baseline on the P-score. At 3B parameters, STaR-VL-3B-P attains **P-score = 50.4**, decisively outperforming every closed-source baseline — ChatGPT-5.4 (35.1), ChatGPT-4o (30.8), Gemini-2.5-flash (32.6) — and every open-source baseline regardless of scale, including Qwen3-VL-235B (36.2), Qwen2.5-VL-72B (35.8), and InternVL3-8B (23.2). The lead over the strongest non-STaR baseline is $+14.2$ points. The transformation is sharper still on the underlying base: Qwen2.5-VL-3B scores only 11.7, so STaR-VL-3B-P lifts the same backbone by $+38.7$ absolute — evidence that capability-aligned RL on perception, not parameter count, is what closes the gap to frontier-scale baselines on this suite.

Object identification (SOR) drives the headline gain. The most pronounced improvement is on SOR, the per-frame object-identification metric. STaR-VL-3B-P reaches **97.4**, while no non-STaR baseline exceeds 53: the strongest is Qwen2.5-Omni-7B at 52.5, followed by Qwen2.5-VL-72B (48.3), Qwen3-VL-235B (42.4), and the GPT/Gemini family in the 40–42 range. STaR-VL leads the strongest non-STaR baseline by $+44.9$ absolute on SOR alone. Because SOR directly measures whether the model lists the correct set of objects on a given receptacle — the most basic capability Δ_{P} is supervised on — this gap is the cleanest evidence that capability-aligned RL on a small base model produces task-specific competence that scaling alone does not.

Visual grounding (VGL) is the second-largest gain in absolute reward terms. STaR-VL-3B-P scores VGL = **13.7**, compared with the 3B Qwen base at 1.2 — a $+12.5$ absolute lift via Δ_{P}, of which $+3.1$ is attributable to GRPO over SFT alone (see RL contribution below). Among non-STaR baselines, only Qwen3-VL-235B (30.7) and Qwen2.5-VL-72B (25.7) exceed STaR-VL on VGL, both at $\geq 25\times$ the parameter count; every Qwen2.5-VL variant under 32B sits below 9. VGL is the only metric measuring spatial coordinate prediction, and the rapid acquisition of a non-trivial coordinate-output capability on a 3B backbone, where the base model itself produces near-zero scores, is the clearest activation effect of explicit IoU-based reward.

Relative spatial relation (RSR) shows the strongest scale-vs.-supervision gap. STaR-VL-3B-P reaches **70.4** on RSR, exceeding every non-STaR baseline regardless of scale — ChatGPT-5.4 (53.9), Gemini-2.5-flash (50.3), Qwen2.5-VL-72B (48.7), Qwen3-VL-235B (47.4) — by $+16.5$ to $+23.0$ points. RSR is a binary left/right relational judgment whose absolute scores are sensitive to position priors when models lack visual grounding, but STaR-VL-3B-P attains its RSR result jointly with the highest VGL score among $<$30B models, indicating that the relational signal is grounded rather than guessed.

RL adds consistent gains across every learnable metric. Comparing Δ_{P} (SFT+GRPO) against Δ_{p} (SFT only), we observe positive deltas across the suite: SOR $+5.5$ ($91.9 \rightarrow 97.4$), VGL $+3.1$ ($10.6 \rightarrow 13.7$), RSR $+4.0$ ($66.4 \rightarrow 70.4$), CC $+2.2$ ($18.8 \rightarrow 21.0$), IC $+1.7$ ($17.1 \rightarrow 18.8$). The aggregate P-score improves by $+3.7$ ($46.7 \rightarrow 50.4$). The deltas are larger on capabilities the reward function targets directly (SOR, VGL, RSR through structured F1/IoU scoring) and smaller on the count metrics (CC/IC) where SFT already saturates — the same capability-aligned-reward pattern that holds for Δ_{M} and Δ_{V} in Sec. 4.3 and Sec. 4.2.

DPE excluded from the aggregate. Depth prediction error (DPE) saturates near 100 across the field — most models cluster in the 89–100 range, including the 3B Qwen base. Because DPE does not discriminate between methods, we do not aggregate it into the headline P-score; STaR-VL-3B-P shows no regression on this metric relative to the base (-1.2 absolute), confirming that capability-aligned RL on the other five tasks does not damage depth competence.

Takeaway. STaR-VL-3B-P establishes that capability-aligned RL on a 3B backbone surpasses every evaluated baseline up to 235B parameters on the perception suite, with the largest gains concentrated on SOR ($+44.9$ over the strongest non-STaR baseline) and the only sub-10B coordinate-prediction

