HAPRL: HUMAN-ANCHORED PROCESS REINFORCEMENT LEARNING FOR VISUAL SEARCH AGENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-turn visual search agents answer questions about high-resolution images by iteratively deciding where to look. Reinforcement learning for these agents rewards only the final answer, leaving the search process unsupervised. Consequently, faulty routes in which the reasoning process is erroneous yet the final result is correct arise frequently, which *in turn leads to ineffective training, i.e., scaling along the wrong paths.* In this paper, we introduce HaPRL, the first framework to *reinforce the search process with human search behavior*. We first build an annotation platform and collect $1K+$ human-annotated data with fine-grained behavioral signals. During training, a carefully designed judge scores each rollout with task-adaptive weights, anchored on the distilled trace of how a human annotator actually searched the same image. Extensive experiments show that HaPRL consistently outperforms outcome-based RL, and early-stage process supervision yields $6.7\times$ more improvement in subsequent outcome-based scaling. Our results also demonstrate the importance of aligning model behavior with human process annotation signals, which offer new insight into the training of foundation models.

1 INTRODUCTION

High-resolution visual reasoning often depends on evidence that occupies only a small fraction of an image. Human vision handles this constraint through sequential attention, turning a series of fixations into an efficient search process (Treisman & Gelade, 1980; Itti & Koch, 2001). Multi-turn visual agents follow this principle by interleaving reasoning with crop-and-zoom tool calls, i.e., they choose a region, inspect the resulting view, and decide where to look next (Wu & Xie, 2024; Shao et al., 2024a; Hu et al., 2024; Zheng et al., 2025; Wang et al., 2025a; Lai et al., 2025). A capable agent should reach the correct answer through a search trajectory that acquires the supporting evidence.

Training a capable visual search agent through supervised fine-tuning alone amounts to off-policy behavior cloning, which limits generalization beyond the demonstration distribution (Figure 1). Many methods therefore apply reinforcement learning after a cold start to unlock the agent's potential. Existing visual-search agents commonly adopt outcome-based rewards that score only the final answer (Zheng et al., 2025; Wang et al., 2025a; Lai et al., 2025; Liu et al., 2025). Such outcome-based rewards are attractive because correctness is inexpensive to verify (Shao et al., 2024b; DeepSeek-AI, 2025; Lambert et al., 2024). However, *human annotators also produce fine-grained behavioral signals through their mouse interactions during the search process*, such as movement trajectories, cursor velocity, dwell duration, etc.. *These signals carry rich information about search quality but are entirely discarded* by outcome-based training. As a result, a correct answer reached by an incorrect search hacks full reward. On multiple-choice and short-answer tasks, for example, an agent may inspect irrelevant regions, miss the target evidence, and still arrive at the right answer through random guessing. Rewarding such trajectories *reinforces faulty routes and causes optimization to scale along wrong search paths* (Figure 1).

More fundamentally, outcome-based GRPO (Shao et al., 2024b) standardizes rewards within each sampled group and provides no mechanism to rank correct trajectories by search quality. That is, a trajectory that localizes the sign, zooms in, and reads it receives the same credit as one that crops empty regions and guesses. We verify this on VisualProbe Hard, where outcome-based RL raises accuracy by 28.6% while the judged search quality of its correct answers *drops* by 8.4% (Section 4.2), i.e., *the agent answers more questions correctly yet searches worse doing so*.

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

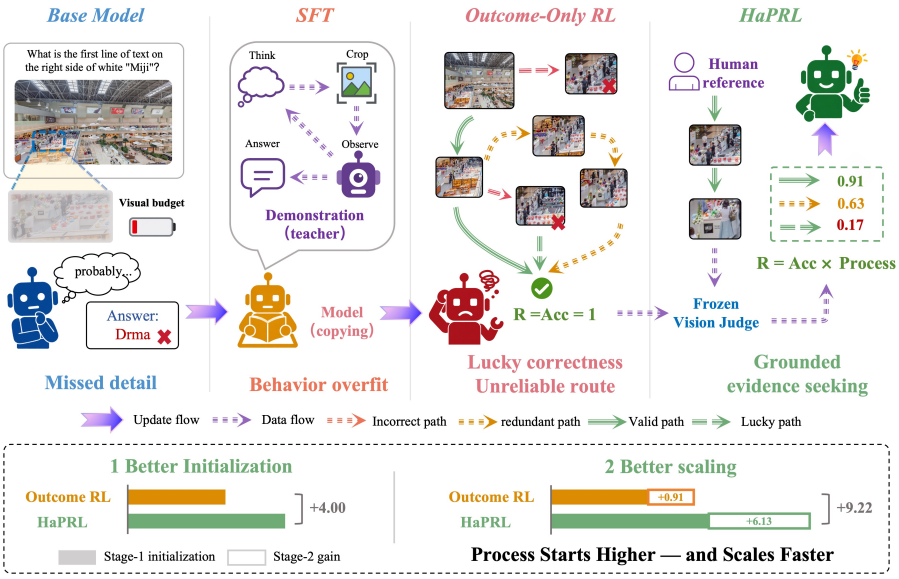


Figure 1: **Human-anchored process supervision improves both initialization and scaling.** *Top*: Base models miss fine-grained details; SFT enables tool use but overfits to demonstrations; Outcome-based RL rewards grounded, redundant, and lucky trajectories equally. HaPRL ranks search quality anchored on the human reference trace, steering the agent toward evidence-grounded search. *Bottom*: HaPRL yields both a stronger initialization (**+4.00** over outcome-based RL) and better scaling (**6.7×** gain from the same outcome-based continuation).

