EGOP^{2}C: INTERNALIZING PLAN-PERCEIVE-COMPOSE FOR SPATIAL REASONING IN EGOCENTRIC VIDEO

Anonymous authors
Paper under double-blind review

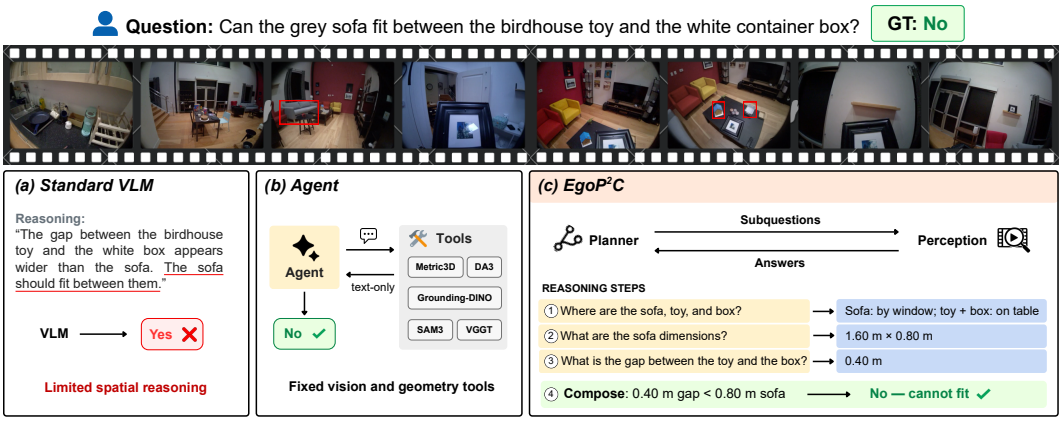


Figure 1: **EgoP^{2}C internalizes planning, perception, and composition within a single VLM for egocentric spatial reasoning.** (a) A standard VLM reasons directly from the video and incorrectly concludes that the sofa fits between the objects. (b) An agentic system queries external vision and geometry tools to obtain the required measurements. (c) In EgoP^{2}C, the Planner generates targeted questions and decides when enough evidence has been gathered, Perception answers those questions from the video, and the Planner composes the final answer.

ABSTRACT

Answering complex spatial reasoning questions from egocentric video requires determining what evidence is needed, acquiring targeted observations, and combining them through multiple reasoning steps. Existing vision-language models (VLMs) struggle to perform this process reliably. Agentic systems address this challenge by querying external vision and geometry tools, but training their reasoning leaves these tools unchanged, so inaccurate observations can still lead to incorrect answers. We introduce EgoP^{2}C, a framework that internalizes a plan-perceive-compose process within a single VLM. To this end, we first train specialized Planner and Perception teachers using separate adapters (Dual-LoRA), then fine-tune the corresponding experts in a mixture-of-experts (MoE) student under their supervision. We subsequently train the student on trajectories generated by its current policy using a segment-level extension of multi-teacher on-policy distillation (MOPD), assigning different trajectory segments to different supervisors. The Planner teacher supervises which evidence-seeking questions to ask, when to stop gathering evidence, and how to compose the final answer. The Perception teacher supervises the video-grounded answers to those questions. At inference, the student executes this adaptive workflow end-to-end, without external spatial tools. Across three spatial reasoning benchmarks, EgoP^{2}C students built from Qwen3-VL-8B and SenseNova-SI-1.3-8B outperform their base models by an average of 9.2 and 7.5 percentage points, respectively. Across both backbones, EgoP^{2}C reduces inference FLOPs by 65.8-66.3% and achieves 1.29-1.49$\times$ speedups over their shared Dual-LoRA teacher.

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.

1 INTRODUCTION

Spatial reasoning in egocentric (first-person) video requires understanding object locations, distances, and spatial constraints (Li et al., 2026a; Horton et al., 2026). This capability can support wearable assistants (Majumdar et al., 2024; Huang et al., 2025), visual assistance for blind and low vision users (Xiao et al., 2025; Liu et al., 2026), robot navigation and manipulation (Cheng et al., 2025; Driess et al., 2023), and augmented reality (Yuan et al., 2025). However, camera motion, occlusion, and a limited field of view cause relevant objects to appear only briefly or partially, scattering evidence across frames (Yang et al., 2025a; Horton et al., 2026). To answer complex spatial questions, a model must determine which spatial facts are needed, extract the corresponding observations from this fragmented evidence, and combine them through multiple reasoning steps. For example, deciding whether the sofa fits between the two objects in Figure 1 requires locating the objects, estimating the sofa dimensions and available gap, and comparing these measurements. An incorrect spatial estimate can invalidate the answer even when the subsequent reasoning is sound.

Prior work improves spatial understanding by training vision-language models (VLMs) on spatial questions (Chen et al., 2024) or providing estimated 3D object coordinates as textual input (Gholami et al., 2026). These models reason over supplied video or coordinates, but are not explicitly trained to recognize when those inputs are insufficient and seek further evidence. Agentic systems query external vision and geometry tools for evidence at inference (Chen et al., 2026; Zhang et al., 2026b; Dai et al., 2026), but the tools remain fixed: improving the reasoning component does not improve their measurements.

We introduce EgoP^{2}C (Egocentric Plan-Perceive-Compose), a framework that implements these three stages through two specialist roles: *Planner* and *Perception*. Given a video and a spatial question, the Planner either selects direct answering or generates subquestions for Perception and uses its answers to compose the final response. Neither path uses external spatial tools. Because training both roles with shared parameters can lead to conflicting updates, we first train separate Planner and Perception teachers using low-rank adaptation (LoRA) modules on a shared frozen VLM backbone (Hu et al., 2022). In this Dual-LoRA configuration, each adapter is trained through supervised fine-tuning (SFT) followed by reinforcement learning using Group Relative Policy Optimization (GRPO) (Shao et al., 2024).

Executing the Dual-LoRA teachers requires adapter switching and repeated context processing, so we integrate their capabilities into a single student while retaining separate parameters for planning and perception. A mixture-of-experts (MoE) architecture provides dedicated role experts within a shared backbone. We construct this student through sparse upcycling (Komatsuzaki et al., 2023), which copies selected feedforward networks of a dense VLM into experts for planning and composition, perception, and direct answering. We initialize the Planner and Perception experts with supervision from their corresponding teachers. The student executes the interaction within one checkpoint and reuses a common video prefix across roles. To train the initialized student on its own trajectories, we extend multi-teacher on-policy distillation (MOPD) (Ma et al., 2026) by assigning specialist teachers to corresponding segments within each trajectory. The Planner teacher supervises planning and composition, and the Perception teacher supervises observations. The frozen base VLM supervises direct answers.

We evaluate EgoP^{2}C initialized from Qwen3-VL-8B and SenseNova-SI-1.3-8B on three egocentric spatial reasoning benchmarks. Both students outperform their respective base models and reduce inference FLOPs and latency relative to the shared Dual-LoRA teacher system.

In summary, our contributions are as follows:

- We introduce EgoP^{2}C, a framework that internalizes a plan-perceive-compose process within a single VLM. The resulting model answers complex egocentric spatial reasoning questions by generating and answering subquestions without external tools.
- We develop segment-level MOPD, assigning the Planner teacher to planning and composition segments, and the Perception teacher to observation segments within each student rollout.
- On three spatial reasoning benchmarks, the Qwen3-VL-8B and SenseNova-SI-1.3-8B students consistently outperform their respective base models, with average gains of 9.2 and 7.5 percentage points, respectively. Both students also reduce inference FLOPs by 65.8-66.3% and achieve a 1.29-$1.49\times$ speedup over the shared Dual-LoRA teachers.

2 Related Work

Egocentric spatial reasoning requires inferring object relations and scene geometry from partial observations captured by a moving camera (Yang et al., 2025a; Li et al., 2026a; Horton et al., 2026). Existing methods train VLMs on object relations (Chen et al., 2024), distance estimation (Cai et al., 2026b), depth understanding (Cai et al., 2025), and camera pose estimation (Cai et al., 2026a). Others represent geometry explicitly through textual maps of estimated object coordinates (Gholami et al., 2026) or descriptions of camera trajectories and scene structure (Hua et al., 2026). Spatial agents obtain evidence through visual programs (Marsili et al., 2025; Cho et al., 2026; Luo et al., 2026), vision and geometry tools (Chen et al., 2026; Zhang et al., 2026b; Dai et al., 2026), or world simulation (Zhu et al., 2026). Recent methods also transfer spatial knowledge through synthetic viewpoint changes (Zhang et al., 2026a), depth and camera geometry supervision (Tang et al., 2026), cognitive-map distillation (Li et al., 2026b), and supervision from tool interactions (Zhou et al., 2026). EgoP^{2}C focuses on integrating separately trained planning and perception capabilities into one VLM, with specialist teachers supervising the model's own trajectories.

Mixture-of-Experts Models. Sparse MoE models increase parameter capacity while limiting computation by activating a subset of experts for each token (Fedus et al., 2022; Jiang et al., 2024; Dai et al., 2024; Muennighoff et al., 2025). Sparse Upcycling (Komatsuzaki et al., 2023) constructs an MoE from a pretrained dense checkpoint by copying its feedforward layers into experts. Branch-Train-MiX (Sukhbaatar et al., 2024) and FlexOlmo (Shi et al., 2025) combine experts trained on different data domains, allowing specialization before integration. BAR (Morrison et al., 2026) extends modular expert training to post-training, combining domain experts through parameter merging and lightweight router training. In multimodal settings, MoSE (Xu et al., 2025) organizes experts around skills such as perception, prediction, and planning, using skill supervision to train the routers. To integrate separately trained capabilities, EgoP^{2}C initializes role-specific experts through supervision from specialist teachers. Each segment activates the expert assigned to its role.

On-Policy Distillation. Recent distillation methods improve language and vision-language models using teacher feedback on student-generated sequences (Gu et al., 2024; Ko et al., 2024; 2025; Yuan et al., 2026). In on-policy distillation (OPD), this feedback is computed on sequences sampled from the current student (Agarwal et al., 2024). MOPD (Ma et al., 2026) extends this approach to multiple specialist teachers, selecting one teacher for each student trajectory according to the task domain. Other methods adapt teacher supervision at the token or step level (Wang et al., 2026a; Xu et al., 2026). ViGOS (Wang et al., 2026b) supervises visual descriptions and subsequent reasoning with the same frozen model under different contexts. H-OPD (Yin et al., 2026) combines vision-language and text-only teachers through confidence-weighted token-level supervision. EgoP^{2}C instead assigns Planner and Perception teachers to their corresponding segments in the trajectories generated by the student, using their token-level feedback for segment-level MOPD.

3 Method

Given an input $x{=}(v, q)$ consisting of an egocentric video v and a spatial question q, the task is to predict an answer a. Under our protocol, the question is answered either directly or through a plan-perceive-compose process.

The Planner first selects a mode $b \in \{\texttt{DIRECT}, \texttt{PLAN}\}$. With `DIRECT`, the model answers the original question without requesting observations. With `PLAN`, the Planner requests spatial observations over one or more rounds before composing the answer. At round j, it generates a request u_j containing one or more self-contained spatial subquestions. Perception answers each from the video, producing a list of textual observations o_j. The Planner uses the accumulated requests and observations to either request further subquestions or compose the final answer. Perception receives only the video and its current subquestion. Let J denote the number of planning rounds and c the final composition, including the answer a. The output is $y{=}a$ for `DIRECT` and $y{=}(u_1, o_1, \ldots, u_J, o_J, c)$ for `PLAN`. Appendix B.3 provides the full execution details and probabilistic formulation.

Figure 2(a-c) summarizes three training stages. We train Planner and Perception teachers in a *Dual-LoRA* configuration (Section 3.1). These teachers provide supervision for initializing the corresponding roles in a MoE student (Section 3.2). We then apply segment-level MOPD, where each teacher supervises its corresponding segments of student-generated trajectories (Section 3.3). At

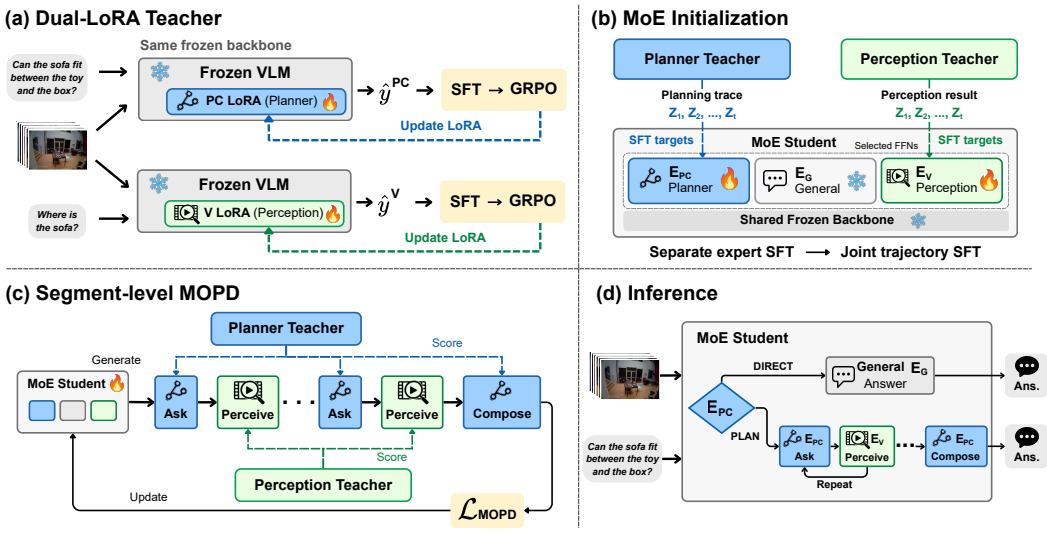


Figure 2: **Overview of EgoP^{2}C.** The framework integrates planning, perception, and composition into a single VLM. (a) Planner (PC) and Perception (V) teachers share a frozen backbone and use separate LoRA adapters trained with SFT followed by GRPO. (b) Teacher-generated sequences initialize corresponding MoE experts through separate role training followed by joint SFT. z_t denotes the t-th token. (c) Segment-level MOPD trains the student on its own interactions, with PC supervising subquestions and composition and V supervising observations. (d) At inference, the Planner expert either answers directly using the General expert (`DIRECT`) or iteratively requests subquestion answers from Perception before composing the final answer (`PLAN`).

inference, the student's Planner expert selects the General expert for direct answering or iteratively asks the Perception expert to answer subquestions before composing the final answer (Figure 2(d)).