Model	Identify	Localize	Relate	Count		P-score
	SOR	VGL	RSR	CC	IC	
Proprietary Models						
ChatGPT-5.4 (OpenAI, 2025)	41.8	18.0	53.9	22.2	31.3	35.1
ChatGPT-4o (OpenAI, 2024)	39.9	12.8	49.4	18.2	23.9	30.8
Gemini-2.5-flash (Google DeepMind, 2025)	40.9	8.1	50.3	26.1	**36.4**	32.6
Open-source Models						
Qwen3-VL-235B-a22b-Instruct (Qwen Team, 2025)	42.4	**30.7**	47.4	23.9	25.0	36.2
Qwen3-VL-32B-Instruct (Qwen Team, 2025)	40.5	10.4	46.6	16.5	17.6	28.6
Qwen3-VL-8B-Instruct (Qwen Team, 2025)	29.5	6.1	10.6	25.0	16.5	16.7
Qwen2.5-VL-72B-Instruct (Bai et al., 2025)	48.3	25.7	48.7	22.7	18.2	35.8
Qwen2.5-VL-32B-Instruct (Bai et al., 2025)	32.4	8.3	45.1	31.3	22.7	28.2
Qwen2.5-Omni-7B (Xu et al., 2025)	52.5	10.2	33.2	16.5	15.9	28.0
Qwen2.5-VL-7B-Instruct (Bai et al., 2025)	39.4	5.5	8.3	19.3	9.1	16.9
Qwen2.5-VL-3B-Instruct (Bai et al., 2025)	26.4	1.2	2.4	14.2	19.3	11.7
LLaVA-Interleave-Qwen-7B (Li et al., 2024b)	33.5	6.8	12.3	5.0	10.2	15.1
LLaVA-v1.6-Mistral-7B (Liu et al., 2024a)	37.8	6.6	9.7	30.7	11.9	18.9
LLaVA-OneVision-Qwen2-7B (Li et al., 2024a)	27.7	7.2	11.7	15.4	18.2	15.9
InternVL3.5-8B (Wang et al., 2025b)	37.9	7.4	24.2	12.6	21.6	21.7
InternVL3-8B (Zhu et al., 2025)	34.6	10.1	23.4	**32.1**	17.6	23.2
Ours						
STaR-VL-3B-P	**97.4**	13.7	**70.4**	21.0	18.8	**50.4**
STaR-VL-3B-p	91.9	10.6	66.4	18.8	17.1	46.7

Table 6: Perception-centric spatial reasoning evaluation (P-score regime). All metrics are scaled to $[0, 100]$. Best result in each column is in bold and second-best is underlined.

capability of comparable strength to $\geq$32B baselines (VGL). Ablating GRPO removes a consistent 2–5.5 points per metric. This pattern foreshadows the larger capability-decomposed gains observed in the memory and visual-fusion suites (Sec. 4.3, Sec. 4.2).

D.3 Performance-Gap Analysis

The P, M, and V evaluations probe per-frame perception, symbolic-history queries, and visual-history queries. Tables 7 and 8 collect the corresponding reported differences. Symbolic histories contain ground-truth scene state, including bounding-box information, while visual histories require extracting that state from bounded RGB observations. These differences therefore summarize operational behavior rather than a pure modality intervention.

Per-frame recall versus historical recall. For the baseline rows with available task-level values, per-frame SOR exceeds visual-history SOR-M. STaR-VL's perception and full configurations score 97.4 and 98.5, respectively, giving a -1.1 difference. This comparison highlights strong historical object recall in the trained system. Relation and grounding tasks show different patterns, so object recall alone does not establish uniformly preserved visual-temporal capability.

Corrected aggregate and descriptive plots. Qwen2.5-VL-3B's corrected V-score is 21.85. Its corrected per-task vector is not available in the supplied evaluation summary, so dependent task-level cells are left unreported. Figure 2(a) uses its corrected aggregate with the reported M-score of 42.6; panel (b) excludes that model. Both plots are reconstructed from the displayed score tables, with baseline-only fits and STaR-VL shown separately. The M$-$V axis shares a V term with the vertical V-score axis, creating mathematical coupling. Cross-model correlations, as well as differences in coverage and state quality, prevent a causal interpretation of the fitted trends.

Model	SOR (Recall)			RSR (Spatial relation)		
	P	V (SOR-M)	$\Delta_{\text{P-V}}$	P	V (RSR-M)	$\Delta_{\text{P-V}}$
Proprietary Models						
ChatGPT-5.4 (OpenAI, 2025)	41.8	19.1	+22.7	53.9	38.4	+15.5
ChatGPT-4o (OpenAI, 2024)	39.9	18.4	+21.5	49.4	35.5	+13.9
Gemini-2.5-flash (Google DeepMind, 2025)	40.9	14.4	+26.5	50.3	37.6	+12.7
Open-source Models						
Qwen3-VL-235B (Qwen Team, 2025)	42.4	17.0	+25.4	47.4	37.7	+9.7
Qwen3-VL-32B (Qwen Team, 2025)	40.5	29.4	+11.1	46.6	35.6	+11.0
Qwen3-VL-8B (Qwen Team, 2025)	29.5	1.7	+27.8	10.6	2.2	+8.4
Qwen2.5-VL-72B (Bai et al., 2025)	48.3	18.4	+29.9	48.7	40.9	+7.8
Qwen2.5-VL-32B (Bai et al., 2025)	32.4	19.2	+13.2	45.1	38.4	+6.7
Qwen2.5-Omni-7B (Xu et al., 2025)	52.5	4.9	+47.6	33.2	28.1	+5.1
Qwen2.5-VL-7B (Bai et al., 2025)	39.4	2.0	+37.4	8.3	0.4	+7.9
Qwen2.5-VL-3B (Bai et al., 2025)	26.4	—	—	2.4	—	—
LLaVA-Interleave-Qwen-7B (Li et al., 2024b)	33.5	0.9	+32.6	12.3	1.8	+10.5
LLaVA-v1.6-Mistral-7B (Liu et al., 2024a)	37.8	1.6	+36.2	9.7	1.2	+8.5
LLaVA-OneVision-Qwen2-7B (Li et al., 2024a)	27.7	0.3	+27.4	11.7	1.8	+9.9
InternVL3.5-8B (Wang et al., 2025b)	37.9	14.0	+23.9	24.2	6.4	+17.8
InternVL3-8B (Zhu et al., 2025)	34.6	20.4	+14.2	23.4	15.3	+8.1
Ours						
STaR-VL-3B (P / Full)	**97.4**	**98.5**	**−1.1**	**70.4**	**54.3**	+16.1

Table 7: **Perception-to-memory gap** on the two tasks (SOR, RSR) that share a definition between the P-score (perception) and V-score (multi-frame visual memory) suites. $\Delta_{\text{P-V}}$ is computed as P − V on each task. A large positive gap indicates that the model can identify objects or relations from a perception-only input but loses that capability under a multi-frame temporal-memory query. The STaR-VL row pairs STaR-VL-3B-P (perception adapter) against STaR-VL-3B (Full) (perception+memory+visual fusion). Best (smallest) gap per task in bold. Dashes indicate an unavailable corrected task-level breakdown for Qwen2.5-VL-3B.