Process supervision addresses analogous failures in mathematical reasoning by evaluating intermediate steps (Uesato et al., 2022; Lightman et al., 2024; Wang et al., 2024; Luo et al., 2024), but *perceptual search does not decompose into self-contained steps* that can be labeled correct or incorrect. The value of a crop depends on the question, the regions already inspected, and whether the action makes progress toward the target. A useful process signal must therefore evaluate the trajectory as a whole rather than classify isolated coordinates. Human search behavior provides the missing reference, recording which fine-grained regions attracted attention, how the view was refined, and where the evidence was ultimately found. Addressing this requires solving two challenges: **G1) Human process signal acquisition**: systematically capture the fine-grained behavioral signals that human annotators produce during visual search; and **G2) Dense process alignment**: align agent training with these human signals to provide graded supervision.

For (G1), we build a browser-based annotation platform and collect human search traces of 1,104 visual-search questions. The platform logs timestamped pointer movements, hovers, and zooms over the full-resolution image, preserving how annotators attend to and refine fine-grained regions. Each interaction stream is distilled into a compact round-by-round trace of the inspected regions and discovered evidence. *These traces serve as instance-specific anchors that reveal where useful evidence lies and how much visual refinement the question requires.* For (G2), we introduce HaPRL, a training paradigm that aligns visual-search agents with these human process signals at the trajectory level. A frozen vision-language judge evaluates each rollout along five dimensions with task-adaptive weights, anchored on the human-annotated trace of the same image. The resulting process score is multiplied by binary answer correctness, so incorrect answers receive no credit while correct trajectories are ranked by search quality.

Our contributions can be summarized as follows.

- We develop a platform and collect 1,104 human-annotated visual-search examples with process-level information, distilled into round-by-round traces of human attention and evidence acquisition.
- We introduce HaPRL, the first training paradigm that aligns visual-search agents with human process signals. The answer-gated, task-adaptive reward ranks trajectories anchored on traces.

- Extensive experiments across four backbones and six benchmarks show that HaPRL consistently outperforms outcome-based RL. Moreover, process annotation scales favorably, i.e., *early-stage process supervision provides an initialization from which a subsequent outcome-based stage yields **6.7×** (6.13 vs. 0.91) more improvement than the same stage without it.*
- Our results highlight the importance of aligning agent behavior with human process annotation signals, which offers practical insights for future efforts beyond visual search.

2 Related Work

Visual Search Agents. Visual search enables multimodal large language models (MLLMs) to inspect selected regions instead of relying on a single global view. V* combines LLM-guided search with multimodal reasoning to locate small targets in high-resolution images (Wu & Xie, 2024), while Visual Sketchpad equips models with drawing and specialist vision tools for intermediate visual reasoning (Hu et al., 2024). Recent work learns such interactions through post-training. DeepEyes develops active perception with end-to-end reinforcement learning (Zheng et al., 2025), and Pixel Reasoner combines instruction tuning with curiosity-driven reinforcement learning over visual operations (Wang et al., 2025a). Chain-of-Focus learns adaptive search and zooming through SFT followed by outcome- and format-based reinforcement learning (Zhang et al., 2025), while Mini-o3 scales visual-search trajectories through diverse cold-start data and over-turn masking (Lai et al., 2025). Despite different tools and training recipes, *these methods rely primarily on terminal correctness or hand-designed auxiliary signals*. We instead use human search traces as instance-specific references for evaluating complete evidence-acquisition trajectories.

Process Supervision for Reasoning. Process supervision evaluates intermediate reasoning rather than relying exclusively on terminal outcomes. In mathematical reasoning, process-based feedback reduces reasoning errors among final-answer-correct solutions (Uesato et al., 2022), and PRM800K scales human step-level feedback to the MATH dataset (Lightman et al., 2024). Math-Shepherd and OmegaPRM reduce annotation costs by constructing process supervision automatically through repeated completions and tree search (Wang et al., 2024; Luo et al., 2024). Multimodal PRMs extend step-level verification to image-conditioned mathematical reasoning, i.e. VisualPRM and MM-PRM score candidate derivations for Best-of-N inference (Wang et al., 2025b; Du et al., 2025), whereas URSA also incorporates process rewards into online policy optimization (Luo et al., 2025). These methods assess textual reasoning steps whose correctness can be evaluated individually. *Visual search differs because every perceptual action changes the evidence available to subsequent decisions.* Accordingly, we evaluate the complete search trajectory against human behavior and use this trajectory-level signal for reinforcement learning.

Rubric-Based Reinforcement Learning. LLM judges provide scalable model-based evaluation (Zheng et al., 2023), and recent work converts structured criteria into reinforcement signals. Rubrics as Rewards uses instance-specific rubrics for on-policy training in medical and scientific domains (Gunjal et al., 2025). Reinforcement Learning from Checklist Feedback extracts instruction-specific criteria and aggregates judgments from language models and specialized verifiers (Viswanathan et al., 2025), while Rubric Anchors scales open-ended alignment with a large collection of human- and model-authored rubrics (Huang et al., 2025). *These methods primarily evaluate final responses when a single verifiable outcome is unavailable.* In visual search, the final answer is verifiable but does not reveal whether the agent acquired the necessary evidence. HaPRL therefore applies a task-adaptive rubric to the search process, anchors the evaluation on a matched human trace, and gates the process score by answer correctness.