3.1 Specialist Teachers

The Dual-LoRA teachers share a pretrained VLM backbone, π_{B} (Figure 2(a)). Its vision encoder, visual-language aligner, and base language parameters remain frozen. Separate LoRA adapters in the language model (Hu et al., 2022) define the Planner policy, π_{PC}, for both planning and composition, and the Perception policy, π_{V}. Direct answers use π_{B} with both adapters disabled.

Planner SFT covers the initial `DIRECT`/`PLAN` choice. For `PLAN` examples, we also supervise subquestion generation and final answer composition. Perception SFT covers answers to those subquestions. For `DIRECT` examples, we supervise only the Planner's mode-selection token, excluding the model's answer from the SFT loss. For data construction, a language model generates subquestions and answer composition without access to the correct final answer. Because an annotation for the original question does not necessarily provide answers to its subquestions, we obtain supervision for each subquestion from ground-truth answers when available or a strong perception model's weak labels otherwise. The final answer is revealed only afterward to filter incorrect trajectories. Section 4.1 summarizes the data sources, generation models, and sample counts.

After SFT, we apply GRPO in two stages, first to Perception and then to the Planner. Without ground-truth subquestion answers, Perception GRPO uses the change in the frozen Planner's margin for the correct final option as a weak reward. Both this setting and Planner GRPO use original multiple-choice questions with known correct options.

Perception GRPO. For each video and subquestion, the current Perception policy samples a group of answers. A verifier scores each against the ground-truth subquestion answer when available. Otherwise, we use `unknown` as a placeholder for a missing subquestion answer. Utility is the change in the frozen SFT-trained Planner's margin for the correct final option when this placeholder is replaced by the sampled answer. The video and other observations remain fixed. This utility reward does not independently verify the subquestion answer. The Planner evaluates composition without route selection or further questions.

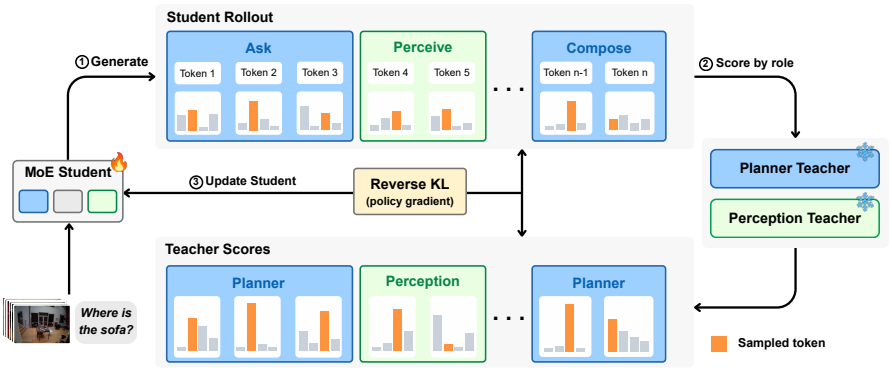


Figure 3: **Overview of the segment-level MOPD pipeline.** Each training iteration (i) selects a training sample, (ii) generates a rollout with the MoE student model, (iii) assigns each segment to the corresponding frozen teacher, (iv) obtains teacher log probabilities for the sampled tokens, and (v) updates the student with a reverse-KL policy gradient. The example follows the `PLAN` path. Orange bars indicate the probabilities of the same sampled token under the student (top) and its assigned teacher (bottom).

Planner GRPO. We freeze the trained Perception teacher and optimize the Planner starting from its initial route choice. The frozen backbone answers under `DIRECT`, while the Planner queries Perception and composes the answer under `PLAN`. Final correctness rewards the initial choice in both branches and the Planner's subsequent outputs in the interaction branch. Auxiliary terms measure improvement over direct answering by π_{B} and penalize invalid outputs. Base and Perception answer tokens are excluded from Planner updates. Appendix B.1 defines the rewards and training masks.

These teachers specialize in distinct roles, but their separate execution requires adapter switching and repeated context processing. We next transfer their capabilities to one student.

3.2 MoE Initialization

We construct the student from a pretrained dense VLM, π_0, by copying the feedforward networks in selected decoder layers into three role-specific experts (Figure 2(b)). Each role $r \in \{\mathrm{PC}, \mathrm{V}, \mathrm{G}\}$ has an expert E_r for planning and composition, perception, or direct answering, respectively. These roles execute the direct or plan-perceive-compose workflow in Figure 2(d). PC and V also use role-specific attention residuals, while the shared backbone and E_{G} remain frozen. The common video prefix uses the frozen base path and shares its KV cache across roles (Appendix B.3).

We first train the PC and V experts and attention residuals separately. Planner teacher sequences supervise PC targets, while Perception teacher sequences supervise V targets. Available ground truth supervises subquestion answers and final option labels. Let $\mathcal{L}^r_{\mathrm{seq}}$ and $\mathcal{L}^r_{\mathrm{gt}}$ denote the corresponding token cross-entropy losses. We minimize $\mathcal{L}_{\mathrm{init}} = \sum_{r\in\{\mathrm{PC},\mathrm{V}\}} w_r(\lambda_{\mathrm{seq}}\mathcal{L}^r_{\mathrm{seq}} + \lambda_{\mathrm{gt}}\mathcal{L}^r_{\mathrm{gt}})$, where w_r balances roles and the λ coefficients weight supervision sources. Joint SFT then applies the same objective to subquestion samples and complete trajectories from both routes. For `DIRECT`, only PC's mode choice is supervised, excluding G's answer tokens. Appendices A.2 and B.2 detail sample reuse, trainable parameters, and masks.

3.3 Segment-Level MOPD

We apply MOPD (Ma et al., 2026) to student trajectories spanning the initial mode choice through the final answer. The Planner teacher supervises the mode choice, subquestions, and composition segments. The Perception teacher supervises observation segments, and the frozen teacher backbone π_{B} supervises direct answer segments. At this stage, we introduce an attention residual for G and train it with the PC and V parameters. The shared backbone and E_{G} remain frozen. Let $T(r)$ denote the teacher assigned to role r. Figure 3 illustrates this process for a `PLAN` trajectory.

Let π^r_θ denote the student policy for role r. At sampling time, the student parameters $\bar{\theta}$ induce the trajectory distribution $p_{\bar{\theta}}(\cdot \mid x)$ under the interaction protocol defined above. We sample $Y \sim p_{\bar{\theta}}(\cdot \mid x)$, where Y contains all generated tokens. The frozen teachers score these tokens using the same role-specific information available during trajectory generation and student updates. For the t-th

generated token Y_t, with role r_t and context $C_t^{r_t}$, the clipped policy gradient advantage is

$$\widetilde{A}_t = \mathrm{clip}\left(\mathrm{sg}\left[\log \frac{T(r_t)(Y_t \mid C_t^{r_t})}{\pi_{\bar{\theta}}^{r_t}(Y_t \mid C_t^{r_t})}\right], -A_{\max}, A_{\max}\right), \tag{1}$$

where sg stops gradients and $A_{\max}$ bounds the advantage. For segment $s \in \mathcal{S}(Y)$, let r_s be its role and $\mathcal{I}_s$ its target positions. Let $N_r(Y)$ count all supervised tokens of role r. We minimize

$$\mathcal{L}_{\mathrm{MOPD}}(\theta) = -\mathbb{E}_{x,\, Y \sim p_{\bar{\theta}}}\left[\sum_{s \in \mathcal{S}(Y)} \frac{\omega_{r_s}}{N_{r_s}(Y)} \sum_{t \in \mathcal{I}_s} \widetilde{A}_t \log \pi_\theta^{r_s}(Y_t \mid C_t^{r_s})\right]. \tag{2}$$

The objective uses a clipped local reverse-KL surrogate, with ω_r balancing roles and $N_r(Y)$ normalizing their supervised token counts. The student executes both routes within one checkpoint. Each role selects its corresponding FFN expert and attention residual while reusing the common video KV cache. Appendix B.2 details scoring, and Appendix B.3 describes execution and cache reuse.

4 Experiments

We evaluate EgoP^{2}C on egocentric spatial reasoning benchmarks and analyze inference efficiency, teacher specialization, and the effects of MoE initialization and segment-level MOPD.

4.1 Experimental setup

Implementation. The Planner and Perception teachers share the SenseNova-SI-1.3-Qwen3-VL-8B backbone (Cai et al., 2026b), with separate rank-16 LoRA modules in the language model. Students use Qwen3-VL-8B (Bai et al., 2025) or SenseNova-SI-1.3-Qwen3-VL-8B as the backbone, abbreviated as Qwen3-VL and SenseNova-SI. Both students use 12 MoE layers and rank-16 attention residuals. Student SFT runs for three epochs per stage with a learning rate of 5×10^{-5} and batch size 16. MOPD runs for 600 updates with a learning rate of 3×10^{-5} and 16 trajectories per update. We train on eight NVIDIA H100 GPUs. Appendix B provides additional details on model parameterization, training objectives, and configurations.

Training Data. We construct training samples from Aria Digital Twin (ADT) (Pan et al., 2023), Ego-Exo4D (Grauman et al., 2024), VSI-590K (Yang et al., 2025b), SPAR-7M (Zhang et al., 2025), and Ego4D (Grauman et al., 2022). The videos used for training are disjoint from those in evaluation benchmarks. For teacher SFT, the DeepSeek-V4-Pro API (DeepSeek-AI et al., 2026) generates subquestions and composes the final answer. Subquestion answers come from ground-truth annotations when available; otherwise, they come from the Qwen3.5-397B API (Qwen Team, 2026). The Perception and Planner teachers use $35,811$ and $17,817$ samples for SFT, and $3,160$ and $3,872$ samples for GRPO, respectively. Student initialization draws $15,000$ Perception and $4,100$ Planner samples from these pools, followed by joint SFT on 8,000 samples. MOPD uses 8,000 prompts. Appendices A and D provide construction details, sample distributions, and prompt templates.

Benchmarks. We evaluate EgoP^{2}C on three egocentric spatial reasoning benchmarks. VSI-Bench (Yang et al., 2025a) covers spatial relations, distance estimation, and route planning in indoor scenes. EgoProx (Li et al., 2026a) tests egocentric navigation and object interactions using spatial relations and action context. We use its single-answer multiple-choice tasks and exclude Chain of Actions, which requires structured action-sequence prediction. R3D-Bench (Horton et al., 2026) tests spatial reasoning from wearable video, including comparing and combining object distance and dimension measurements. We do not provide models with R3D-Bench's depth maps or camera poses. Agentic baselines may estimate them from RGB inputs using external tools.

Evaluation Settings. We sample videos at 1 fps with a maximum of 32 frames per input, and cap each model generation at 2,048 tokens. Dual-LoRA and EgoP^{2}C may ask at most six Perception subquestions per question. The same sampled frames are used within each controlled comparison.

4.2 Results

Table 1 shows that both students significantly improve over their base models on all three benchmarks. EgoP^{2}C $_{\textbf{Qwen3-VL}}$'s average score across the benchmarks exceeds that of the larger Qwen3-

Table 1: **Performance on egocentric spatial reasoning benchmarks.** Subscripts indicate the backbone or agent model. Green parentheses report gains in percentage points over the baseline row. Student gains on all three benchmarks have 95% bootstrap confidence intervals excluding zero (Appendix C.1).