Direct controls guide interpretation. Increasing STaR-VL's frame budget recovers 4.80 V-score points. Self-generated persistent memory also improves both evaluated backbones, while oracle state yields further gains on InternVL3-8B. These interventions support complementary roles for coverage and accessible scene state. The same-backbone training comparison remains the direct evidence for the practical benefit of the complete STaR-VL framework over the evaluated joint-training and memory alternatives.

D.4 Out-of-Distribution Benchmark Visualization

Figure 4 provides a per-benchmark visualization of the out-of-distribution evaluation results reported in Table 2 of the main paper.

D.5 Adapter Breakdown Analysis

Per-capability breakdown.

Detailed observations. Figure 5 reveals that different adapters contribute to distinct capability groups. PMV consistently dominates across Recall, Locate, Track, and Detect, showing a benefit from the learned composition in this evaluation. In contrast, Accumulate shows smaller differences, suggesting it is less sensitive to long-horizon reasoning.

Capability decomposition. The configurations differ across capability groups. Because components were acquired sequentially, removing an adapter changes the learned composition; the

Model	SOR-M	LOR	VGL-M	TL	RSR-M	STT	SED+	SED−	CSR+	CSR−	Mean
Proprietary Models											
ChatGPT-5.4 (OpenAI, 2025)	+80.8	**+0.1**	+7.9	+19.4	+54.6	+3.7	+9.9	+13.2	+1.1	+4.6	+19.5
ChatGPT-4o (OpenAI, 2024)	+60.4	+30.9	+3.3	+23.3	+57.5	+30.4	+23.5	+8.7	+19.3	+24.1	+28.1
Gemini-2.5-flash† (Google DeepMind, 2025)	+84.5	+7.6	+0.9	**+8.8**	+55.4	+2.2	**+2.7**	**+1.8**	**+0.6**	+1.8	+16.6
Open-source Models											
Qwen3-VL-235B (Qwen Team, 2025)	+82.6	+53.9	+22.3	+40.7	+54.9	+21.0	+21.6	+12.8	+33.3	+20.0	+36.3
Qwen3-VL-32B (Qwen Team, 2025)	+70.3	+55.9	+0.9	+23.8	+57.0	+30.4	+23.7	+19.0	+42.6	+28.4	+35.2
Qwen3-VL-8B† (Qwen Team, 2025)	+88.1	+81.4	+2.7	+78.6	+40.7	**+0.0**	+72.8	+69.2	+36.5	+45.5	+51.6
Qwen2.5-VL-72B (Bai et al., 2025)	+81.2	+33.9	+17.5	+30.8	+52.1	+27.5	+22.0	+22.8	+21.4	+19.2	+32.8
Qwen2.5-VL-32B (Bai et al., 2025)	+79.1	+9.7	+0.2	+29.6	+56.5	+24.6	+16.7	+16.2	+1.1	+5.2	+23.9
Qwen2.5-Omni-7B (Xu et al., 2025)	+82.4	+45.8	+3.3	+8.5	+62.8	+1.1	+21.4	+3.2	+36.1	+22.9	+28.8
Qwen2.5-VL-7B† (Bai et al., 2025)	+84.6	+79.7	+1.3	+70.4	+25.4	**+0.0**	+59.1	+56.1	+33.8	+45.3	+45.6
Qwen2.5-VL-3B (Bai et al., 2025)	—	—	—	—	—	—	—	—	—	—	+20.8
LLaVA-Interleave-Qwen-7B† (Li et al., 2024b)	+46.1	+10.3	**+0.0**	+26.4	+36.6	+7.5	+21.3	+20.1	+8.4	+4.1	+18.1
LLaVA-v1.6-Mistral-7B (Liu et al., 2024a)	+79.3	+53.4	+0.2	+55.3	+55.1	+4.2	+34.7	+48.7	+27.4	+21.2	+38.0
LLaVA-OneVision-Qwen2-7B† (Li et al., 2024a)	+86.4	+55.9	+0.6	+34.3	+36.9	**+0.0**	+31.2	+53.8	+22.1	+41.5	+36.3
InternVL3.5-8B† (Wang et al., 2025b)	+75.5	+68.3	+0.2	+44.6	+58.0	**+0.0**	+34.5	+5.6	+31.1	**+0.2**	+31.8
InternVL3-8B† (Zhu et al., 2025)	+60.8	+19.4	+1.7	+34.7	**+4.3**	**+0.0**	+21.0	+7.8	+11.4	+2.0	+16.3
Ours											
STaR-VL-3B (M / Full)	**+0.2**	+26.8	+4.7	+20.6	+10.0	+13.9	+16.6	+19.4	+24.2	+1.5	**+13.8**

Table 8: **Symbolic–visual differences.** Task columns show M−V and the final column summarizes the reported aggregates. For Qwen2.5-VL-3B the corrected aggregate difference is $42.6 - 21.85 \approx 20.8$; task-level differences are unavailable. STaR-VL compares its memory and full visual configurations (61.4 versus 47.6). These are descriptive comparisons with different coverage and state information; a small difference can coexist with low absolute performance.