3 Method

Method Overview. We study a visual-search policy π_θ that answers a question q about a high-resolution image I_0 by interleaving language reasoning with crop actions, following the tool-use protocol of Mini-o3 (Lai et al., 2025). At round t, the policy emits a reasoning segment h_t and either terminates with an answer a or invokes $c_t = (v_t, b_t)$, where v_t identifies a previously observed view and b_t specifies a normalized bounding box. The environment executes the crop and returns the

resulting observation o_t. A trajectory containing T crop actions is

$$\tau = (h_1, c_1, o_1, \ldots, h_T, c_T, o_T, h_{T+1}, a). \tag{1}$$

The central idea of HaPRL is to *use human search behavior as a reward anchor rather than a policy demonstration.* As shown in Figure 2, the policy samples multiple trajectories for each image-question pair. A frozen vision-language judge compares each search process with the matched human trace and produces a task-adaptive process score, while a separate answer judge evaluates terminal correctness. Gating the process score by correctness preserves the answer objective and introduces graded preferences among correct trajectories. Human traces affect training only through this reward pathway and never enter the policy context.

In Section 3.1, we describe how human interaction signals are collected and distilled into reference traces. In Section 3.2, we introduce the human-anchored process judge. In Section 3.3, we present the policy optimization with answer-gated reward. Algorithms 1 and 2 summarize the data-construction and training procedures.

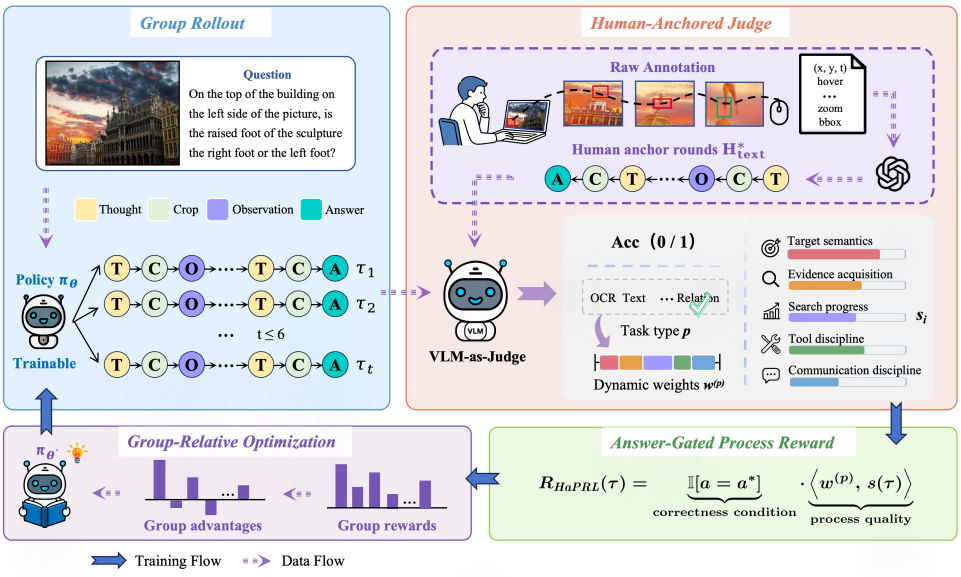


Figure 2: **HaPRL converts human search behavior into graded rewards for visual-search training.** Human interaction logs are distilled into reference traces. A frozen vision-language judge then compares each sampled trajectory with its matched trace along five task-adaptive dimensions, while answer correctness gates the weighted process score. Finally, group-relative optimization uses the resulting rewards to update the policy model.

3.1 Human Process Data Collection

Interaction logging. We build a browser-based platform and collect human search behavior for 1,104 questions from the VisualProbe training set. For each question, the human annotator searches the full-resolution image and uses the mouse to mark the region containing the supporting evidence. The platform automatically records the interaction stream $E^* = (e_1, \ldots, e_M)$ throughout this process. We represent each event as $e_m = (\kappa_m, t_m, \mathbf{p}_m, v_m, \alpha_m, b_m)$, where κ_m is the event type, t_m is its timestamp, $\mathbf{p}_m$ is the pointer position, v_m and α_m are its velocity and acceleration, and b_m is an optional zoom or selection box. *These measurements retain the temporal and spatial structure of the search*, including scan direction, dwell, redirection, and progressive refinement.

Trace distillation. Raw interaction streams are long, redundant, and misaligned with the discrete rounds of an agent trajectory. An offline distillation stage converts E^* into an ordered textual trace $H^* = (r_1^*, \ldots, r_L^*)$. Each round r_l^* records the inspected region, its source view, the normalized crop when available, and the evidence revealed at that stage. Consecutive pointer samples that do not change the inspected region are suppressed, while meaningful dwell, redirection, and coarse-to-fine

refinement are retained. The resulting training example is $z = (I_0, q, a^*, E^*, H^*)$, where a^* denotes the reference answer. The policy observes only (I_0, q). The distilled trace and annotation metadata are reserved for reward computation. Thus, H^* identifies the evidence requirements and search difficulty of an instance without prescribing the exact actions that the policy must reproduce.

3.2 Human-Anchored Process Judge

Human-anchored assessment. A frozen vision-language judge J_ϕ takes (I_0, q, τ, H^*) as input. The matched human trace anchors the evaluation to the evidence requirements and search difficulty of the specific instance without prescribing the annotator's exact actions. The judge returns a score vector $\mathbf{s}(\tau) \in [0,1]^5$ over five dimensions. (i) Target semantics measures whether the search follows the queried object, attribute, text, count, or relation. (ii) Evidence acquisition measures whether the observed views provide sufficient support for the answer. (iii) Search progress captures information gain across successive actions. (iv) Tool discipline evaluates the validity and granularity of crop operations. (v) Communication discipline evaluates whether the reasoning remains concise and grounded in visible evidence. More details are provided in Appendix A.3.