#	Method	VSI-Bench	EgoProx	R3D-Bench	Avg.
	Agentic Systems (w/ tools)				
1	SpatialClaw $_{\text{Gemma4-31B}}$ (Cho et al., 2026)	48.5	47.9	40.7	45.7
2	Think3D $_{\text{Gemini 2.5 Pro}}$ (Zhang et al., 2026b)	62.0	42.1	43.4	49.2
3	S-Agent $_{\text{GPT-5.4}}$ (Dai et al., 2026)	66.7	49.8	45.9	54.1
	Open-source General Models (w/o tools)				
4	Qwen3-VL-8B (Bai et al., 2025)	57.9	31.0	35.2	41.4
5	Qwen3.5-9B (Qwen Team, 2026)	54.3	32.8	37.3	41.5
6	InternVL3.5-8B (Wang et al., 2025)	56.1	29.3	33.6	39.7
7	Qwen3-VL-30B-A3B (Bai et al., 2025)	63.2	38.5	39.0	46.9
	Open-source Spatial Models (w/o tools)				
8	VST-7B-SFT (Yang et al., 2026)	60.6	29.2	31.4	40.4
9	Cambrian-S-7B (Yang et al., 2025b)	62.9	27.1	36.3	42.1
10	SenseNova-SI-1.3-8B (Cai et al., 2026b)	64.8	34.2	37.1	45.4
11	RynnBrain-8B (Dang et al., 2026)	68.6	39.6	35.9	48.0
	Proprietary Models (w/o tools)				
12	Gemini 2.5 Pro (Comanici et al., 2025)	53.5	40.3	39.5	44.4
13	Gemini 3 Pro (Pichai et al., 2025)	53.7	42.9	44.3	47.0
14	GPT-5.5 (OpenAI, 2026)	60.4	44.2	44.7	49.8
	Ours (w/o tools)				
15	Dual-LoRA $_{\text{SenseNova-SI}}$	67.6	49.0	44.8	53.8
16	**EgoP^{2}C $_{\text{Qwen3-VL}}$** (△ vs. #4)	**62.1** (+4.2)	**45.9** (+14.9)	**43.7** (+8.5)	**50.6** (+9.2)
17	**EgoP^{2}C $_{\text{SenseNova-SI}}$** (△ vs. #10)	**66.9** (+2.1)	**47.2** (+13.0)	**44.5** (+7.4)	**52.9** (+7.5)

Table 2: **Inference cost of Dual-LoRA and EgoP^{2}C.** Subscripts identify the starting checkpoints. Green parentheses report relative reductions from Dual-LoRA. FLOPs, latency, and generated tokens are averaged over all questions across the three benchmarks. Generated tokens include outputs from all roles.

Method	TFLOPs	Latency (s)	Generated tokens	Latency/token (ms)
Dual-LoRA $_{\text{SenseNova-SI}}$	165.7	9.4	353.1	26.6
EgoP^{2}C $_{\text{Qwen3-VL}}$	**55.9** (-66.3%)	**7.3** (-22.3%)	**315.4** (-10.7%)	**23.1** (-13.2%)
EgoP^{2}C $_{\text{SenseNova-SI}}$	**56.6** (-65.8%)	**6.3** (-33.0%)	**276.8** (-21.6%)	**22.8** (-14.3%)

VL-30B-A3B (50.6 vs. 46.9). The gains for SenseNova-SI show that EgoP^{2}C can improve a model already trained for spatial understanding. EgoP^{2}C $_{\text{SenseNova-SI}}$ scores 52.9, exceeding GPT-5.5 (49.8), the strongest proprietary baseline, and close to S-Agent (54.1), the strongest agentic baseline in our comparison. It preserves much of the Dual-LoRA teachers' performance while executing all roles within a single checkpoint and reusing a shared video prefix.

The subtask results in Appendix C.2 also show that both students outperform their respective base models on every evaluated subtask. On VSI-Bench, EgoP^{2}C $_{\text{Qwen3-VL}}$ and EgoP^{2}C $_{\text{SenseNova-SI}}$ gain 7.2 and 4.4 points on *Object Absolute Distance*, which requires estimating the distance between two objects. On R3D-Bench, they gain 12.7 and 11.8 points on *Total Walk Distance*, which requires summing the horizontal distances from the camera wearer to one object and from that object to another. On EgoProx, they gain 29.3 and 24.1 points on *Exploration Relative*, which asks for the direction of the next navigation step toward a goal. These results show that EgoP^{2}C improves basic spatial perception and achieves larger gains on tasks that combine multiple spatial observations.

To test whether additional QA training alone accounts for these gains, we train a SenseNova-SI baseline with SFT + GRPO on the same original questions and subquestion QA data as the Dual-LoRA teacher. For SFT, the baseline is trained only on answers and receives no supervision on intermediate planning and interaction sequences. GRPO uses both the Perception training samples and the original multiple-choice questions used for Planner GRPO. At inference, the baseline answers directly, without generating subquestions. Appendix C.3 shows statistically significant gains for both Dual-LoRA and EgoP^{2}C $_{\text{SenseNova-SI}}$ over this baseline. EgoP^{2}C $_{\text{SenseNova-SI}}$ improves by 1.4, 6.5, and

Table 3: **Further supervised training of the student.** All systems use the SenseNova-SI-1.3-8B backbone. Average is the unweighted mean across the three benchmarks.

Method	VSI-Bench	EgoProx	R3D-Bench	Avg.
Dual-LoRA teachers	**67.6**	49.0	44.8	53.8
EgoP^{2}C	66.9	47.2	44.5	52.9
EgoP^{2}C + additional SFT	67.5	**49.8**	**46.4**	**54.6**

Table 4: **Effect of joint and separate role training.** All variants use SenseNova-SI-1.3-8B.

Method	Training	VSI-Bench	EgoProx	R3D-Bench
Base	–	64.8	34.2	37.1
Joint LoRA	SFT	62.1	37.9	37.7
Perception-only LoRA	SFT	65.4	39.0	39.2
Planner-only LoRA	SFT	61.8	43.9	41.0
Dual-LoRA	SFT	65.8	46.7	43.2
Joint LoRA	SFT + GRPO	62.6	36.3	37.3
Dual-LoRA	SFT + GRPO	**67.6**	**49.0**	**44.8**

4.4 points on VSI-Bench, EgoProx, and R3D-Bench, respectively. These results support the value of plan-perceive-compose beyond directly learning answers from the same QA data.

Inference Efficiency. We compare both EgoP^{2}C students with their shared SenseNova-SI Dual-LoRA teachers across all three benchmarks. All models use vLLM (Kwon et al., 2023) with the same hardware, video sampling, resolution, batch size, and generation limits. Each system follows its own routing and stopping decisions. Measurements cover the complete interaction, including visual encoding and all generation steps. Table 2 shows that EgoP^{2}C $_{\textbf{Qwen3-VL}}$ and EgoP^{2}C $_{\textbf{SenseNova-SI}}$ reduce FLOPs by 66.3% and 65.8% and achieve $1.29\times$ and $1.49\times$ speedups over Dual-LoRA, respectively. Both students generate fewer tokens and have lower end-to-end latency per generated token, despite making slightly more Perception calls (2.6 and 2.7 vs. 2.5 per question). Peak GPU memory increases from 19.4 GB for Dual-LoRA to 25.2 and 25.3 GB for the two students. The efficiency gains also persist when comparing subsets in which each student and Dual-LoRA generate the same number of subquestions for each original question (Appendix C.4).

4.3 Ablation Studies

Further Supervised Training of the Student. To assess improvement after MOPD, we apply additional SFT to EgoP^{2}C $_{\textbf{SenseNova-SI}}$ using 3,000 samples in total, drawn from the Planner and Perception SFT pools. Although these samples were used to train the teachers, they were not used for training the student in its preceding training stages. We update the Planner and Perception experts without retraining the teachers or repeating MOPD. Table 3 shows that additional SFT improves the student's scores on all three benchmarks, raising its average from 52.9 to 54.6, above the Dual-LoRA teachers' 53.8. It also surpasses the teachers on EgoProx and R3D-Bench. These results show that the student benefits from further direct supervision using samples from the teachers' SFT pools.

Joint versus Separate Training of Planning and Perception. We compare joint and separate role training using SenseNova-SI. Joint LoRA uses a single rank-32 adapter for both roles, matching Dual-LoRA's total number of trainable adapter parameters. We also evaluate single-role adapters trained with SFT on the corresponding role's data. Perception-only uses its trained adapter to answer the original question without planning, while Planner-only uses the frozen base model for perception. Table 4 shows that Joint SFT improves over the base model on EgoProx and R3D-Bench but reduces VSI-Bench performance. Perception-only SFT improves over the base model on all three benchmarks, while Planner-only SFT outperforms Joint SFT on EgoProx and R3D-Bench. After SFT, Dual-LoRA outperforms Joint LoRA and both single-role variants on all three benchmarks. These results show that each type of supervision is useful on its own, but Joint LoRA does not consistently outperform the single-role variants. GRPO further improves Dual-LoRA on all three benchmarks, and Joint LoRA improves slightly on VSI-Bench but declines on EgoProx and R3D-Bench. After SFT and GRPO, Dual-LoRA outperforms Joint LoRA by 5.0, 12.7, and 7.5

Table 5: **Effects of student architecture and training stages.** Role initialization includes separate role SFT followed by joint SFT. Dense + SFT + MOPD matches the SFT data used for role initialization.

Configuration	VSI-Bench	EgoProx	R3D-Bench
Qwen3-VL-8B	57.9	31.0	35.2
Dense + MOPD	58.4	36.3	32.8
Dense + SFT + MOPD	59.8	38.6	35.1
MoE + MOPD	60.4	40.4	39.6
MoE + role initialization	60.8	42.7	41.0
MoE + role initialization + MOPD	**62.1**	**45.9**	**43.7**
SenseNova-SI-1.3-8B	64.8	34.2	37.1
Dense + MOPD	64.7	39.5	39.7
Dense + SFT + MOPD	65.2	43.0	41.9
MoE + MOPD	65.3	44.1	42.4
MoE + role initialization	66.0	45.2	43.3
MoE + role initialization + MOPD	**66.9**	**47.2**	**44.5**

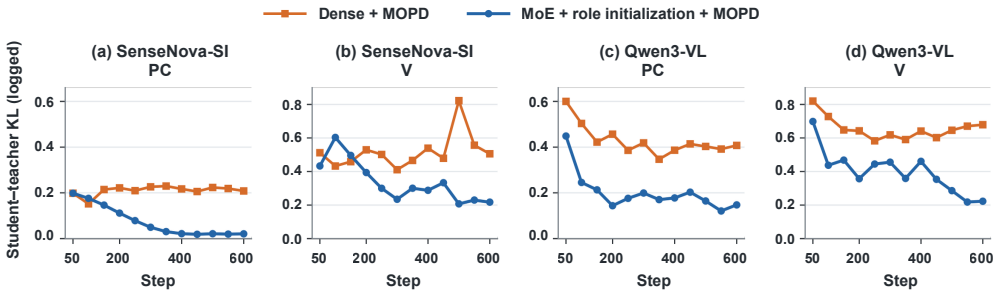


Figure 4: **Student-teacher KL during MOPD.** PC denotes planning and composition, and V denotes perception. The initialized MoE finishes with lower KL for both roles on both backbones.

points on VSI-Bench, EgoProx, and R3D-Bench, respectively. These results support using separate parameters for planning and perception while combining their capabilities during interaction.

Effects of MoE Architecture, Role Initialization, and MOPD. Table 5 compares dense and MoE students across training stages. The dense SFT baseline learns the same role targets using shared parameters. MoE + MOPD omits initialization SFT, while MoE + role initialization is evaluated before MOPD. Adding matched SFT before MOPD improves the dense baseline on both backbones. However, Dense + SFT + MOPD remains below MoE + role initialization + MOPD on all three benchmarks. MoE + MOPD also outperforms Dense + MOPD on both backbones and all three benchmarks. Within the MoE architecture, role initialization further improves MOPD performance on all three benchmarks, with subsequent MOPD providing additional gains. On EgoProx, initialization raises final MOPD scores by 5.5 and 3.1 points for Qwen3-VL and SenseNova-SI, respectively, followed by additional MOPD gains of 3.2 and 2.0 points.

Alongside its performance gains, the full model shows closer agreement with both teachers, with lower final KL than Dense + MOPD for PC and V on both backbones (Figure 4). For SenseNova-SI, PC KL finishes at 0.021, compared with 0.208 for Dense + MOPD. Supplementary gradient analysis in Appendix C.5 also finds opposing gradient directions between the two role objectives in the dense Qwen3-VL student. Together, these results support combining role initialization with subsequent MOPD in the MoE student.

5 Conclusion

We presented EgoP^{2}C, a framework that learns planning and composition separately from perception and integrates these capabilities into one VLM for egocentric spatial reasoning. Dual-LoRA teachers guide the initialization of dedicated MoE experts. Segment-level MOPD then assigns each specialist to the corresponding segments of the student's own interactions, supervising both the observations it produces and the reasoning that uses them. Experiments show consistent improvements over both general and spatially trained backbones. Future work will explore adaptive frame selection for longer videos and training objectives that balance answer accuracy with inference cost.

AI USE STATEMENT

We used LLMs as an auxiliary tool to polish the writing of this manuscript and improve its grammar, clarity, and readability.

ETHICS STATEMENT

The training and evaluation data are derived from existing public datasets, and we collect no new data related to humans. Some source datasets contain egocentric recordings of people and their surroundings and may therefore reflect privacy concerns or biases present in the original data. Model-generated subquestion answers are used only as weak supervision when independent annotations are unavailable and may inherit errors or biases from the generating models.

REPRODUCIBILITY STATEMENT

Section 3 describes the interaction protocol and training method. Section 4.1 reports the model backbones, training settings, hardware, benchmarks, and evaluation budgets. Appendix A details the data sources, sample composition, demonstration generation and filtering procedures, and video-level separation between training and evaluation. Appendix B specifies the reward functions, trainable parameters, supervision masks, and inference procedure. Appendix D provides the data-generation and inference prompts.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

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

Wenxiao Cai, Iaroslav Ponomarenko, Jianhao Yuan, Xiaoqi Li, Wankou Yang, Hao Dong, and Bo Zhao. SpatialBot: Precise spatial understanding with vision language models. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2025.

Zhipeng Cai, Zhuang Liu, Yunyang Xiong, Zechun Liu, Vikas Chandra, and Yangyang Shi. VLM3: Vision language models are native 3D learners. *arXiv preprint arXiv:2605.30561*, 2026a.