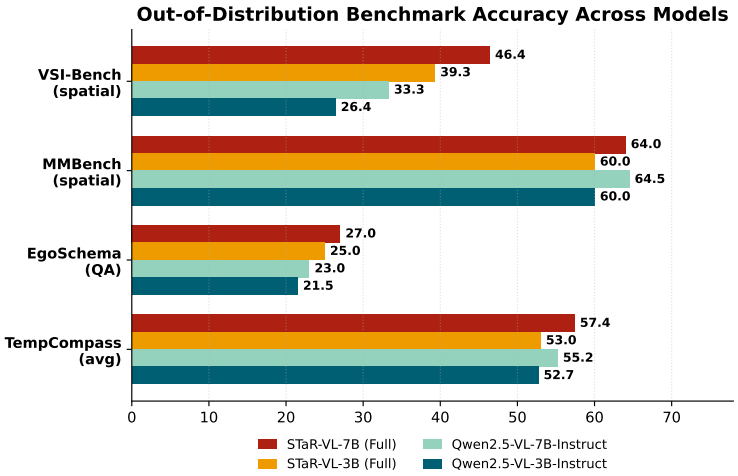

Figure 4: Out-of-distribution benchmark accuracy across four benchmarks and four models. STaR-VL-7B (Full) achieves the highest accuracy on three of four benchmarks (VSI-Bench, EgoSchema, TempCompass), while remaining within 0.5 points of Qwen2.5-VL-7B on MMBench-Spatial.

comparison does not establish a unique assignment of non-overlapping capabilities to individual adapters.

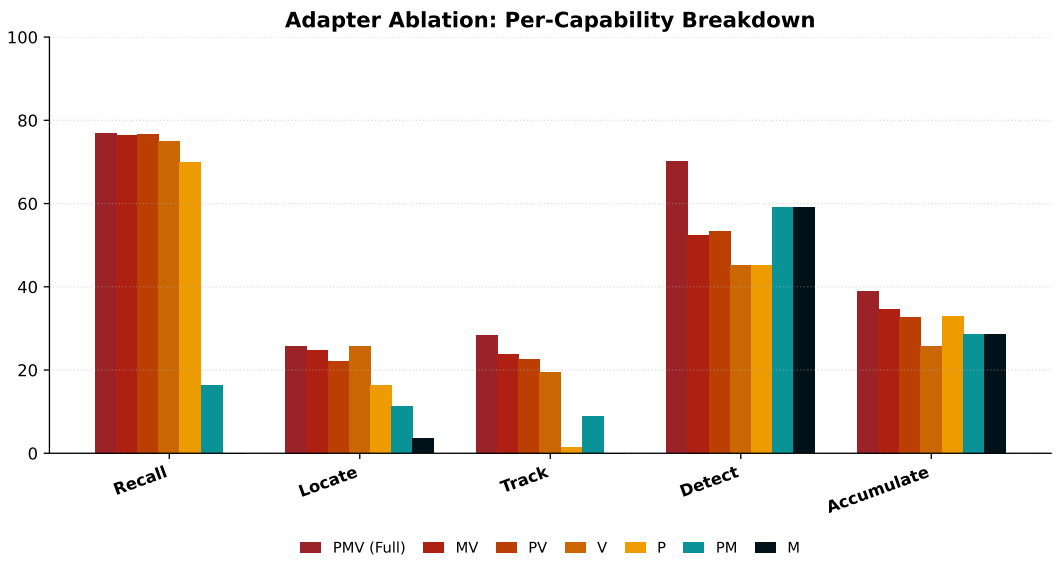

Figure 5: Per-capability breakdown across adapter configurations. Each capability is averaged over its corresponding sub-metrics. PMV dominates Recall, Locate, Track, and Detect, while Accumulate shows smaller gaps due to short-horizon CSR.

Model	Rec.	Loc.	Track	Det.	Acc.	V-score
Ablation (3B)						
PMV	**76.9**	**25.8**	**28.4**	**70.1**	**39.0**	**47.7**
PM	16.3	11.2	8.8	59.0	28.7	24.5
MV	76.5	24.8	23.9	52.4	34.7	42.1
PV	76.7	22.0	22.6	53.3	32.6	41.1
P	70.0	16.3	1.4	45.3	32.9	32.9
M	0.0	3.6	0.0	59.0	28.7	17.9
V	75.0	25.7	19.5	45.3	25.7	37.9

Table 9: Adapter ablation on STaR-VL-3B across five capabilities—RECALL, LOCATE, TRACK, DETECT, and ACCUMULATE—and their macro average (*V-score*). Best results are in bold and second-best are underlined.

Full results. We provide the complete metric breakdown in Table 9.

D.6 PROMPT FORMAT ANALYSIS

This section provides the full reference material and analysis for the prompt-format study summarized in Sec. 4.5 of the main paper (Prompt-format robustness paragraph). We first document the exact prompt templates used during training, evaluation, and the inference-time prompt-format ablation (Sec. D.6.1); we then report the per-capability ablation results and findings (Sec. D.6.2). Italic angle-bracket placeholders such as ⟨r⟩ are filled in at prompt-construction time from per-episode metadata. All templates are reproduced verbatim from our public code release.

D.6.1 PROMPT SPECIFICATIONS

Adapter Δ_P (perception). Δ_P is trained on six per-frame perception tasks, each conditioned on a target receptacle name ⟨r⟩ and exemplar object names ⟨m_1⟩, ⟨m_2⟩. All prompts request a strict JSON output to enable verifiable F1 / IoU / NED reward computation under GRPO.