Task-adaptive scoring. Different visual-search tasks require different forms of evidence, so the five criteria should not contribute equally. The judge assigns each instance to a profile $p \in \{\text{OCR/text}, \text{count/relation}, \text{attribute}, \text{general search}\}$ and applies the corresponding weights $\mathbf{w}^{(p)}$. OCR/text places greater emphasis on evidence legibility, count/relation on spatial coverage and cross-region consistency, and attribute on precise target grounding. General search adopts a more balanced weighting across the shared criteria. Given the vector $\mathbf{s}(\tau)$, we compute the process score as:

$$r_{\text{proc}}(\tau) = \left\langle \mathbf{w}^{(p)}, \mathbf{s}(\tau) \right\rangle = \sum_{k=1}^{5} w_k^{(p)} s_k(\tau), \qquad \sum_{k=1}^{5} w_k^{(p)} = 1. \tag{2}$$

3.3 Answer-Gated Process Optimization

Correctness-gated reward. Process quality should *distinguish successful trajectories without rewarding an incorrect answer*. A separate frozen answer judge J_{ans} compares the extracted answer $a(\tau)$ with the reference a^*:

$$r_{\text{out}}(\tau) = J_{\text{ans}}(q, a(\tau), a^*) \in \{0, 1\}. \tag{3}$$

We invoke the process judge only when $r_{\text{out}}(\tau) = 1$ and define the training reward as

$$R_{\text{HaPRL}}(\tau) = r_{\text{out}}(\tau)\, r_{\text{proc}}(\tau). \tag{4}$$

The gate assigns zero reward to incorrect trajectories and ranks correct trajectories by evidence-acquisition quality. In contrast, the outcome-based baseline uses $R_{\text{Outcome-RL}}(\tau) = r_{\text{out}}(\tau)$ and assigns the same reward to grounded, redundant, and lucky correct trajectories.

Group-relative optimization. For each input $x = (I_0, q)$, we sample G trajectories $\{\tau_i\}_{i=1}^{G}$ and standardize their rewards within the group:

$$A_i = \frac{R_i - \bar{R}}{\sqrt{G^{-1} \sum_{j=1}^{G} (R_j - \bar{R})^2 + \epsilon}}, \qquad \bar{R} = G^{-1} \sum_{j=1}^{G} R_j. \tag{5}$$

An all-correct group receives a constant outcome reward and therefore yields zero relative advantage. HaPRL preserves within-group variation through r_{proc}, *allowing successful trajectories to be distinguished by search quality*. All-wrong groups remain at zero because the correctness gate suppresses process credit. We optimize these advantages with the clipped GRPO objective (Shao et al., 2024b):

$$\mathcal{L}(\theta) = -\mathbb{E}_{i,t}[\min(\rho_{i,t} A_i, \text{clip}(\rho_{i,t}, 1-\varepsilon, 1+\varepsilon) A_i)] + \beta D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}}), \tag{6}$$

where $\rho_{i,t} = \pi_\theta(y_{i,t} \mid x, y_{i,<t}) / \pi_{\theta_{\text{old}}}(y_{i,t} \mid x, y_{i,<t})$ is the token-level importance ratio, and π_{ref} is the reference policy used for KL regularization. At inference time, the learned policy interacts directly with the crop tool without judges or human traces.

4 EXPERIMENTS

Benchmarks and metrics. We evaluate on VisualProbe Easy/Medium/Hard (Lai et al., 2025), V*Bench QA (Wu & Xie, 2024) and HR-Bench 4K/8K (Wang et al., 2025c). Each agent answers every question with one greedy rollout, capped at a maximum of six tool rounds. Base models are evaluated by answering in a single pass with no tool calls, so their rows report a starting point rather than a search policy. All models share that visual token budget, so no gap below comes from more pixels. Besides, V*QA is graded open-ended by the judge rather than by option matching, which denies the policy the option-elimination shortcut. "Avg." is the unweighted mean over the six sets. Search quality is scored separately on *answer-correct* rollouts only, separating how an agent searched from how often it was right. See Appendix B.1 for more details.

Training. Four backbones span two families and scales: Qwen3-VL-4B/8B-Instruct (Bai et al., 2025) and LLaVA-OneVision-1.5-4B/8B (An et al., 2025). None emits the grounding syntax reliably on its own, so each is cold-started on filtered Mini-o3 trajectories (Lai et al., 2025). Two RL conditions follow. Outcome-RL optimizes answer correctness alone, HaPRL optimizes the answer-gated product.

Process annotation from Human. Human annotations cover $1{,}104$ training questions from VisualProbe. Annotators worked in a browser interface logging hover and zoom motion, and each event stream is distilled into a per-round textual trace.

Hyper-parameters. RL uses GRPO (Shao et al., 2024b) in verl (Sheng et al., 2025) with vLLM rollouts (Kwon et al., 2023): 8 rollouts per prompt, prompt batch 48, actor learning rate 5×10^{-7}, and KL coefficient 3×10^{-3}. The judge is a frozen Qwen3-VL-30B-A3B-Instruct. Both arms select checkpoints on the same held-out split. Remaining settings are in Appendices B.2–B.4.

4.1 BENCHMARKING OUTCOME-RL-BASED VLMS

Comprehensive Improvements. As shown in Table 1, HaPRL wins all 24 backbone-benchmark cells over Outcome-RL and the base model. Macro-average margins over matched Outcome-RL reach **+30.3%** on Qwen3-VL-4B (44.26 vs. 33.98) and **+15.3%** on LLaVA-OneVision-1.5-8B (47.80 vs. 41.46), with **+13.8%** and **+5.8%** on the other two.