Zhongang Cai, Ruisi Wang, Chenyang Gu, Fanyi Pu, Junxiang Xu, Yubo Wang, Wanqi Yin, Zhitao Yang, Chen Wei, Qingping Sun, Tongxi Zhou, Jiaqi Li, Hui En Pang, Oscar Qian, Yukun Wei, Zhiqian Lin, Xuanke Shi, Kewang Deng, Xiaoyang Han, Zukai Chen, Xiangyu Fan, Hanming Deng, Lewei Lu, Liang Pan, Bo Li, Ziwei Liu, Quan Wang, Dahua Lin, and Lei Yang. Scaling spatial intelligence with multimodal foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026b.

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. SpatialVLM: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Siyi Chen, Mikaela Angelina Uy, Chan Hee Song, Faisal Ladhak, Adithyavairavan Murali, Qing Qu, Stan Birchfield, Valts Blukis, and Jonathan Tremblay. SpaceTools: Tool-augmented spatial reasoning via double interactive RL. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

An-Chieh Cheng, Yandong Ji, Zhaojing Yang, Zaitian Gongye, Xueyan Zou, Jan Kautz, Erdem Biyik, Hongxu Yin, Sifei Liu, and Xiaolong Wang. NaVILA: Legged robot vision-language-action model for navigation. In *Proceedings of Robotics: Science and Systems (RSS)*, 2025.

Seokju Cho, Ryo Hachiuma, Abhishek Badki, Hang Su, Byung-Kwan Lee, Chan Hee Song, Sifei Liu, Subhashree Radhakrishnan, Seungryong Kim, Yu-Chiang Frank Wang, and Min-Hung Chen. SpatialClaw: Rethinking action interface for agentic spatial reasoning. *arXiv preprint arXiv:2606.13673*, 2026.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Damai Dai, Chengqi Deng, Chenggang Zhao, R. X. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y. K. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. DeepSeekMoE: Towards ultimate expert specialization in mixture-of-experts language models. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

Yalun Dai, Hao Li, Shulin Tian, Runmao Yao, Yuhao Dong, Fangzhou Hong, Zhaoxi Chen, Fangfu Liu, Tao Wang, Kim-Hui Yap, and Ziwei Liu. S-Agent: Spatial tool-use elicits reasoning for spatial intelligence. *arXiv preprint arXiv:2606.20515*, 2026.

Ronghao Dang, Jiayan Guo, Bohan Hou, Sicong Leng, Kehan Li, Xin Li, Jiangpin Liu, Yunxuan Mao, Zhikai Wang, Yuqian Yuan, et al. RynnBrain: Open embodied foundation models. *arXiv preprint arXiv:2602.14979*, 2026.

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. DeepSeek-V4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026.

Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. PaLM-E: An embodied multimodal language model. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2023.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research (JMLR)*, 2022.

Mohsen Gholami, Ahmad Rezaei, Weimin Zhou, Yong Zhang, and Mohammad Akbari. Spatial reasoning with vision-language models in ego-centric multi-view scenes. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2026.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4D: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2022.

Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-Exo4D: Understanding skilled human activity from first- and third-person perspectives. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

Maxwell Horton, Wei Lu, Quan Tran, Yury Astashonok, Kirmani Ahmed, Babak Damavandi, Anuj Kumar, Xiao Zhang, and Seungwhan Moon. R3D: Quantitative 3D spatial reasoning for egocentric wearables. *arXiv preprint arXiv:2607.02921*, 2026.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.

Jiacheng Hua, Yishu Yin, Yuhang Wu, Tai Wang, Yifei Huang, and Miao Liu. Unleashing spatial reasoning in multimodal large language models via textual representation guided reasoning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, 2026.

Yifei Huang, Jilan Xu, Baoqi Pei, Lijin Yang, Mingfang Zhang, Yuping He, Guo Chen, Xinyuan Chen, Yaohui Wang, Zheng Nie, Jinyao Liu, Dechen Lin, Fang Fang, Kunpeng Li, Chang Yuan, Yu Qiao, Yali Wang, and Limin Wang. Vinci: A real-time smart assistant based on egocentric vision-language model for portable devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT)*, 2025.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. DistiLLM: Towards streamlined distillation for large language models. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2024.

Jongwoo Ko, Tianyi Chen, Sungnyun Kim, Tianyu Ding, Luming Liang, Ilya Zharkov, and Se-Young Yun. DistiLLM-2: A contrastive approach boosts the distillation of LLMs. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2025.

Aran Komatsuzaki, Joan Puigcerver, James Lee-Thorp, Carlos Riquelme Ruiz, Basil Mustafa, Joshua Ainslie, Yi Tay, Mostafa Dehghani, and Neil Houlsby. Sparse upcycling: Training mixture-of-experts from dense checkpoints. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023.

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

Jinzhao Li, Yinuo Chen, Dongxu Piao, Panwang Pan, Yifan Yu, Dong Wang, Honglei Yan, Liang Yue, Shaofei Wang, Yixin Chen, Siyuan Huang, and Miao Liu. EgoProx: Evaluating MLLMs on egocentric 3D proximity reasoning across a cognitive hierarchy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026a.

Xudong Li, Mengdan Zhang, Peixian Chen, Jiaxi Tan, Zihao Huang, Jingyuan Zheng, Yan Zhang, Xiawu Zheng, Xing Sun, and Rongrong Ji. OmniView-Space: Reinforcing spatial reasoning via multi-perspective spatial mapping. *arXiv preprint arXiv:2607.00881*, 2026b.

Ruiping Liu, Jiaming Zhang, Angela Schön, Karin Müller, Junwei Zheng, Kailun Yang, Anhong Guo, Kathrin Gerling, and Rainer Stiefelhagen. ObjectFinder: An open-vocabulary assistive system for interactive object search by people who are blind. *International Journal of Human-Computer Interaction (IJHCI)*, 2026.

Zhanpeng Luo, Ce Zhang, Silong Yong, Cunxi Dai, Qianwei Wang, Haoxi Ran, Guanya Shi, Katia Sycara, and Yaqi Xie. pySpatial: Generating 3D visual programs for zero-shot spatial reasoning. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2026.

Wenhan Ma, Jianyu Wei, Liang Zhao, Hailin Zhang, Bangjun Xiao, Lei Li, Qibin Yang, Bofei Gao, Yudong Wang, Rang Li, Jinhao Dong, Zhifang Sui, and Fuli Luo. MOPD: Multi-teacher on-policy distillation for capability integration in LLM post-training. *arXiv preprint arXiv:2606.30406*, 2026.

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 Berges, Shiqi Zhang, Pulkit Agrawal, Yonatan Bisk, Dhruv Batra, Mrinal Kalakrishnan, Franziska Meier, Chris Paxton, Sasha Sax, 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 (CVPR)*, 2024.

Damiano Marsili, Rohun Agrawal, Yisong Yue, and Georgia Gkioxari. Visual agentic AI for spatial reasoning with a dynamic API. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Jacob Morrison, Sanjay Adhikesaven, Akshita Bhagia, Matei Zaharia, Noah A. Smith, and Sewon Min. Train separately, merge together: Modular post-training with mixture-of-experts. *arXiv preprint arXiv:2604.18473*, 2026.

Niklas Muennighoff, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Jacob Morrison, Sewon Min, Weijia Shi, Pete Walsh, Oyvind Tafjord, Nathan Lambert, Yuling Gu, Shane Arora, Akshita Bhagia, Dustin Schwenk, David Wadden, Alexander Wettig, Binyuan Hui, Tim Dettmers, Douwe Kiela, Ali Farhadi, Noah A. Smith, Pang Wei Koh, Amanpreet Singh, and Hannaneh Hajishirzi. OLMoE: Open mixture-of-experts language models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025.

OpenAI. GPT-5.5 system card. Technical report, OpenAI, 2026. URL `https://deploymentsafety.openai.com/gpt-5-5/gpt-5-5.pdf`.

Xiaqing Pan, Nicholas Charron, Yongqian Yang, Scott Peters, Thomas Whelan, Chen Kong, Omkar Parkhi, Richard Newcombe, and Yuheng (Carl) Ren. Aria Digital Twin: A new benchmark dataset for egocentric 3D machine perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Sundar Pichai, Demis Hassabis, and Koray Kavukcuoglu. A new era of intelligence with Gemini 3. Google Blog, 2025. URL `https://blog.google/products-and-platforms/products/gemini/gemini-3/`.

Qwen Team. Qwen3.5: Towards native multimodal agents, 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

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.

Weijia Shi, Akshita Bhagia, Kevin Farhat, Niklas Muennighoff, Jacob Morrison, Pete Walsh, Dustin Schwenk, Shayne Longpre, Jake Poznanski, Allyson Ettinger, Daogao Liu, Margaret Li, Dirk Groeneveld, Mike Lewis, Wen-tau Yih, Luca Soldaini, Kyle Lo, Noah A. Smith, Luke Zettlemoyer, Pang Wei Koh, Hannaneh Hajishirzi, Ali Farhadi, and Sewon Min. FlexOLMo: Open language models for flexible data use. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Sainbayar Sukhbaatar, Olga Golovneva, Vasu Sharma, Hu Xu, Xi Victoria Lin, Baptiste Rozière, Jacob Kahn, Daniel Li, Wen-tau Yih, Jason Weston, and Xian Li. Branch-Train-MiX: Mixing expert LLMs into a mixture-of-experts LLM. In *Proceedings of the Conference on Language Modeling (COLM)*, 2024.

Jiayu Tang, Yuchen Zhou, and Chao Gou. SpatialSV: Internalizing interpretable 3D spatial awareness in MLLMs via task-oriented visual supervision. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2026.

Jianze Wang, Ying Liu, Jinlong Chen, Xuchun Hu, Qilong Zhang, Yu Cao, Jun Wang, Hua Yang, Yong Xie, and Qianglong Chen. MAD-OPD: Breaking the ceiling in on-policy distillation via multi-agent debate. *arXiv preprint arXiv:2605.01347*, 2026a.

Sihan Wang, Xiyao Liu, Lianqing Liu, and Zhi Han. Seeing before reasoning: Decoupling perception and reasoning for shortcut-resilient multimodal on-policy self-distillation. *arXiv preprint arXiv:2606.19120*, 2026b.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long 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*, 2025.

Junbin Xiao, Nanxin Huang, Hao Qiu, Zhulin Tao, Xun Yang, Richang Hong, Meng Wang, and Angela Yao. EgoBlind: Towards egocentric visual assistance for the blind. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Lu Xu, Jiaqian Yu, Xiongfeng Peng, Yiwei Chen, Weiming Li, Jaewook Yoo, Sunghyun Chunag, Dongwook Lee, Daehyun Ji, and Chao Zhang. MoSE: Skill-by-skill mixture-of-experts learning for embodied autonomous machines. *arXiv preprint arXiv:2507.07818*, 2025.

Yuanda Xu, Hejian Sang, Zhengze Zhou, Ran He, Zhipeng Wang, and Alborz Geramifard. TIP: Token importance in on-policy distillation. *arXiv preprint arXiv:2604.14084*, 2026.

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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025a.

Rui Yang, Ziyu Zhu, Yanwei Li, Jingjia Huang, Shen Yan, Siyuan Zhou, Zhe Liu, Xiangtai Li, Shuangye Li, Wenqian Wang, Yi Lin, and Hengshuang Zhao. Visual spatial tuning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2026.

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

Qixiang Yin, Huanjin Yao, Yuchen Cai, Jianghao Chen, Ziyi Wang, Min Yang, Fei Su, and Zhicheng Zhao. H-OPD: Confidence aware heterogeneous multi-teacher multimodal on-policy distillation. *arXiv preprint arXiv:2607.02592*, 2026.

Qianhao Yuan, Jie Lou, Xing Yu, Hongyu Lin, Le Sun, Xianpei Han, and Yaojie Lu. Vision-OPD: Learning to see fine details for multimodal LLMs via on-policy self-distillation. *arXiv preprint arXiv:2605.18740*, 2026.

Wentao Yuan, Jiafei Duan, Valts Blukis, Wilbert Pumacay, Ranjay Krishna, Adithyavairavan Murali, Arsalan Mousavian, and Dieter Fox. RoboPoint: A vision-language model for spatial affordance prediction in robotics. In *Proceedings of the Conference on Robot Learning (CoRL)*, 2025.

Jiahui Zhang, Yurui Chen, Yanpeng Zhou, Yueming Xu, Ze Huang, Jilin Mei, Junhui Chen, Yu-Jie Yuan, Xinyue Cai, Guowei Huang, Xingyue Quan, Hang Xu, and Li Zhang. From flatland to space: Teaching vision-language models to perceive and reason in 3D. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Wanyue Zhang, Wenxiang Wu, Wang Xu, Jiaxin Luo, Helu Zhi, Yibin Huang, Shuo Ren, Zitao Liu, and Jiajun Zhang. World2VLM: Distilling world model imagination into VLMs for dynamic spatial reasoning. *arXiv preprint arXiv:2604.26934*, 2026a.

Zaibin Zhang, Yuhan Wu, Lianjie Jia, Yifan Wang, Zhongbo Zhang, Yijiang Li, Binghao Ran, Fuxi Zhang, Zhuohan Sun, Yizhuang Peng, Zhenfei Yin, Lijun Wang, and Huchuan Lu. Think3D: Thinking with space for spatial reasoning. *arXiv preprint arXiv:2601.13029*, 2026b.