SOR — per-frame object recognition

```
List all objects on the '⟨r⟩'.  Return ONLY a JSON list.
```

VGL — visual grounding / localization

```
Detect objects on '⟨r⟩'.  Return ONLY a JSON dict
{name:  [x1, y1, x2, y2]}, where coordinates are normalized to [0,1]
relative to image size (e.g., [0.2, 0.3, 0.8, 0.9]).
```

DPE — depth / distance estimation

```
Estimate distance (meters) for objects on '⟨r⟩'.  Return ONLY a JSON
dict
{name:  float}.
```

CC — category count

```
How many categories are on '⟨r⟩'?  Return ONLY an int.
```

IC — instance count

```
How many instances are on '⟨r⟩'?  Return ONLY an int.
```

RSR — relative spatial relation

```
Is '⟨m_2⟩' to the 'left' or 'right' of '⟨m_1⟩'?  Return ONLY a JSON
dict.
```

Adapter Δ_{M} (memory, symbolic history). Δ_{M} is trained on five tasks that condition on a symbolic history buffer $h_{<t}$ and a previous time index $\mathrm{prev}_t = t - 1$.

SOR-M — object recognition from memory

```
From memory of Time ⟨prev_t⟩, list all object TYPES present.  Return
ONLY a JSON list of strings.
```

VGL-M — grounding from memory

```
From memory of Time ⟨prev_t⟩, provide 2D bounding boxes for object TYPES.
Return ONLY a JSON dict
{type:  [x1, y1, x2, y2]}, normalized to [0,1].
```

RSR-M (anchor-neighbors) — relational memory

```
From memory of Time ⟨prev_t⟩, name up to 3 neighbor object TYPES of
'⟨anchor⟩'.  Return ONLY a JSON dict (keys are neighbor types).
```

RSR-M (receptacle-fallback) — containment memory

```
From memory of Time ⟨prev_t⟩, list up to 5 object TYPES that are ON the
receptacle '⟨rec⟩'.  Return ONLY a JSON list of strings.
```

QA-dict (eight-key target) — multi-aspect memory query

```
Focus on target object TYPE '⟨tt⟩'.  Return ONLY a JSON dict with keys:
all_objects_since_time_0, added_since_last_step,
removed_since_last_step, added_since_time_0, removed_since_time_0,
migration_path, first_seen, last_seen.
```

Adapter Δ_{V} (visual fusion, multi-image). Δ_{V} is trained on the same five task templates as Δ_{M}, but conditioned on the actual image-frame buffer rather than a symbolic history. Identity in template wording is intentional: it isolates the modality bridge from the task-format effect, so any performance gap between Δ_{M} and Δ_{V} is attributable to visual-vs-symbolic input rather than prompt re-engineering.

Evaluation prompts. The M-score evaluates the capability of $\Delta_{\mathrm{P}}+\Delta_{\mathrm{M}}$ on a held-out symbolic-history split; eval-time prompts are byte-identical to the training prompts above for SOR-M, VGL-M, RSR-M, and QA-dict. The V-score evaluates the full $\Delta_{\mathrm{P}}+\Delta_{\mathrm{M}}+\Delta_{\mathrm{V}}$ stack on a held-out multi-image split; prompts are byte-identical to the corresponding Δ_{V} training prompts. The V-score additionally exposes a long-horizon evaluation pass that asks a cumulative QA at episode end:

Long-horizon SOR — cumulative recall

```
From memory of Time ⟨past_t⟩, list all object TYPES present.  Return
ONLY a JSON list of strings.
```

Long-horizon QA-dict — cumulative multi-aspect query

```
Focus on target object TYPE '⟨tt⟩'.  Return ONLY a JSON dict with keys:
⟨same eight keys as above⟩.
```

Reusing the same prompt format across train and eval is a deliberate design choice: it eliminates a confounder between prompt distribution shift and capability gap, so M-score → V-score regression cleanly attributes to the modality bridge.

Inference-time prompt-format variants. We compare six prompt-format variants at inference time on the same trained PMV (Full) checkpoint. Each variant is implemented as a thin wrapper around the base task prompt p defined above; only the wrapper text differs between variants. Table 10 summarizes the six wrappers, and the exact wrapper text – as inserted by our `wrap_prompt` routine – is shown below each variant.

Variant	Wrapping mode	Intended effect
Train-aligned	none	Default; matches the training-time prompt distribution.
+CoT	prepend reasoning trigger	Activate chain-of-thought reasoning.
+JSON	append format constraint	Enforce strict JSON output.
+R1	append `<think>`/`<answer>` tags	DeepSeek-R1-style scratchpad-then-answer scheme.
+NoReason	append no-reasoning instruction	Negative control mirroring +CoT; suppresses reasoning.
+Spatial	prepend spatial-attention hint	Prime spatial layout / orientation reasoning.

Table 10: Overview of inference-time prompt-format variants. Each variant wraps the base task prompt p; only Train-aligned leaves p unchanged.

Train-aligned. The unwrapped prompt, identical to the format used during training. This is the default reference and the configuration reported in the main results table. The prompt is the raw p with no wrapper text added.

+CoT. Prepends an explicit chain-of-thought activation phrase before the base prompt, following the dominant convention in CoT prompting (Wei et al., 2022).

+CoT wrapper (prepended to p)

```
Think step by step about what you observe, then answer.
```

+JSON. Appends a strict format constraint after the base prompt, requiring the output to be a single valid JSON object.

+JSON wrapper (appended to p)

```
IMPORTANT: Output must be a single valid JSON. No prose, no markdown.
```

+R1. Wraps the response in DeepSeek-R1 style `<think>`/`<answer>` delimiters (DeepSeek-AI, 2025). The model is instructed to first emit free-form reasoning inside `<think>`, then the final JSON inside `<answer>`; the JSON extractor recovers the answer from the inner span.