SFT Trade-off Correction. Cold-start SFT costs 3.00 to 8.71 points of accuracy and enables a policy that can call the crop tool. Outcome-only RL fails to repay that debt on half the backbones, leaving Qwen3-VL-4B 6.09 points below its own base model (33.98 vs. 40.07). HaPRL clears the base model on all four, by **+4.19** to **+7.40** points. Qwen3-VL-4B on HR-Bench 4K is the cleanest case: the cold start gives up 6.62 points (48.88 vs. 55.50), Outcome-RL a further 1.75, and HaPRL recovers 7.62 of them (56.50). The cold start is a wager that tool use is worth an accuracy deficit, and *only the process reward collects on it*.

Hard-search Gains. A binary outcome reward takes two values inside a sampled group, so its advantage is zero among the correct rollouts and zero among the incorrect ones. The only gradient it supplies separates right answers from wrong ones. *HaPRL gates a rubric score by correctness leaves the incorrect set at zero and turns the correct set into a ranking, placing gradient exactly where outcome supervision has none.* Specifically, against the own cold start, Outcome-RL gains $+11.91$ on V*QA and $+8.51$ on VisualProbe Easy but only $+1.68$ and $+1.89$ on Medium and Hard, the splits demanding the longest search. HaPRL inverts that profile, adding **+71.8%** on VisualProbe Hard on average and peaking at **+144.4%** on Qwen3-VL-4B (20.75 vs. 8.49) and **+127.7%** on LLaVA-OneVision-1.5-8B (27.92 vs. 12.26). *That is, an outcome reward can only sharpen a decision the answer prior already informs. A process reward orders the correct rollouts among themselves, and that ordering is where search discipline lives.*

4.2 SEARCH QUALITY UNDER DIFFERENT SETTINGS

Higher accuracy does not imply better search. Table 2 scores answer-correct rollouts only, and Outcome-RL moves search quality by $+0.08$ points over its cold start (82.08 vs. 81.99) while falling on two of six benchmarks. HaPRL adds **+2.78** (84.77) and improves all six.

Table 1: Accuracy comparison of generalist VLMs, SFT, +outcome-based RL, and +our method (HaPRL) on VisualProbe Easy/Medium/Hard (Lai et al., 2025), V*QA (Wu & Xie, 2024), HR-4K/8K (Wang et al., 2025c). All RL post-training methods are trained for three epochs. Avg. is the unweighted mean over the six sets, and the parenthesized value is the HaPRL margin over the matched Outcome-RL control. Best per column of the same backbone in **bold**.

	VisualProbe			High-resolution QA				
Model	Easy	Medium	Hard	V*QA	HR-4K	HR-8K	Avg.	
Qwen3-VL-4B-Instruct								
Base	36.88	16.42	13.21	67.16	55.50	51.25	40.07	
+SFT	27.66	14.55	6.60	46.07	48.88	44.38	31.36	
+SFT+Outcome-RL	29.08	17.91	8.49	58.64	47.13	42.63	33.98	
+SFT+HaPRL	**41.13**	**26.49**	**20.75**	**68.06**	**56.50**	**52.63**	**44.26**	(+10.28)
Qwen3-VL-8B-Instruct								
Base	34.04	18.66	13.21	71.59	64.25	56.38	43.02	
+SFT	26.95	21.64	19.81	45.03	54.75	45.50	35.61	
+SFT+Outcome-RL	34.04	27.61	15.09	56.54	70.25	62.25	44.30	
+SFT+HaPRL	**36.88**	**36.57**	**20.09**	**72.73**	**70.38**	**65.88**	**50.42**	(+6.12)
LLaVA-OneVision-1.5-4B								
Base	34.75	11.94	13.21	60.21	62.75	58.13	40.17	
+SFT	26.95	18.66	9.43	47.12	64.13	56.75	37.17	
+SFT+Outcome-RL	41.13	18.66	17.92	60.21	63.50	56.25	42.95	
+SFT+HaPRL	**43.97**	**20.52**	**23.58**	**61.26**	**64.75**	**58.50**	**45.43**	(+2.48)
LLaVA-OneVision-1.5-8B								
Base	41.13	11.19	16.04	64.40	62.63	57.00	42.07	
+SFT	31.21	18.66	10.38	53.40	60.38	59.25	38.88	
+SFT+Outcome-RL	42.55	16.04	12.26	63.87	60.13	53.88	41.46	
+SFT+HaPRL	**50.52**	**18.28**	**27.92**	**65.97**	**64.38**	**59.75**	**47.80**	(+6.34)

Table 2: The process score (%) on answer-correct rollouts of different methods on Qwen3-VL-4B; Δ against the cold start. The best results are in **bold**. Outcome-RL raises answer accuracy while leaving some search quality flat, whereas HaPRL maintains a positive gain on every split.

	VisualProbe			High-resolution QA			
Model	Easy	Medium	Hard	V*QA	HR-4K	HR-8K	Avg.
+SFT	84.90	86.05	71.54	80.34	86.49	82.64	81.99
+SFT+Outcome-RL	83.55	86.92	65.56	83.22	87.24	85.96	82.08
Δ	−1.35	+0.87	−5.98	+2.88	+0.75	+3.32	+0.08
+SFT+HaPRL	**86.71**	**87.27**	**75.28**	**83.24**	**88.66**	**87.47**	**84.77**
Δ	+1.81	+1.22	+3.74	+2.90	+2.17	+4.83	+2.78

VisualProbe Hard exposes the failure directly. Outcome-RL lifts accuracy there by $+28.6\%$ (8.49 vs. 6.60) while its process score drops 8.4% (65.56 vs. 71.54): the policy answers more hard questions correctly and searches worse doing it. HaPRL recovers **+14.8%** over that control (75.28 vs. 65.56). An outcome reward is not neutral toward search; it erodes it. Because *it cannot tell an answer earned by search from one reached by random guessing, accuracy on the hardest split rises while search quality falls.* This is *reward hacking* whose signature is invisible to the metric the field reports.