Yang Zhou, Zixuan Huang, Sunzhu Li, Zhuo Yang, Chen Zhang, Shunian Chen, Caijun Yan, Jianyao Xu, Shunyu Liu, Weijie Fu, Peiliang Li, Xiaozhi Chen, and Yuxiang Cai. SpatialCLI: Learning to reason with spatial tools, then without them. *arXiv preprint arXiv:2607.27703*, 2026.

Chenming Zhu, Jingli Lin, Yilin Long, Peizhou Cao, Tai Wang, Jiangmiao Pang, and Xihui Liu. Thinking with imagination: Agentic visual spatial reasoning with world simulators. *arXiv preprint arXiv:2606.06476*, 2026.

A Training Data Generation

Section 4.1 summarizes the data sources, generation models, and sample counts. Here, we describe how annotations and generated trajectories form role supervision, followed by the detailed data composition.

A.1 Data Construction

ADT (Pan et al., 2023) provides camera and object poses together with geometry and identity annotations, from which we derive answers to spatial subquestions about distance, direction, size, location, and identity. Ego-Exo4D (Grauman et al., 2024), VSI-590K (Yang et al., 2025b), and SPAR-7M (Zhang et al., 2025) supply original questions and final answers, including multiple-choice questions. Ego-Exo4D also provides annotated subjective questions, and Ego4D (Grauman et al., 2022) contributes additional Planner samples. Annotations for the original question support trajectory verification but do not independently verify every intermediate observation.

We construct demonstrations for both `DIRECT` and `PLAN`. For `DIRECT`, the frozen teacher backbone answers the original video and question with no LoRA active. For `PLAN`, the planning model generates subquestions and receives observations before deciding whether to request further evidence or compose the final answer. Available subquestion annotations are inserted as answers to the corresponding subquestions. When an observation must be generated, the visual model receives only the video and subquestion, without the original answer options or other planning history. Generated observations without independent annotations serve as weak supervision.

We compare each generated final answer with the task annotation and allow up to three retries for an incorrect `PLAN` trajectory. If the resulting `PLAN` answer is correct, we retain it as a `PLAN` example regardless of the frozen base's answer. If `PLAN` remains incorrect but the frozen base answers correctly, we retain a `DIRECT` example. We discard examples for which both routes are incorrect. From retained `PLAN` trajectories, we extract subquestion-answer pairs for Perception and planning or composition targets with their preceding histories for the Planner. Appendix D describes the generation prompts and their conversion into training conversations.

A.2 Data Composition

Table 6: Source composition of the teacher training sets. All values count samples, and dashes indicate no samples from that source.

	SFT		GRPO	
Source	Perception	Planner	Perception	Planner
ADT	27,279	6,926	1,085	1,900
Ego-Exo4D	1,167	2,600	1,004	778
VSI-590K	5,502	3,777	535	896
SPAR-7M	1,863	1,792	536	298
Ego4D	–	2,722	–	–
Total	35,811	17,817	3,160	3,872

Table 7: Question types in the Perception SFT set. Counts sum to 35,811 samples.

Type	Samples	Type	Samples
Bearing	12,283	Count	1,330
Distance	11,339	Other	1,820
Camera distance	2,732	Identity	711
Size	2,609	Egomotion bearing	580
Object location	2,255	Egomotion distance	152

Table 6 reports the source composition of the teacher training sets, and Table 7 summarizes the Perception SFT question types. Perception GRPO includes 2,089 samples with objective subques-

tion annotations from ADT and Ego-Exo4D, and 1,071 without such annotations from SPAR-7M and VSI-590K. The latter have multiple-choice original questions with annotated correct options, which support the observation utility rewards defined in Appendix B.1. Planner GRPO also uses multiple-choice questions with annotated correct options.

Student initialization samples are selected through correctness filtering. The Planner subset comprises 2,566 planning samples and 1,534 composition samples. These samples cover both routes, with initial choices included among the planning targets. Joint SFT combines complete interaction samples from both routes with reused subquestion samples. The MOPD mixture contains 6,500 spatial questions and 1,500 general (non-spatial) questions. The latter are randomly sampled from the teacher training sets and cover tasks such as action recognition. A source question can yield several subquestion samples and can be reused across stages, so the reported counts do not represent disjoint collections of questions. We verified that our training data contain none of the source recordings used for benchmark evaluation.

B Additional Implementation Details

This section specifies the teacher rewards, student parameterization and training objectives, and inference procedure.

B.1 Teacher Training

Training order. Section 4.1 specifies the teacher backbone and adapter configuration. Both roles first undergo SFT. We supervise the Planner's initial `DIRECT`/`PLAN` choice and, for `PLAN`, its subquestions and final composition. The frozen base's direct answer is excluded from Planner supervision. During Perception GRPO, the current Perception policy samples a group of answers to each subquestion. Objective annotations provide verifier rewards where available. For answers without a ground-truth subquestion answer, the frozen SFT-trained Planner, denoted by $\bar{\pi}_{\mathrm{PC}}$, provides utility scores during training as detailed below. Only the Perception adapter is updated in this stage. After training Perception, we freeze the resulting policy and optimize the Planner through the complete interaction in Equation 13. Each rollout begins with the current Planner's mode choice. A `DIRECT` rollout obtains its answer from the frozen base. A `PLAN` rollout continues with Planner subquestions, frozen Perception observations, and Planner composition. A sampled group for one original question can contain both modes. Each role uses its own SFT policy as the GRPO KL reference. For `DIRECT`, the Planner loss and KL masks cover only its initial choice. For `PLAN`, they also cover its subquestions and composition. Frozen base and Perception outputs are excluded. The final frozen policies π_{PC} and π_{V} subsequently supervise student training.

Objective perception rewards. For an observation o with an independent subquestion annotation z, the reward is

$$R_{\mathrm{obj}}(o, z) = \mathbf{1}[\mathrm{valid}(o)]\, V_{\kappa}(o, z), \tag{3}$$

where validity requires parsing as the requested answer type κ. All verifier scores lie in $[0, 1]$. Size questions are included in SFT but not in objective Perception GRPO. We use the following rules:

- **Bearing.** Index the eight directions around a circle. For predicted and annotated indices i_o, i_z, let $d{=}\min(|i_o - i_z|, 8 - |i_o - i_z|)$. The scores for $d{=}0, 1, 2$, and $d \geq 3$ are 1.00, 0.60, 0.18, and 0, respectively.
- **Distance.** For a nonzero annotated distance, define $e{=}|o - z|/|z|$. The mean relative accuracy is $V_{\mathrm{dist}}(o, z){=}\frac{1}{11}\sum_{k=0}^{10} \mathbf{1}[e \leq 1 - c_k]$, where $c_k{=}0.50 + 0.045k$.
- **Identity.** Compute precision and recall from normalized token overlap and take their F1 score. To discourage listing multiple candidate identities, apply an additional length penalty by multiplying this score by $|z|/(|z| + \mathrm{extra})$, where $\mathrm{extra} = \max(0, |o| - |z|)$ and $|o|, |z|$ count normalized predicted and reference tokens, respectively.
- **Count and presence.** Counts receive $\exp(-|o - z|)$. Presence receives $\mathbf{1}[o{=}z]$ after parsing.
- **Location.** Average the bearing and distance scores for the corresponding components.

Answer margins. Both the observation utility and Planner reward use the margin of the correct final option. For answer options $\mathcal{A}$, annotated option $a^\star$, policy π, and context C, define

$$M_\pi(C) = \ell_\pi(a^\star \mid C) - \log \sum_{a \in \mathcal{A} \setminus \{a^\star\}} \exp\big(\ell_\pi(a \mid C)\big), \tag{4}$$

where $\ell_\pi(a \mid C)$ is the log probability of the option label. The options in $\mathcal{A}$ are the candidate answers to the original question. The correct-option annotation is used for scoring and is withheld from generation prompts. Perception answers the intermediate subquestions with numbers or short text. At inference, the model can also answer questions without predefined options. Throughout this appendix, $\mathrm{clip}(z, a, b) = \min\{b, \max\{a, z\}\}$.

Observation utilities without ground-truth subquestion answer. The original question has an annotated final answer even when its subquestion has no ground-truth. We use this final answer to evaluate each answer sampled by the current Perception policy. Perception receives only the video and subquestion. It does not receive the original question's answer options or annotated answer.

For a sampled answer o, let $C[o]$ contain the video, original question and options, and the observations used for composition, with o filling the slot for the current subquestion. The comparison context $C[\texttt{unknown}]$ replaces only that answer with `unknown`. We force the frozen SFT-trained Planner to compose on these two contexts and score the final answer options using Equation 4. The video and all other evidence are identical, and the Planner neither selects a mode nor asks further subquestions during this evaluation. The utility of the sampled answer is

$$\begin{aligned} \delta(o) &= M_{\bar{\pi}_{\mathrm{PC}}}(C[o]) - M_{\bar{\pi}_{\mathrm{PC}}}(C[\texttt{unknown}]), \\ u_+(o) &= \mathrm{clip}\left(\frac{\delta(o) - \epsilon}{s_{\mathrm{V}}}, 0, 1\right), \\ u_-(o) &= \mathrm{clip}\left(\frac{-\delta(o) - \epsilon}{s_{\mathrm{V}}}, 0, 1\right). \end{aligned} \tag{5}$$

Positive $\delta(o)$ means that providing the answer increases the correct option's margin, while negative $\delta(o)$ means that it reduces the margin. The dead zone ϵ excludes small changes, and s_{V} sets their scale. Every newly sampled answer is scored directly during training. The `unknown` margin is common to answers in the same group and can be reused while the full input context and frozen Planner remain unchanged. The subjective reward is

$$R_{\mathrm{subj}}(o) = \gamma n \begin{cases} \alpha_+ u_+(o) - \alpha_- u_-(o) - \beta^{\mathrm{V}}_{\mathrm{leak}} I_{\mathrm{leak}}(o), & \mathrm{valid}(o), \\ -\beta^{\mathrm{V}}_{\mathrm{invalid}}, & \text{otherwise.} \end{cases} \tag{6}$$

For each initially correct prompt, we remove the reference observation and recompose the final answer with the frozen Planner. We set the offline necessity weight n_{pre} to 1.0 if this removal changes the correct answer and to 0.1 otherwise. When no offline necessity value is available, the data builder uses a default value of 0.25. For subquestions supervised by teacher-generated references rather than ground-truth answers, we multiply this weight by 0.5 before GRPO. The resulting value is stored with each training sample and used as n in Equation 6. Thus, n is 0.5, 0.05, or 0.125 for necessary, redundant, or default teacher-referenced samples, respectively. If a training sample lacks a stored necessity weight, the reward computation uses $n = 0.25$ as a fallback. I_{leak} detects explicit selection of a multiple-choice option label. The outer factor γn also scales both penalties. Rewards within each sampled group are used to compute GRPO advantages and update the Perception adapter. Subsequent groups are sampled from the updated Perception policy, while the Planner remains frozen throughout this stage. This reward provides weak supervision for an observation's usefulness during composition under fixed evidence. GRPO regularizes the Perception policy toward its SFT checkpoint, and the weak reward penalizes invalid outputs and explicit option-label answers. Objective verifiers provide supervision for subquestion accuracy where independent annotations are available.

Planner reward. We score the complete path selected by the current Planner's initial action b. Let $r_{\mathrm{out}} = \mathbf{1}[\hat{a} = a^\star]$ denote final correctness. The direct margin M_{direct} uses the frozen teacher backbone π_{B} with no LoRA active on the original video and question, including any answer options, with no intermediate observations or Planner-private text. For $b = \texttt{PLAN}$, the selected-path margin M_{route}

Table 8: Reward and training coefficients. n_{pre} denotes the necessity weight before reference downweighting.

Component	Parameter	Value
LoRA alpha (teachers / student residuals)	α	32
GRPO KL coefficient (Planner / Perception)	β_{KL}	0.04
No-ground-truth reward scale	γ	0.5
Necessity (necessary / redundant / default)	n_{pre}	1.0, 0.1, 0.25
Teacher-reference downweight	m_{ref}	0.5
Missing-weight fallback	n_{fallback}	0.25
Positive / negative utility	α_+, α_-	0.60, 0.30
Perception leakage penalty	$\beta^{\text{V}}_{\text{leak}}$	0.50
Perception invalid-answer penalty	$\beta^{\text{V}}_{\text{invalid}}$	1
Utility dead zone	ϵ	0.10
Utility scale	s_{V}	2
Planner margin scale	s_{PC}	2
Planner gain / regression	$\lambda_{\text{gain}}, \lambda_{\text{reg}}$	0.0875, 0.10
Invalid interaction structure	$\beta^{\text{PC}}_{\text{parse}}$	0.25
Unresolved reference	$\beta^{\text{PC}}_{\text{ref}}$	0.10
Planner interface leakage	$\beta^{\text{PC}}_{\text{leak}}$	0.30
Sequence / ground-truth supervision	$\lambda_{\text{seq}}, \lambda_{\text{gt}}$	1, 1

uses the current Planner's composition with observations from the frozen Perception teacher. For b=`DIRECT`, it uses the same frozen base and identical context as M_{direct}, so $M_{\text{route}} = M_{\text{direct}}$. For d=$(M_{\text{route}} - M_{\text{direct}})/s_{\text{PC}}$, the reward is

$$\begin{aligned} R_{\text{PC}} = r_{\text{out}} + \lambda_{\text{gain}}(1 - r_{\text{out}})\,\text{clip}(d, 0, 1) \\ - \lambda_{\text{reg}}\,\text{clip}(-d, 0, 1) - P_{\text{protocol}}. \end{aligned} \tag{7}$$

Correctness supplies the principal signal. Positive margin gain provides additional feedback when the final answer is incorrect. Regression below the direct base policy is penalized regardless of correctness. For `PLAN`, the gain includes both specialization and evidence collection because the direct reference is a different policy. For `DIRECT`, the margin difference is zero. The protocol penalty is

$$P_{\text{protocol}} = \beta^{\text{PC}}_{\text{parse}} I_{\text{parse}} + \beta^{\text{PC}}_{\text{ref}} I_{\text{ref}} + \beta^{\text{PC}}_{\text{leak}} I_{\text{leak}}. \tag{8}$$

The indicators identify an invalid mode or interaction structure, unresolved references in a subquestion, and disclosure of answer options through the perception interface. Validity follows the selected mode's protocol: a `DIRECT` path is valid without subquestions, and the reference and interface penalties apply only when those components occur. These penalties apply regardless of the final answer. Table 8 summarizes reward and training coefficients.