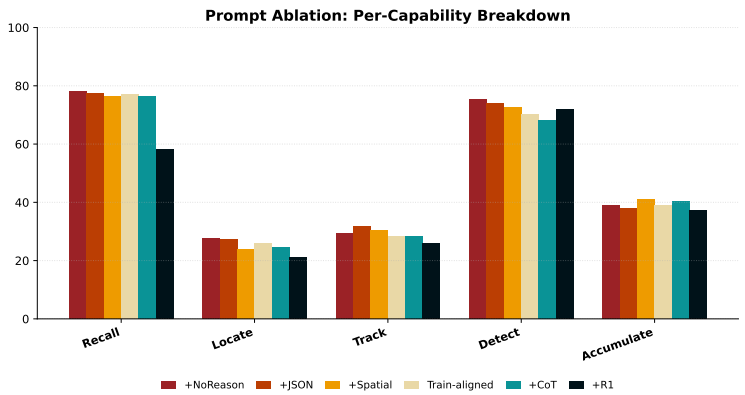

Figure 6: Per-capability breakdown across prompt variants. Differences concentrate on Recall and Detect, while most capabilities remain stable across formats.

+R1 wrapper (appended to p)

```
First, reason inside <think>...</think> tags.  Then provide your final
answer inside <answer>...</answer> tags.  The content of <answer> must
be the JSON specified above.
```

+NoReason. A negative control mirroring +CoT: instead of activating reasoning, this variant explicitly instructs the model to suppress it. Comparing +CoT and +NoReason isolates the effect of CoT activation on a model already trained with structured-answer rewards.

+NoReason wrapper (appended to p)

```
Answer directly without any reasoning or explanation.  Output the
answer only.
```

+Spatial. Prepends a task-specific spatial-attention hint, priming the model to attend to layout, positions, and relative orientations – properties directly relevant to STaR-VL's evaluation tasks.

+Spatial wrapper (prepended to p)

```
Pay close attention to the spatial layout, object positions, and
relative orientations in the scene.
```

Implementation note. All variants share the same `extract_json` routine for parsing model output. For +R1, the extractor is intentionally permissive: it greedily matches the JSON object inside the `<answer>` block, which means a well-formed `<answer>` containing valid JSON is parsed correctly, but malformed or truncated `<answer>` blocks fall through to the parent regex and may match content from inside the `<think>` block by accident. We discuss the consequences of this interaction in the per-capability analysis below.

D.6.2 Per-capability Analysis

We now analyze the per-capability impact of the six variants defined above on the same trained PMV (Full) checkpoint. All variants share identical model weights and evaluation data; only the inference-time prompt template changes.

Robustness within a narrow band. Five of the six variants fall within a 2.3-point V-score range, indicating that STaR-VL's capability is largely encoded in model weights rather than prompt design.

JSON constraint improves hard tasks. The +JSON variant triples performance on STT, the most difficult task, suggesting that structured output reduces hallucination and improves consistency.

CoT is not beneficial under RL tuning. Chain-of-thought prompting slightly degrades performance, while "no-reasoning" improves results. This indicates that RL training already optimizes reasoning behavior.

Prompt	Rec.		Loc.		Track		Det.		Acc.		V-Score
	SOR	LOR	VGL	TL	RSR	STT	SED+	SED-	CSR+	CSR-	
Train-aligned	**98.5**	55.2	6.4	45.2	54.3	2.5	74.0	66.2	29.7	48.3	47.7
+CoT	95.9	56.8	6.5	42.5	54.5	2.4	72.4	64.0	29.2	<u>51.7</u>	47.3
+JSON	<u>97.3</u>	57.8	6.2	<u>48.5</u>	56.0	**7.5**	76.2	<u>71.6</u>	**32.8**	43.0	<u>49.4</u>
+R1	64.6	51.7	<u>6.7</u>	35.2	49.6	2.4	<u>76.7</u>	67.1	27.7	46.5	42.5
+NoReason	97.2	**59.0**	6.7	**48.5**	<u>56.3</u>	2.3	**78.5**	**72.7**	<u>31.3</u>	46.2	**49.6**
+Spatial	95.0	<u>58.0</u>	**7.0**	40.6	**58.1**	<u>2.6</u>	75.0	70.5	29.5	**52.4**	48.6

Table 11: Prompt ablation on **PMV (Full)**. All variants share the same trained checkpoint; only the inference-time prompt template changes. The Train-aligned variant uses the format used during training; the others apply a single modification (CoT prefix, JSON formatting, R1-style reasoning tags, no-reasoning constraint, or spatial prompting). All metrics are scaled to $[0, 100]$. Best results are in bold and second-best are underlined.

R1 degradation is an extraction artifact. Performance drop under R1-style wrapping is primarily due to evaluation pipeline mismatch (the permissive `extract_json` routine described above), not model capability.

Capability-specific prompt preference. Different prompt formats favor different capabilities, suggesting potential for capability-conditional prompting.

Summary. Prompt format and adapter composition act on largely orthogonal axes: the former selects which capability is emphasized, while the latter determines what the model can achieve.

D.7 Training Dynamics

Setting. We log per-step SFT cross-entropy loss and per-step GRPO reward across the entire training pipeline of all three adapters. Figure 7 shows all six curves arranged as a 3×2 grid: rows correspond to the three adapters (Δ_{P}, Δ_{M}, Δ_{V}), columns to the SFT and GRPO phases. SFT panels report autoregressive cross-entropy loss; RL panels report the per-adapter capability-aligned GRPO reward (Appendix C.1) (range $[0, 1]$, higher is better). The three colors – teal (■), amber (■), and crimson (■) – index the three adapters consistently across all figures and tables in this paper.