4.3 Scaling Behavior

Accuracy scales with annotation volume. Accuracy rises at every budget in Table 3 and Figure 3(a), from 36.62 at 200 annotated prompts to 44.26 at 1,104 (**+20.9%**), with no flattening at the largest budget we collected. Annotating 18% of the prompts already beats Outcome-RL trained on all of them (36.62 vs. 33.98). Returns concentrate on VisualProbe Medium (**+12.31**) and V*QA (**+10.99**) and thin out on VisualProbe Hard (**+3.77**) and HR-Bench 8K (**+3.75**). *Thus, process annotation is meaningful and scales the accuracy.*

Table 3: Accuracy of three-epoch training at different data sizes on Qwen3-VL-4B. Accuracy rises monotonically with the number of process-annotated prompts. The best is **bold**.

	VisualProbe			High-resolution QA			
Annotated prompts	Easy	Medium	Hard	V*QA	HR-4K	HR-8K	Avg.
200	31.21	14.18	16.98	57.07	51.38	48.88	36.62
400	32.70	15.19	17.92	56.97	52.88	48.12	37.30
600	35.46	17.16	18.87	60.40	54.25	49.13	39.21
800	37.99	20.19	19.21	63.40	55.88	50.94	41.27
1000	40.15	24.34	20.55	65.69	56.12	51.68	43.09
1104 (full)	**41.13**	**26.49**	**20.75**	**68.06**	**56.50**	**52.63**	**44.26**
Δ (200 $\rightarrow$ 1104)	+9.92	+12.31	+3.77	+10.99	+5.12	+3.75	+7.64

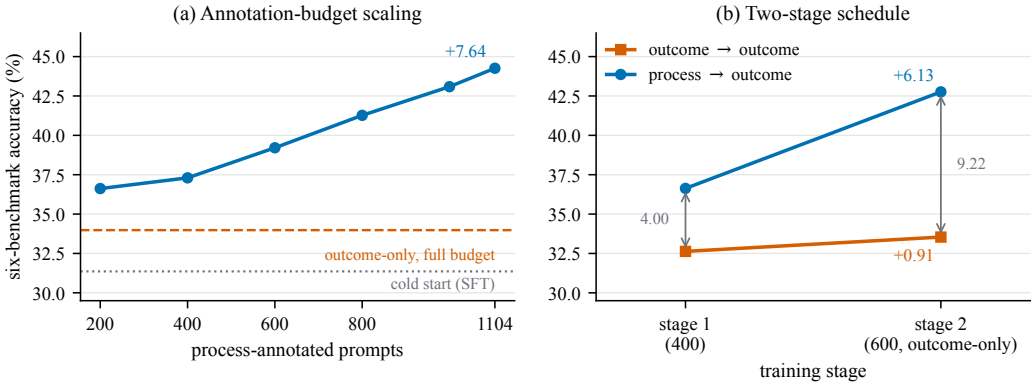


Figure 3: HaPRL **scales with the amount of process-annotated data**, and **injecting human-aligned process data in the early stage enables stronger scaling** even when later training uses only the outcome signal. (a) Accuracy grows monotonically with the process-annotation budget. 1/5 annotated data already beat Outcome-RL trained on all of the them. (b) A second, outcome-only stage returns +0.91 points to an Outcome-RL-trained policy and +6.13 points to a HaPRL-trained one (**6.7×**).

Early process supervision scales better. Table 4 and Figure 3(b) hold stage 2 identical across two arms and vary only stage 1. The same 600-prompt outcome-only continuation returns $+2.8\%$ to the Outcome-RL-trained policy (33.54 vs. 32.63) and $+16.7\%$ to the HaPRL-trained one (42.76 vs. 36.63), a $6.7\times$ difference produced by the initialization alone. The gap between arms widens from $+4.00$ to $+9.22$ points, uniformly across all six benchmarks ($+7.73$ to $+11.06$).

That is, *process data can first establish correct search behavior, and a later outcome signal amplifies what is already working.* Outcome supervision used throughout has no such foundation to build on. It accepts rollouts that reach the right answer through a faulty search, and a policy trained on those trajectories has little real signal left to learn from.

4.4 Ablation Study

We further ablate the human reference trace in three ways on Qwen3-VL-4B, as shown in Table 5.

Shuffled pairing. Re-pairing each trace with another question's trajectory preserves the label distribution and destroys only the alignment, at a cost of 35.3% (28.63 vs. 44.26). Falling 5.35 points below Outcome-RL, and 2.73 below the cold start, is the informative part. A broken regularizer would decay toward the unregularized baseline, whereas a mis-paired trace pushes the policy toward search that was correct for a different image.

Judge without the human trace. Dropping the human-annotated reference trace and letting the frozen judge score the same five-dimensional rubric on its own gives 36.41, only $+2.43$ over Outcome-RL. Restoring the trace adds **+7.85** points ($+21.6\%$, 44.26 vs. 36.41), which is 76% of the

Table 4: Accuracy of one-epoch training on Qwen3-VL-4B. Stage 1 trains on 400 data under either reward (outcome-based vs. HaPRL); stage 2 continues under the same outcome reward on another 600 data. The best is **bold**.