B.2 Student Training

Experts and trainable parameters. For both student backbones in Section 4.1, we replace the feedforward networks in the last 12 of the 36 decoder layers with role experts. These are layers 24-35 under zero-based indexing. In each selected layer ℓ, the original feedforward network $F_0^{(\ell)}$ is copied into three experts, with one active expert per token outside the common prefix:

$$\text{FFN}^{(\ell)}(h_t, r_t) = E^{(\ell)}_{r_t}(h_t), \qquad E^{(\ell)}_r\big|_{\text{init}} = F_0^{(\ell)}, \quad r \in \{\text{G}, \text{PC}, \text{V}\}. \tag{9}$$

The segment role determines r_t. In the same layers, rank-16 role attention residuals modify `q_proj`, `k_proj`, `v_proj`, and `o_proj`. Outside the common prefix, each such projection uses $W_r^{(\ell)}$=$W_0^{(\ell)} + B_r^{(\ell)} A_r^{(\ell)}$, with the scaling absorbed into $B_r^{(\ell)}$. The PC and V residuals use $B_r^{(\ell)}$=0

at initialization. These copies and zero residuals preserve the base computation for the same input context at initialization. We freeze the vision encoder, aligner, and original language parameters. During initialization, the PC and V experts and their attention residuals are trainable. The general feedforward expert remains frozen throughout training. At the start of MOPD, we add a general attention residual for `DIRECT` answers, using the same rank, projections, and layers as PC and V. MOPD trains this residual together with the PC and V parameters. All role attention residuals are disabled on the common video prefix, which always uses the frozen base path.

Initialization supervision. For role r, let $\mathcal{D}^r_{\mathrm{seq}}$ contain teacher-generated targets and $\mathcal{D}^r_{\mathrm{gt}}$ contain available ground-truth targets. The components of the initialization objective are

$$\mathcal{L}^r_k = -\mathbb{E}_{(C,z,\mathcal{I})\sim\mathcal{D}^r_k}\left[\frac{1}{|\mathcal{I}|}\sum_{t\in\mathcal{I}}\log \pi^r_\theta(z_t \mid C, z_{<t})\right], \quad k \in \{\mathrm{seq}, \mathrm{gt}\}. \tag{10}$$

The sequence targets are generated greedily by the corresponding teacher and inserted as fixed targets into the student's template. PC sequence targets cover the initial mode choice in both routes and subquestions and composition under `PLAN`, while V sequence targets cover subquestion answers. `DIRECT` answer tokens belong to G and are excluded from initialization supervision. Ground-truth targets supervise annotated subquestion answers and final option labels on the corresponding V and PC segments. The two supervision sources use separate samples and target masks, so their losses do not count the same supervised positions twice. We set $w_{\mathrm{PC}}{=}w_{\mathrm{V}}{=}1$, with each role loss averaged over its supervised token positions as in Equation 10. The roles first update their own parameter sets separately. We then continue joint SFT on subquestions and complete interaction samples from both modes, applying the same objective to each PC or V role's target positions. The joint SFT sampler balances sample counts between PC and V. G is not trained during initialization or joint SFT. The general samples used for MOPD are part of its prompt mixture, as reported in Appendix A.2.

Teacher assignment and scoring. During MOPD, the student samples one trajectory per prompt with temperature 1.0 and top-p 1.0. Each trajectory is generated in full before segment-wise backpropagation. We accumulate gradients over all segments from 16 sampled trajectories before each optimizer update. For a student-generated segment with role r, the teacher is

$$T(r) = \begin{cases} \pi_{\mathrm{PC}}, & r = \mathrm{PC}, \\ \pi_{\mathrm{V}}, & r = \mathrm{V}, \\ \pi_{\mathrm{B}}, & r = \mathrm{G}. \end{cases} \tag{11}$$

The initial mode choice, planning, and composition share the PC teacher, including supervision of the final answer in `PLAN`. A `DIRECT` trajectory has a PC mode-choice segment followed by a G answer segment, supervised by π_{PC} and π_{B}, respectively. For both student backbones, π_{B} is the frozen SenseNova-SI teacher with both LoRA adapters disabled. It supervises the General expert's direct answers. Each new rollout is scored live by its assigned teachers. For each sampled token, its teacher log probability is obtained from its logit minus the log-sum-exp over the complete vocabulary. Scoring uses the student's sampled tokens and preceding context, not a teacher-generated continuation. The student log probability in Equation 1 corresponds to the student parameters $\bar{\theta}$ used to generate the trajectory. The advantage is detached and clipped to $[-5, 5]$, with $A_{\max}{=}5$, before weighting the current student's log probability. We set $\omega_{\mathrm{PC}}{=}\omega_{\mathrm{V}}{=}\omega_{\mathrm{G}}{=}1$. Equation 2 normalizes by the total supervised token count for each role across the trajectory. Thus, a role's contribution does not grow simply because it generates more or longer segments.

Local reverse-KL interpretation. For a fixed role r and context C, let $q_\theta(z) = \pi^r_\theta(z \mid C)$, $\tau(z) = T(r)(z \mid C)$, and $A(z) = \log[\tau(z)/q_{\bar{\theta}}(z)]$. Using $\mathbb{E}_{z\sim q_\theta}[\nabla_\theta \log q_\theta(z)] = 0$, we obtain

$$\nabla_\theta D_{\mathrm{KL}}(q_\theta \| \tau)|_{\theta=\bar{\theta}} = -\mathbb{E}_{z\sim q_{\bar{\theta}}}\left[A(z)\, \nabla_\theta \log q_\theta(z)|_{\theta=\bar{\theta}}\right]. \tag{12}$$

Thus, the unclipped token loss $-\,\mathrm{sg}[A(z)] \log q_\theta(z)$ matches this gradient in expectation at $\theta = \bar{\theta}$, before role-length weighting. At $\theta \neq \bar{\theta}$, the fixed-snapshot update need not match the current policy's local reverse-KL gradient. Clipping bounds extreme token weights, while the trajectory-dependent factor $1/N_r(Y)$ averages each role's contribution. These operations modify the gradient, so Equation 2 is a local surrogate evaluated on fixed sampled contexts.

Context boundaries. Perception receives the video, the current self-contained subquestion, and the preceding tokens of its own answer. It does not receive the original answer options or other interaction history. The Planner receives the video, original question, and accumulated history available before each output. For a `DIRECT` answer, the frozen base teacher and student G role receive the original video and question, including any answer options, without the Planner's private prefix or mode-choice text. Rollout generation, teacher scoring, and student loss computation use the same information boundaries, with consistent token positions within each model's template. For `DIRECT`, π_{B} uses its native answer template, while G uses the shared prefix and G-role template. The G attention residual is trained to align their output distributions. Completed-sequence scoring can process known target tokens in parallel, while the interaction itself remains autoregressive. Teacher adapters are activated separately for scoring, with a fresh decoder prefill for the corresponding role context. For direct base scoring, the PC adapter is disabled and the original video–question context receives a fresh base prefill. Shared frozen visual features are cached, but decoder KV states are not reused across teacher adapter changes.

B.3 Inference

Routing and interaction. The initial tool name determines the mode: `answer_with_base` selects `DIRECT`, while `ask_perception` selects `PLAN`. For `PLAN`, the remainder of the initial request contains the first subquestion or subquestions. Both modes are available on every benchmark. The execution loop parses `answer_with_base` to activate G for direct answering, or `ask_perception` to activate V for the requested observations. Each request u_j contains one or more `<tool_call>` blocks, each closed by `</tool_call>`. Each block contains an `ask_perception` call with one `sub_question` argument. Independent subquestions may be issued in the same request. A subquestion that depends on an earlier observation is generated in a subsequent Planner turn, with the required information included explicitly in its wording. After all calls in the request have completed, the subquestions and their corresponding observations are appended to the Planner's history in request order. Execution then returns to PC, which can request further evidence or compose the final answer. No external teacher or spatial tool is called during student inference.

Interaction distribution. We formalize the interaction protocol introduced in Section 3 for both the teacher system and the student. Let ρ_{PC}, ρ_{V}, and ρ_{D} denote the planning-and-composition, perception, and direct answering policies of the system under consideration. For the teachers, these policies are $(\pi_{\mathrm{PC}}, \pi_{\mathrm{V}}, \pi_{\mathrm{B}})$. For the student, they are $(\pi_\theta^{\mathrm{PC}}, \pi_\theta^{\mathrm{V}}, \pi_\theta^{\mathrm{G}})$.

At planning round j, the request u_j contains $m_j \geq 1$ self-contained subquestions $\tilde{q}_{j,1}, \ldots, \tilde{q}_{j,m_j}$. Perception produces the corresponding answers $o_j{=}(o_{j,1}, \ldots, o_{j,m_j})$. The interaction history after j rounds is $H_j{=}((u_1, o_1), \ldots, (u_j, o_j))$, with $H_0{=}()$.

Let $p(b, y \mid x)$ denote the probability of the initial mode choice and complete output given input x. Let $p_{\mathrm{int}}(y \mid x, b)$ denote the conditional distribution of the interaction under `PLAN`. These distributions factorize as

$$
\begin{aligned}
p(b, y \mid x) &= \rho_{\mathrm{PC}}(b \mid x) \begin{cases} \rho_{\mathrm{D}}(a \mid x), & b = \texttt{DIRECT}, \\ p_{\mathrm{int}}(y \mid x, b), & b = \texttt{PLAN}, \end{cases} \\
p_{\mathrm{int}}(y \mid x, b) &= \prod_{j=1}^{J} \left[\rho_{\mathrm{PC}}(u_j \mid x, b, H_{j-1}) \prod_{k=1}^{m_j} \rho_{\mathrm{V}}(o_{j,k} \mid v, \tilde{q}_{j,k}) \right] \\
&\quad \times \rho_{\mathrm{PC}}(c \mid x, b, H_J).
\end{aligned}
\tag{13}
$$

Here $\rho_{\mathrm{PC}}(b \mid x)$ scores the initial tool choice: `answer_with_base` selects `DIRECT`, while `ask_perception` selects `PLAN`. For $b = \texttt{PLAN}$, u_1 denotes the remaining tokens of the initial request after the route-selecting tool name, so the two factors cover disjoint token positions. Subsequent Planner outputs either request additional observations or compose the final answer, which determines J during generation. The total number of subquestions is $\sum_{j=1}^{J} m_j$. For the student, the resulting trajectory distribution is denoted by p_θ, while the snapshot used for MOPD rollouts induces $p_{\bar{\theta}}$.

Table 9: Student improvements over their corresponding base models and 95% paired bootstrap confidence intervals. Gains and intervals are in percentage points.

Backbone	Benchmark	Gain	95% CI
Qwen3-VL-8B	VSI-Bench	4.2	[+3.4, +5.0]
	EgoProx	14.9	[+12.7, +17.1]
	R3D-Bench	8.5	[+7.2, +9.7]
SenseNova-SI-1.3-8B	VSI-Bench	2.1	[+1.6, +2.6]
	EgoProx	13.0	[+10.8, +15.3]
	R3D-Bench	7.4	[+6.3, +8.5]

Common prefix and KV reuse. The student encodes the video once and prefills a common prefix containing only information shared by all roles. Prefix tokens follow the frozen base path: the general feedforward expert is used, and every role attention residual is disabled at these positions. Consequently, the prefix computation is independent of the current role and subquestion. Role instructions, the original question, and answer options occur after the common prefix, within their permitted contexts. All student roles reuse the prefix KV states while processing their own text contexts. The same prefix path, visibility rules, and position assignments are used in training and inference. Perception therefore reuses the video prefix without gaining access to the Planner's private text context. After a `DIRECT` choice, G reuses the frozen video prefix and processes its own original-question context with the general attention residual trained during MOPD. It does not reuse the PC text-suffix KV states. The teacher comparison also caches the shared visual features, but prefills the decoder context separately for its role adapters and for the base with adapters disabled. The student completes the interaction autoregressively within one checkpoint without adapter switching.

C Additional Experimental Details and Results

This section supplements the experiments in Section 4 with bootstrap confidence intervals, complete results by subtask, direct-answer training baselines, inference efficiency, and gradient analysis of multi-teacher distillation.

C.1 Statistical Analysis

We compare each student with its corresponding base model using a paired bootstrap over videos with 10,000 resamples. Each resample includes all questions from the sampled videos, with identical sampling indices used for both models. We recompute the benchmark scores for each resample and obtain the 95% confidence interval from the 2.5th and 97.5th percentiles of the student-minus-base score differences. As shown in Table 9, all six intervals exclude zero, indicating statistically significant improvements across both backbones and all three benchmarks.