Δ_{P}: extended training over heterogeneous capabilities. Perceptual Grounding requires the model to learn six output structures (object sets, bounding boxes, integer counts, depth) from scratch, so we train it for substantially more steps than the other two adapters. SFT loss declines monotonically, with a long tail near zero indicating the model has saturated on the supervised data. The GRPO reward stabilizes within the $[0.4, 0.7]$ band, consistent with the heterogeneous reward mix in Δ_{P} (set F1, IoU, exact-match counting): this is an aggregate reward trace, so task-level evaluation is needed to assess how performance is distributed across outputs.

Δ_{M}: SFT convergence and checkpoint selection. Text-conditioned Memory exhibits clean SFT convergence: loss decreases from 1.3 to roughly 0.25 over 400 steps. The RL phase operates in a high-context, multi-capability regime (long history strings combined with structured outputs spanning enumeration, bounding boxes, and dictionary-keyed memory queries), and the GRPO reward fluctuates within an early-training band rather than monotonically improving. We therefore use the standard practice of *checkpoint selection on a held-out split*: the checkpoint with the best validation M-score (epoch 2) is retained as Δ_{M}, producing the M-score of 61.4 reported in Sec. 4.3. Stabilizing GRPO further in this regime — e.g., through KL-anchored exploration or task-stratified group normalization — is a promising direction for future work but is orthogonal to the conclusions of this paper, which are based on the selected checkpoint.

Δ_{V}: rapid convergence via dependency-ordered design. Visual Spatio-Temporal Memory shows the cleanest training dynamics of the three adapters. SFT loss drops from ~ 5 to near zero within ~ 50 steps, and GRPO reward saturates near 0.95 throughout the entire 200-step RL phase. These curves are consistent with useful initialization from earlier stages. They do not separately identify the effect of stage order or establish end-to-end compute savings over independently trained alternatives. The training controls in Sec. 4.1 assess the complete framework's task performance.

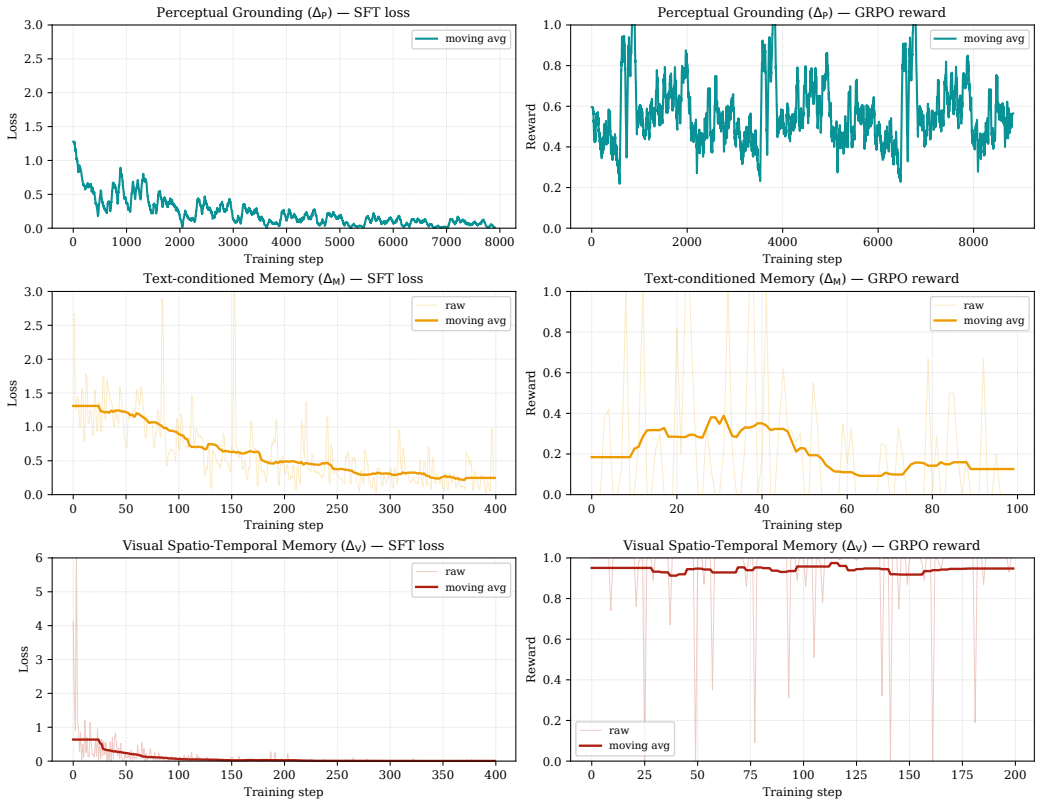

Figure 7: Training dynamics across the three adapters of STaR-VL. **Top row:** Perceptual Grounding (Δ_{P}, teal) — SFT loss (left) decreases from $\sim$1.3 to near 0 over $\sim$8K steps; the GRPO reward (right) stabilizes around 0.55, reflecting the heterogeneous task mix (set, bbox, count, depth) on which Δ_{P} is trained. **Middle row:** Text-conditioned Memory (Δ_{M}, amber) — SFT loss converges from 1.3 to $\sim$0.25 over 400 steps; the GRPO reward fluctuates within an early-training band, and we select the highest-performing checkpoint (epoch 2) for downstream use. **Bottom row:** Visual Spatio-Temporal Memory (Δ_{V}, crimson) — SFT loss drops rapidly from $\sim$5 to $\sim$0.1 within the first 50 steps; GRPO reward stays at $\sim$0.95 throughout, reflecting that the merged $\Theta_0 \oplus \Delta_{\mathrm{P}} \oplus \Delta_{\mathrm{M}}$ initialization already provides a strong starting point that GRPO then transfers to multi-frame visual inputs.