	VisualProbe			High-resolution QA			
Training schedule	Easy	Medium	Hard	V*QA	HR-4K	HR-8K	Avg.
Stage 1 only (400 prompts)							
Outcome-RL	28.35	14.94	11.21	54.84	46.12	40.30	32.63
HaPRL	32.70	15.19	17.92	56.97	52.88	44.12	36.63
Δ	+4.35	+0.25	+6.71	+2.13	+6.76	+3.82	+4.00
Stage 1 → outcome-only stage 2 (600 prompts)							
Outcome-RL → Outcome-RL	28.68	16.85	9.34	57.84	47.03	41.51	33.54
HaPRL → Outcome-RL	**39.74**	**24.58**	**19.36**	**66.42**	**55.21**	**51.27**	**42.76**
Δ	+11.06	+7.73	+10.02	+8.58	+8.18	+9.76	+9.22

Table 5: Ablation study of different training variants. Six-benchmark average accuracy of three-epoch training on Qwen3-VL-4B; Δ against full HaPRL. Shuffled re-pairs each reference trace with another question's trajectory, keeping the label distribution intact. Judge-only removes the trace from the judge prompt and leaves the rubric and task weighting intact. Noise perturbs 10% of the recorded dwell and hover events.

Variant	Human trace	Correct pairing	Avg.	Δ
Shuffled process labels	✓		28.63	−15.63
Judge-only process			36.41	−7.85
+ 10% trace noise	✓	✓	41.74	−2.52
HaPRL (full)	✓	✓	**44.26**	–

full +10.28 margin. The rubric supplies the axes of judgement; the trace supplies where good search sits on those axes for this particular image. A judge denied the trace has to invent that reference from the question alone with its own knowledge. *Thus, the human annotated data is important and carries the learning signal.*

Robustness to annotation noise. Perturbing 10% of the recorded dwell and hover events costs 5.7% (41.74 vs. 44.26) and still leaves +7.76 over Outcome-RL. What the reward takes from a trace is the order in which regions were worth visiting, and that ordering survives jitter in individual events.

Therefore, *the process reward is a channel for human search behavior rather than a hand-designed prior. Its value tracks how faithfully a trace is paired with its trajectory.*

5 Conclusion and Limitation

Outcome-based reinforcement learning trains visual-search agents by scoring only the final answer, so a correct result reached through an incorrect search receives full credit and later optimization can scale along the wrong paths. Thus, *we introduce HaPRL, the first framework that reinforces the search process with human search behavior.* We collect 1,104 human traces with fine-grained interaction signals and distill them into round-by-round descriptions of inspected regions and discovered evidence. A frozen judge then scores each rollout under task-adaptive weights, anchored on the matched human trace, and gates that score by answer correctness. Across all benchmarks, HaPRL consistently outperforms matched outcome-based RL in both accuracy and search quality, and exhibits favorable scaling trends. These results have important implications for the training of foundation models and for expansion to more other domains.

Limitation & Future Work. HaPRL offers a new path and training paradigm for future research. *A natural next step is to extend this human-process alignment recipe to more domains*, especially those in which human annotation signals can play a central role, such as 3D rigging, GUI agents, etc..

AI USE STATEMENT

In this work, we used generative AI tools to polish manuscript wording after an author-written draft, and to assist with some training/inference codes. All AI-assisted text and code were manually verified by the authors. We take responsibility for the final content of this work.

ETHICS STATEMENT

This study strictly adheres to the ICLR Code of Ethics. The datasets utilized in our experiments are publicly available, fully anonymized, and do not involve human subjects, privacy infringement, or harmful discrimination concerns.

REPRODUCIBILITY STATEMENT

To ensure reproducibility, we provide full implementation details, hyperparameter configurations, and training scripts in the anonymized code repository submitted as supplementary material.

REFERENCES

Xiang An, Yin Xie, Kaicheng Yang, Wenkang Zhang, Xiuwei Zhao, Zheng Cheng, Yirui Wang, Songcen Xu, Changrui Chen, Chunsheng Wu, Huajie Tan, Chunyuan Li, Jing Yang, Jie Yu, Xiyao Wang, Bin Qin, Yumeng Wang, Zizhen Yan, Ziyong Feng, Ziwei Liu, Bo Li, and Jiankang Deng. LLaVA-OneVision-1.5: Fully open framework for democratized multimodal training. *arXiv preprint arXiv:2509.23661*, 2025.

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.

DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Lingxiao Du, Fanqing Meng, Zongkai Liu, Zhixiang Zhou, Ping Luo, Qiaosheng Zhang, and Wenqi Shao. MM-PRM: Enhancing multimodal mathematical reasoning with scalable step-level supervision. *arXiv preprint arXiv:2505.13427*, 2025.

Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Yunzhong He, Bing Liu, and Sean Hendryx. Rubrics as rewards: Reinforcement learning beyond verifiable domains. *arXiv preprint arXiv:2507.17746*, 2025.

Yushi Hu, Weijia Shi, Xingyu Fu, Dan Roth, Mari Ostendorf, Luke Zettlemoyer, Noah A. Smith, and Ranjay Krishna. Visual sketchpad: Sketching as a visual chain of thought for multimodal language models. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Zenan Huang, Yihong Zhuang, Guoshan Lu, Zeyu Qin, Haokai Xu, Tianyu Zhao, Ru Peng, Jiaqi Hu, Zhanming Shen, Xiaomeng Hu, Xijun Gu, Peiyi Tu, Jiaxin Liu, Wenyu Chen, Yuzhuo Fu, Zhiting Fan, Yanmei Gu, Yuanyuan Wang, Zhengkai Yang, Jianguo Li, and Junbo Zhao. Reinforcement learning with rubric anchors. *arXiv preprint arXiv:2508.12790*, 2025.