C.2 Results by Subtask

Tables 10, 11, and 12 report the complete subtask results for both backbones. Both EgoP^{2}C students outperform their respective base models on every evaluated subtask. The improvements span both individual spatial measurements and tasks that require combining multiple spatial observations.

For individual measurements, EgoP^{2}C $_{\textbf{Qwen3-VL}}$ and EgoP^{2}C $_{\textbf{SenseNova-SI}}$ gain 0.4 and 0.1 points on VSI-Bench's *Object Size Estimation*, which asks for an object's longest dimension. On R3D-Bench, they gain 13.1 and 8.8 points on *Global How Long*, which likewise requires estimating an object's longest dimension. These results show improvements on questions that require extracting geometric measurements from video.

We also observe substantial gains on tasks involving comparisons, aggregation, or goal-conditioned spatial relations. On VSI-Bench, the students improve by 5.6 and 3.2 points on *Object Relative Distance*, which requires comparing candidate objects by their distance to a reference. On R3D-Bench, they gain 17.7 and 14.8 points on *Total Fly Distance*, which requires combining two straight-line distance estimates. On EgoProx, they gain 23.1 and 19.9 points on *Intention Relative*, which

Table 10: **Results by subtask on VSI-Bench.** Scores (%) use mean relative accuracy (MRA) for numerical questions and accuracy for multiple-choice questions. n is the number of evaluated questions. **Bold** indicates student scores. Δ is the improvement over the corresponding base.

		Qwen3-VL-8B			SenseNova-SI-1.3-8B		
Subtask	n	Base	EgoP^{2}C	Δ	Base	EgoP^{2}C	Δ
Object Relative Direction	968	57.23	**58.88**	+1.65	63.33	**63.74**	+0.41
Object Size Estimation	953	84.66	**85.04**	+0.38	84.93	**85.05**	+0.12
Object Absolute Distance	834	42.49	**49.65**	+7.16	45.97	**50.36**	+4.39
Object Relative Distance	710	48.03	**53.66**	+5.63	59.15	**62.39**	+3.24
Object Appearance Order	618	69.42	**70.23**	+0.81	74.60	**74.92**	+0.32
Object Counting	565	64.99	**67.38**	+2.39	79.56	**79.88**	+0.32
Room Size Estimation	288	61.84	**66.60**	+4.76	67.05	**69.38**	+2.33
Route Planning	194	34.54	**45.36**	+10.82	43.81	**49.48**	+5.67
Overall	5,130	57.90	**62.10**	+4.20	64.80	**66.90**	+2.10

Table 11: **Results by subtask on EgoProx.** Scores are multiple-choice accuracy (%). n is the number of evaluated questions. **Bold** indicates student scores. Δ is the improvement over the corresponding base.

		Qwen3-VL-8B			SenseNova-SI-1.3-8B		
Subtask	n	Base	EgoP^{2}C	Δ	Base	EgoP^{2}C	Δ
Exploitation Relative	693	34.63	**46.75**	+12.12	34.92	**49.64**	+14.72
Exploitation Approx	422	37.68	**46.21**	+8.53	41.47	**47.16**	+5.69
Intention Approx	386	32.64	**42.23**	+9.59	41.45	**49.48**	+8.03
Intention Relative	342	29.24	**52.34**	+23.10	30.70	**50.58**	+19.88
Exploration Relative	191	17.28	**46.60**	+29.32	15.71	**39.79**	+24.08
Exploration Approx	187	16.58	**36.90**	+20.32	25.67	**34.76**	+9.09
Overall	2,221	31.02	**45.88**	+14.86	34.22	**47.19**	+12.97

requires interpreting head or gaze movement relative to a specified goal. Together, these results show gains across distinct forms of spatial reasoning, including object-level estimation, relative comparison, and distance aggregation.

C.3 Comparison with Direct-Answer Training

We examine whether direct-answer training on the same supervision can match Dual-LoRA and the SenseNova-SI student. All methods start from SenseNova-SI-1.3-8B. The direct-answer baseline uses a single rank-32 LoRA on the same language-model layers and projections as Dual-LoRA, matching the total trainable adapter parameters of its two rank-16 teachers.

SFT uses the same original questions and subquestion QA from the Planner and Perception training pools, with the same answer sources and filtering criteria. Original questions supervise only the final answer, while subquestion QA is treated as independent video question answering. GRPO uses the same Perception samples and original multiple-choice questions as Dual-LoRA, with weak labels as reward references when subquestion ground truth is unavailable and final-answer correctness for original questions. We use the same SFT and GRPO hyperparameters as the Dual-LoRA teachers.

We evaluate checkpoints after SFT and SFT+GRPO. At evaluation, the baseline answers the original question directly from the video and provided answer options, without subquestions or role interaction. For Dual-LoRA and the student, we report score differences relative to Direct-answer SFT+GRPO with 95% confidence intervals from 10,000 paired bootstrap resamples over videos.

Table 13 shows that direct-answer SFT improves over the base model on all three benchmarks, and GRPO provides further gains. Dual-LoRA nevertheless significantly outperforms Direct-answer

Table 12: **Results by subtask on R3D-Bench.** Scores (%) use mean relative accuracy (MRA) for numerical questions and accuracy for multiple-choice questions. n is the number of evaluated questions. **Bold** indicates student scores. Δ is the improvement over the corresponding base.

		Qwen3-VL-8B			SenseNova-SI-1.3-8B		
Subtask	n	Base	EgoP^{2}C	Δ	Base	EgoP^{2}C	Δ
Total Walk Distance	228	6.36	**19.08**	+12.72	7.28	**19.08**	+11.80
Total Fly Distance	228	9.56	**27.28**	+17.72	10.04	**24.82**	+14.78
Global How Far	228	27.19	**39.69**	+12.50	33.25	**44.04**	+10.79
Global How Far From Me	228	25.57	**34.30**	+8.73	33.99	**43.77**	+9.78
Global How Long	228	48.95	**62.06**	+13.11	63.38	**72.15**	+8.77
Nearest From Set	228	42.54	**46.49**	+3.95	42.98	**46.05**	+3.07
How Much Longer Dimension	228	34.17	**42.28**	+8.11	37.11	**45.88**	+8.77
Which Longer Dimension	228	67.11	**76.32**	+9.21	75.00	**77.63**	+2.63
Which Taller	227	70.04	**70.48**	+0.44	69.16	**70.48**	+1.32
How Much Taller	226	30.18	**37.48**	+7.30	35.49	**44.25**	+8.76
Top Higher	224	60.27	**66.07**	+5.80	59.82	**62.50**	+2.68
Volume Estimation	168	35.30	**40.12**	+4.82	14.23	**21.85**	+7.62
Pour Room Left	151	2.45	**10.40**	+7.95	1.92	**9.67**	+7.75
Pour Leftover	151	0.86	**6.82**	+5.96	0.86	**6.62**	+5.76
Gap Fit	62	72.58	**75.81**	+3.23	56.45	**59.68**	+3.23
Overall	3,033	35.20	**43.70**	+8.50	37.10	**44.50**	+7.40

Table 13: **Comparison with direct-answer training.** For Dual-LoRA and EgoP^{2}C, Δ reports score differences in percentage points relative to Direct-answer SFT+GRPO, with 95% paired bootstrap confidence intervals.

	VSI-Bench		EgoProx		R3D-Bench	
Method	**Score**	Δ [95% CI]	**Score**	Δ [95% CI]	**Score**	Δ [95% CI]
Base	64.8	–	34.2	–	37.1	–
Direct-answer SFT	65.4	–	39.8	–	39.3	–
Direct-answer SFT + GRPO	65.5	–	40.7	–	40.1	–
Dual-LoRA teachers	67.6	+2.1 [+1.8, +2.6]	49.0	+8.3 [+6.2, +10.5]	44.8	+4.7 [+3.3, +6.1]
EgoP^{2}C	66.9	+1.4 [+0.7, +2.2]	47.2	+6.5 [+4.6, +8.5]	44.5	+4.4 [+3.2, +5.6]

SFT+GRPO by 2.1, 8.3, and 4.7 points on VSI-Bench, EgoProx, and R3D-Bench, respectively. The student also exceeds this baseline by 1.4, 6.5, and 4.4 points. Thus, the available QA supervision benefits direct answering, but does not close the gap to either trained plan-perceive-compose system. These results support the complete pipeline over direct-answer training on the same training data and settings.

C.4 Additional Inference Efficiency Analysis

For each student, we select the evaluation questions on which it and Dual-LoRA naturally generate the same number of subquestions. The Qwen3-VL and SenseNova-SI subsets contain 1,110 and 2,118 questions, respectively. Each subquestion corresponds to one Perception call. The Qwen3-VL and SenseNova-SI subsets average 2.3 and 2.1 Perception calls per question, respectively. Dual-LoRA costs are reported on the questions selected for each student.

On the Qwen3-VL matched subset, Table 14 shows that the student reduces FLOPs by 68.5% and latency from 8.8 to 5.8 seconds, a 1.52× speedup. It generates 16.7% fewer tokens, and end-to-end latency per generated token decreases from 29.5 to 23.3 ms. On the SenseNova-SI matched subset, the student reduces FLOPs by 68.2% and latency from 8.6 to 4.9 seconds, a 1.75× speedup. It generates 21.1% fewer tokens, while latency per generated token decreases from 31.0 to 22.4 ms.

Table 14: **Inference cost with matched subquestion counts.** Each group compares one student with Dual-LoRA on its own matched subset. Metrics follow the definitions in Table 2.

Method	TFLOPs	Latency (s)	Generated tokens	Latency/token (ms)
Subset matched to the Qwen3-VL student				
Dual-LoRA $_{\text{SenseNova-SI}}$	162.1	8.8	298.4	29.5
EgoP^{2}C $_{\text{Qwen3-VL}}$	**51.0**	**5.8**	**248.5**	**23.3**
Subset matched to the SenseNova-SI student				
Dual-LoRA $_{\text{SenseNova-SI}}$	156.3	8.6	277.5	31.0
EgoP^{2}C $_{\text{SenseNova-SI}}$	**49.7**	**4.9**	**219.0**	**22.4**

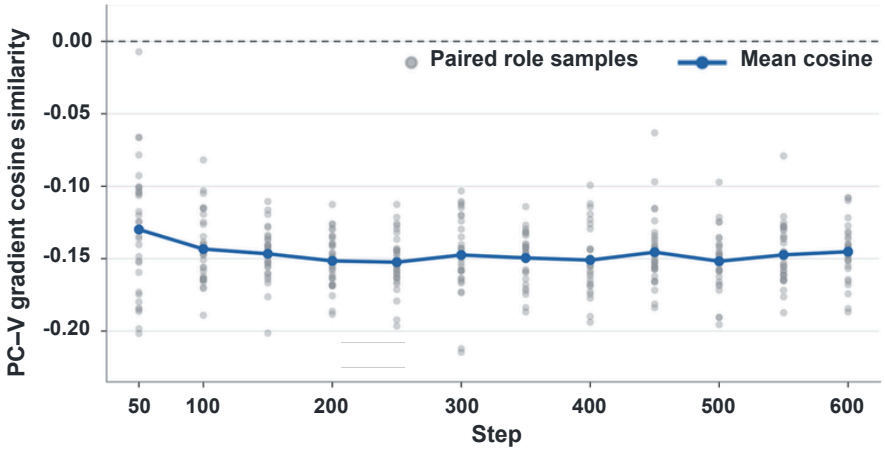


Figure 5: **Gradient alignment between planning and composition (PC) and perception (V) in the dense Qwen student.** Each gray point represents the gradient cosine similarity for one pair of PC and V samples. Blue markers show the mean over 32 pairs at each checkpoint. The same sample pairs are evaluated across all 12 checkpoints.

Both students therefore retain an efficiency advantage when the number of subquestions is matched per original question.

C.5 Gradient Analysis of Multi-Teacher Distillation

The main results compare final task performance and student-teacher KL. Here, we additionally examine whether the Planner (planning and composition) and Perception objectives produce conflicting update directions when learned through shared parameters.

We analyze the dense Qwen3-VL student at 12 checkpoints during MOPD. During training, we update only the FFNs in decoder layers 24-35, corresponding to the layers converted into experts in the MoE student. All remaining parameters are frozen. At each checkpoint, we evaluate the same 32 pairs of PC and V samples. For each pair, we compute the gradients of the two role losses with respect to these shared trainable parameters. Let g_{PC} and g_{V} denote the resulting gradients.

Their cosine similarity is

$$c = \frac{g_{\text{PC}}^{\top} g_{\text{V}}}{\|g_{\text{PC}}\|_2 \|g_{\text{V}}\|_2}. \tag{14}$$

A negative value indicates that the two gradients point in opposing directions: a sufficiently small gradient descent step on one role loss would increase the other to first order.

As shown in Figure 5, all 384 measurements are negative, with an overall mean of -0.147. The disagreement persists across the inspected checkpoints and sample pairs, indicating competing update directions over the evaluated parameters. This provides additional evidence of the difficulty of learning both specialist objectives through shared parameters.