E Additional Curriculum and Training Analysis

Curriculum rationale. The P→M→V sequence first establishes perceptual grounding, then learns temporal-memory behavior on symbolic histories, and finally adapts the initialized system to visual inputs. Symbolic histories offer explicit object and receptacle state, allowing the memory module to learn compression and retrieval before visual-state extraction is added. The matched joint-training comparison supports the complete staged framework. A factorial comparison holding memory architecture, training budget, and objective variants fixed while changing only stage order would be needed to establish the necessity of this particular order.

Training dynamics. The visual stage inherits the merged P/M initialization. Its SFT loss drops from approximately 5 to 0.1 within about 50 optimizer steps and its RL reward remains near 0.95 over the reported 200-step phase (Appendix D.7). These curves describe optimization under the chosen stage objectives. They are consistent with useful initialization but do not by themselves isolate why transfer occurs or compare total compute efficiency against alternative training systems.

Interpreting M-score and V-score. The two regimes share task definitions but differ in available state, coverage, and input representation. For STaR-VL they also compare the memory configuration against the full visual configuration. The 13.8-point aggregate difference is therefore a diagnostic

comparison. The coverage sweep and memory-system controls quantify specific contributors without treating M-score as a formal performance upper bound.

F Additional Training, Memory, and Input Controls

This section reports complementary controls for the comparisons in Sec. 4.1. The same-backbone joint-LoRA and ReMEmbR results are given in Table 3.

F.1 Persistent Memory and Oracle Scene State

An adapted ReMEmbR procedure supplies self-generated persistent memory under the SpaMEM protocol. On InternVL3-8B, we keep the episodes, prompts, history budget, output parsers, and aggregation procedure fixed across conditions. Self-generated memory and ground-truth scene state are distinct: the latter is privileged information and is used as a diagnostic reference.

InternVL3-8B input condition	**V-score**
Raw visual frames	29.40
ReMEmbR-adapted, memory only	39.80
ReMEmbR-adapted, memory + frames	43.60
Oracle symbolic memory only	45.70
Oracle symbolic memory + frames	50.51

Table 12: Self-generated and oracle memory on a common backbone. Oracle inputs contain ground-truth scene state and are not competing practical systems.

Memory plus frames improves over raw frames by 14.20 points. Replacing self-generated memory with oracle state adds another 6.91 points. These contrasts support the importance of accurate, accessible scene state; their different backbone prevents a fully matched ranking against STaR-VL-3B.

F.2 Structured Prompting and Output Serialization

The structured prompt requests frame inspection, object/location lists, adjacent-step comparison, an explicit temporal memory table, and an answer based on that table.

Model	**Reference prompt**	**Memory-table prompt**	**Δ**
Qwen2.5-VL-3B	21.85	22.21	+0.36
STaR-VL-3B	47.60	48.30	+0.70

Table 13: Structured prompting on the untrained backbone and trained model.

Across the six prompt wrappers evaluated on the untrained backbone, V-scores range from 20.92 to 22.60. The best is 0.75 above the reference prompt and 25.00 below STaR-VL with the reference prompt. These results concern the tested prompting procedures rather than all possible inference strategies.

For object recall, we also require different output schemas while holding the task fixed. Each parser is checked by rendering ground truth in that schema and obtaining a round-trip score of 1.000.

Output schema	SOR-M	Change
Trained JSON list	98.50	—
Dictionary	91.20	−7.30
Plain lines	88.60	−9.90
Wrapped object	93.80	−4.70

Table 14: Object recall under alternative validated output serializations.

Performance is format-sensitive but remains high across these alternatives. This supports robustness beyond one exact serialization while leaving broader task specialization as a separate question.

F.3 History Coverage

The visual-budget sweep uses the STaR-VL-3B checkpoint and holds the symbolic-history score fixed.

Frames	M-score	V-score	M−V
3	61.40	47.60	13.80
5	61.40	49.50	11.90
8	61.40	51.20	10.20
16	61.40	52.40	9.00

Table 15: Coverage sensitivity with full symbolic history retained throughout.

The increase from three to sixteen frames recovers 4.80 points, or 34.78% of the initial visual–symbolic difference. Because symbolic history is not reduced to the same observed steps, this is a coverage analysis rather than an information-matched modality intervention. A separate one-seed Qwen2.5-VL-3B symbolic-history rerun scores 44.78 with full history and 43.38 with five steps, a 1.40-point difference. These rerun scores are a separate protocol comparison from the symbolic benchmark row in Table 1.

F.4 Sensitivity to Explicit Temporal Tags

We evaluate the same checkpoint with correct, shuffled, and removed tags.

Tags	SOR-M	RSR-M	SED+	SED−	V-score
Correct	98.50	54.30	74.00	66.20	47.60
Shuffled	97.00	30.50	28.30	22.40	34.40
Removed	96.20	25.80	24.60	18.70	32.30

Table 16: Test-time temporal-tag perturbations after training with correct tags.

Object recall changes little, whereas relation and event metrics decline substantially. The model therefore relies on temporal metadata for order-sensitive behavior. This test changes both available temporal information and the train–test input distribution; it does not establish how a model trained and evaluated without textual tags would perform. The task studied here is memory and reasoning over explicitly ordered observations.