D PROMPT TEMPLATES

We provide prompts for demonstration generation, Dual-LoRA teacher inference, and student inference. Figures 6 and 7 show the prompts used by DeepSeek-V4-Pro and Qwen3.5-397B. DeepSeek generates `PLAN` demonstrations, and `DIRECT` examples are constructed as described in Appendix A.1. The DeepSeek `question` field includes the answer options when applicable. The correct final answer is withheld during generation and used for verification.

The generated JSON records are converted into training conversations. Each subquestion is formatted as an `ask_perception` tool call, and the composition is formatted as the final Planner response. For generated observations, the `answer` field supplies the Perception target.

Figures 8 and 9 show the Dual-LoRA inference prompts. The initial tool choice implements the mode selection: `answer_with_base` selects `DIRECT`, while `ask_perception` starts the `PLAN` interaction. Subsequent calls and observations remain in the Planner's conversation history. Each Perception call receives only the video and its current subquestion.

Figure 10 shows the student prompt template. Role instructions and permitted text inputs follow a shared video prefix. Within the student, `answer_with_base` activates the General expert, while `ask_perception` activates the Perception expert. After receiving observations, the Planner expert can request further evidence or compose the final answer. For both Dual-LoRA and student inference, the `answer_format_instruction` field uses the original answer format instruction provided by the corresponding benchmark.

E QUALITATIVE EXAMPLES

Figure 11 illustrates two forms of subquestion generation. In the R3D-Bench example (top), Qwen3-VL-8B mistakes the dish's larger overall size for greater vertical height. EgoP^{2}C requests the heights of both objects through *independent subquestions within one planning round*. Neither subquestion requires the other's answer, so both can be issued before receiving observations. The returned estimates of 2 cm for the dish and 4 cm for the banana support the correct height comparison.

In the EgoProx example (bottom), SenseNova-SI-1.3-8B identifies the action and movement direction correctly but overestimates the remaining movement distance. EgoP^{2}C uses *dependent subquestions across three planning rounds*. Each subsequent question builds on observations from the preceding round. After identifying the ongoing chain lubrication, the Planner asks about the next action and hand movement. The predicted forward motion then motivates a question about the remaining distance to the end of the chain section in the last frame. The Planner combines the action, direction, and distance observations to select the correct option.

System prompt

You are a spatial reasoning planner. Determine what spatial evidence is needed to answer a question and reason from the observations provided. Do not invent observations or measurements.

User prompt template

Task

Decide the next action for the question below using the observations collected so far. Choose one action for this turn:

– `ask_perception`: Request one or more spatial observations needed to answer the question.

– `compose`: Combine the available observations into the final answer when sufficient evidence is available.

Requirements

– Request the spatial facts needed to answer the question. Do not pass the original option list to Perception or ask it to select an option label.

– You may group independent subquestions in one request. If a subquestion depends on an earlier answer, wait for that answer and include the necessary information in the new subquestion.

– Use only the question, its options, and observations already returned. Do not treat answer options as established facts. After requesting observations, wait for their responses.

– For `compose`, compare measurements or perform calculations as needed, without introducing unsupported visual facts. Follow the required answer format.

Output format

Briefly explain your action in `thought`. Return exactly one JSON object in the corresponding format below, with no Markdown or additional text.

Perception request

```
{
  "thought": "<brief reason>",
  "action": "ask_perception",
  "sub_questions": [
    {
      "sub_question": "<atomic question>"
    }
  ],
  "compose_answer": null
}
```

Final composition

```
{
  "thought": "<brief reasoning>",
  "action": "compose",
  "sub_questions": [],
  "compose_answer": "<final answer>"
}
```

Input

```
Question: {question}
Required answer format: {answer_format}
Interaction history: {history}
Current round: {t}
```

Figure 6: **Prompt template for generating `PLAN` demonstrations with DeepSeek-V4-Pro.** The Planner requests spatial observations or composes an answer from the returned evidence.

System prompt

You answer spatial questions from video frames. Base your answer on the visual evidence and do not invent observations or measurements.

User prompt template

Task

The attached frames are sampled in chronological order from a video. Answer the question below as directly as possible.

Requirements

– Answer only the requested fact, relation, or measurement, using evidence from the frames.

– Follow the temporal scope, reference frame, and units specified in the question.

– For questions about distance or size, provide an estimate when the visual evidence supports one.

Output format

Return exactly one JSON object with the following fields. Do not include Markdown or text outside the JSON. Use a number between 0 and 1 for `confidence`, and a JSON boolean for `answerable`.

```
{
  "answer": "<short direct answer or statement of uncertainty>",
  "confidence": <number between 0 and 1>,
  "visible_evidence": "<brief visual support or limitation>",
  "answerable": <true or false>
}
```

Input

```
Video frames: {video_frames}
Question: {sub_question}
```

Figure 7: **Prompt template for Perception answer generation with Qwen3.5-397B.** The model receives video frames and one self-contained spatial subquestion, without the original answer options or planning history.

System prompt

You are a spatial reasoning planner for egocentric video question answering. Determine what evidence is needed and use it to answer the question.

User prompt template

Task

Answer the question below using the video. At the beginning, choose how to answer:

– If the question can be answered directly from the clip as a whole, call `answer_with_base`.

– Otherwise, call `ask_perception` to obtain the spatial observations needed to answer the question.

Requirements

– You may issue multiple independent calls in one turn. If a subquestion depends on an earlier observation, wait for that observation and include the relevant information explicitly in the new subquestion.

– Once you have requested observations, continue gathering evidence or compose the final answer. Do not switch to `answer_with_base`.

– Use only the observations needed to answer the question. When sufficient evidence is available, combine the observations and give the final answer directly in your assistant response, following the question's answer format.

Output format

Use the following formats for tool calls. Composition is a final assistant response.

Request one observation

```
<tool_call>
{"name": "ask_perception",
 "arguments": {
   "sub_question": "<question>"}}
</tool_call>
```

Request a direct answer

```
<tool_call>
{"name": "answer_with_base",
 "arguments": {}}
</tool_call>
```

Final assistant response

```
<think>{composition}</think>
{final_answer}
```

Input

```
<video>
Question: {question}
{options_if_applicable}
{answer_format_instruction}
```

Figure 8: **Planner prompt for Dual-LoRA inference.** The Planner initially chooses direct answering or perception requests. After gathering observations, it composes the answer in its final assistant response.

System prompt

You answer spatial questions from egocentric video.

User prompt template

Task

Answer the subquestion below using the visual evidence in the video.

Requirements

– Answer the requested fact, relation, or measurement.

– Follow the reference frame, temporal scope, and units specified in the subquestion.

– If the evidence is insufficient, state what cannot be determined.

Output format

Return a concise answer in plain text, following any answer format instruction below.

Input

```
<video>
{sub_question}
{answer_format_instruction}
```

Figure 9: **Perception prompt for Dual-LoRA inference.** Each call contains the video, one self-contained subquestion, and an optional answer format instruction. Perception receives no original answer options or planning history.

Shared prefix: common system instruction and video

You answer spatial questions about egocentric videos. Use the available visual evidence and follow the requested answer format.

```
<video>
```

Planner (PC): choose a route, request evidence, and compose

Role instruction

First turn. Decide whether the original question needs intermediate observations. At the first turn, call `answer_with_base` for direct answering, or `ask_perception` to request evidence.

Subquestions. Ask one self-contained atomic question per call. Name the objects and specify the reference frame, temporal scope, and units when needed. Do not pass the original option list or ask for an option label. You may issue independent calls together. For dependent questions, wait for the earlier answer and include the necessary information explicitly. After issuing calls, wait for their responses.

After observations. Request further evidence or combine the observations and give the final answer. Do not switch to `answer_with_base` after requesting evidence. Avoid repeating answered questions unless their answers are unclear or inconsistent. When sufficient evidence is available, output the final answer in the requested format.

Initial input

```
Question: {question}
{options_if_applicable}
{answer_format_instruction}
```

Subsequent input

```
Observations: {subquestion_answer_pairs_in_request_order}
```

Earlier requests and observations remain in the Planner conversation.

PLAN: request an observation

```
<tool_call>
{"name": "ask_perception",
 "arguments": {
   "sub_question": "{sub_question}"}}
</tool_call>
```

DIRECT: initial turn only

```
<tool_call>
{"name": "answer_with_base",
 "arguments": {}}
</tool_call>
```

PLAN: final composition

```
{composition}
{final_answer}
```

Perception (V): answer one subquestion

Role instruction

Answer the current atomic spatial question using the video. Provide the requested fact, relation, or measurement. Follow the specified reference frame, temporal scope, units, and answer format. Return a concise plain-text answer. If evidence is insufficient, state what cannot be determined.

Input after the shared prefix

```
{sub_question}
{answer_format_instruction}
```

Output

```
{atomic_answer}
```

General (G): answer directly

Role instruction

Answer the original question directly using the video. Do not generate subquestions or request observations. Output the final answer in the requested format.

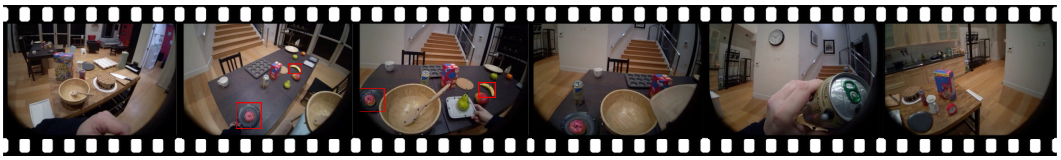
Input after the shared prefix

```
Question: {question}
{options_if_applicable}
{answer_format_instruction}
```

Output

```
{final_answer}
```

Figure 10: **Student prompt template.** All roles share a video prefix and receive separate text contexts. The call syntax denotes internal role dispatch: `ask_perception` activates V, and `answer_with_base` activates G. After observations return, PC can request further evidence or compose the final answer. `DIRECT` and `PLAN` label the routes; they are not additional outputs in this template.



Question: Between the large black ceramic dish and the banana, which one is taller? **GT: banana** **R3D-Bench**

Qwen3-VL-8B: The large black ceramic dish is positioned more distantly and is larger in size compared to the banana, **indicating it is taller**. **The banana is closer to the camera but is significantly shorter in height**. The large black ceramic dish's greater size and distance suggest it is the taller object. **<answer>large black ceramic dish.</answer>** ✗

EgoP^{2}C$_{\text{Qwen3-VL-8B}}$

Planner: I need to obtain the height of each object before comparing them. Use sub-question 1 to determine the vertical height of the large black ceramic dish above the tabletop. Use sub-question 2 to determine the vertical height of the banana in its resting position above the tabletop. Compare the two heights and answer with the name of the taller object.

- **<tool_call>** {"name": "ask_perception", "arguments": {"sub_question": "What is the vertical height of the large black ceramic dish above the tabletop? Answer in centimeters."}} **</tool_call>**
  - **Perception: 2.00 cm.**
- **<tool_call>** {"name": "ask_perception", "arguments": {"sub_question": "What is the vertical height of the banana in its resting position above the tabletop? Answer in centimeters."}} **</tool_call>**
  - **Perception: 4.00 cm.**

Planner: The **banana is 4.00 cm high**, compared with **2.00 cm for the large black ceramic dish**. Therefore, the **banana is taller**. **<answer>banana.</answer>** ✓



Question: Given the recent motion, what is the most likely next interaction? **GT: C** **EgoProx**

A. left hand adjust rear gear shifter ~0.1 m.
B. left hand spin pedal ~0.4 m.
C. right hand move forward ~0.3 m to pour lubricant onto the chain.
D. right hand move forward ~0.9 m to pour lubricant onto the chain.
E. right hand reach ~1.1 m for a tool.

SenseNova-SI-1.3: The person is holding the lubricant in the right hand and is moving it forward toward the bicycle chain. The chain is directly ahead of the right hand, so the next action is likely to continue reaching forward and pour lubricant onto the chain. **The visible chain appears to span a long section of the bicycle**. Therefore, the person will most likely **move the right hand forward 0.9 m to pour lubricant onto the chain**. **<answer>D.</answer>** ✗

EgoP^{2}C$_{\text{SenseNova-1.3-8B}}$

Planner: I first need to identify the current action and the object involved. Use sub-question 1 to determine what the person is doing and which object they are interacting with.

- **<tool_call>** {"name": "ask_perception", "arguments": {"sub_question": "What is the person currently doing, and which object are they interacting with?"}} **</tool_call>**
  - **Perception: The person is applying lubricant to the bicycle chain with the right hand.**

Planner: The person is applying lubricant to the bicycle chain. Use sub-question 2 to determine the most likely next action and the direction in which the hand will move.

- **<tool_call>** {"name": "ask_perception", "arguments": {"sub_question": "What is the most likely next action, and in which direction will the hand move?"}} **</tool_call>**
  - **Perception: The person will continue applying lubricant, moving the right hand forward along the chain.**

Planner: The person will likely apply lubricant to the end of the chain. Options C and D differ only in distance: 0.3 m versus 0.9 m. Use sub-question 3 to determine which matches the remaining distance.

- **<tool_call>** {"name": "ask_perception", "arguments": {"sub_question": "In the last frame, what is the distance from the right hand to the far end of the chain section being lubricated? Answer in meters."}} **</tool_call>**
  - **Perception: 0.5 m.**

Planner: The person will **continue applying lubricant to the chain**, **moving the right hand forward** toward the end of the section. The remaining distance is **0.5 m**, which is **closer to 0.3 m in option C**. **<answer>C.</answer>** ✓

Figure 11: **Illustrative comparisons on R3D-Bench (top) and EgoProx (bottom).** **Red text** marks incorrect reasoning and answers, while **green text** marks correct ones.