Laurent Itti and Christof Koch. Computational modelling of visual attention. *Nature Reviews Neuroscience*, 2(3):194–203, 2001.

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 Symposium on Operating Systems Principles (SOSP)*, 2023.

Xin Lai, Junyi Li, Wei Li, Tao Liu, Tianjian Li, and Hengshuang Zhao. Mini-o3: Scaling up reasoning patterns and interaction turns for visual search. *arXiv preprint arXiv:2509.07969*, 2025.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, et al. Tülu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*, 2024.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations (ICLR)*, 2024.

Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi Wang. Visual agentic reinforcement fine-tuning. *arXiv preprint arXiv:2505.14246*, 2025.

Liangchen Luo, Yinxiao Liu, Rosanne Liu, Samrat Phatale, Meiqi Guo, Harsh Lara, Yunxuan Li, Lei Shu, Yun Zhu, Lei Meng, Jiao Sun, and Abhinav Rastogi. Improve mathematical reasoning in language models by automated process supervision. *arXiv preprint arXiv:2406.06592*, 2024.

Ruilin Luo, Zhuofan Zheng, Yifan Wang, Xinzhe Ni, Zicheng Lin, Songtao Jiang, Yiyao Yu, Chufan Shi, Lei Wang, Ruihang Chu, Jin Zeng, and Yujiu Yang. Unlocking multimodal mathematical reasoning via process reward model. *arXiv preprint arXiv:2501.04686*, 2025.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. ZeRO: Memory optimizations toward training trillion parameter models. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC)*, 2020.

Hao Shao, Shengju Qian, Han Xiao, Guanglu Song, Zhuofan Zong, Letian Wang, Yu Liu, and Hongsheng Li. Visual CoT: Advancing multi-modal language models with a comprehensive dataset and benchmark for chain-of-thought reasoning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024a.

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*, 2024b.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. HybridFlow: A flexible and efficient RLHF framework. In *Proceedings of the Twentieth European Conference on Computer Systems (EuroSys)*, 2025.

Anne M. Treisman and Garry Gelade. A feature-integration theory of attention. *Cognitive Psychology*, 12(1):97–136, 1980.

Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process- and outcome-based feedback. *arXiv preprint arXiv:2211.14275*, 2022.

Vijay Viswanathan, Yanchao Sun, Shuang Ma, Xiang Kong, Meng Cao, Graham Neubig, and Tongshuang Wu. Checklists are better than reward models for aligning language models. *arXiv preprint arXiv:2507.18624*, 2025.

Haozhe Wang, Alex Su, Weiming Ren, Fangzhen Lin, and Wenhu Chen. Pixel reasoner: Incentivizing pixel-space reasoning with curiosity-driven reinforcement learning. *arXiv preprint arXiv:2505.15966*, 2025a.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce LLMs step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 9426–9439, 2024.

Weiyun Wang, Zhangwei Gao, Lianjie Chen, Zhe Chen, Jinguo Zhu, Xiangyu Zhao, Yangzhou Liu, Yue Cao, Shenglong Ye, Xizhou Zhu, Lewei Lu, Haodong Duan, Yu Qiao, Jifeng Dai, and Wenhai Wang. VisualPRM: An effective process reward model for multimodal reasoning. *arXiv preprint arXiv:2503.10291*, 2025b.

Wenbin Wang, Liang Ding, Minyan Zeng, Xiabin Zhou, Li Shen, Yong Luo, Wei Yu, and Dacheng Tao. Divide, conquer and combine: A training-free framework for high-resolution image perception in multimodal large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 7907–7915, 2025c.

Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal LLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13084–13094, 2024.

Xintong Zhang, Zhi Gao, Bofei Zhang, Pengxiang Li, Xiaowen Zhang, Yang Liu, Tao Yuan, Yuwei Wu, Yunde Jia, Song-Chun Zhu, and Qing Li. Adaptive chain-of-focus reasoning via dynamic visual search and zooming for efficient VLMs. *arXiv preprint arXiv:2505.15436*, 2025.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-Bench and chatbot arena. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2023.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. LlamaFactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (System Demonstrations)*, 2024.

Ziwei Zheng, Michael Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, Chao Shen, and Xing Yu. DeepEyes: Incentivizing "thinking with images" via reinforcement learning. *arXiv preprint arXiv:2505.14362*, 2025.

A Additional Method Details

A.1 Algorithms

Algorithm 1 constructs the human process traces used as reward anchors. Algorithm 2 is the subsequent training loop: sampled trajectories are scored by a frozen answer judge, process credit is assigned only when the answer is correct, and GRPO updates the policy from group-relative advantages. Human traces and both judges are used only in this training loop; inference retains the policy and the crop tool.

Algorithm 1 Human process data construction

Require: VisualProbe training questions $\{(I_0, q, a^*)\}$
Ensure: Dataset $\mathcal{D} = \{(I_0, q, a^*, E^*, H^*)\}$
$\mathcal{D} \leftarrow \emptyset$
for each question (I_0, q, a^*) **do**
  Annotator searches I_0 and marks the region containing the supporting evidence
  Log the interaction stream $E^* = (e_1, \ldots, e_M)$, $e_m = (\kappa_m, t_m, \mathbf{p}_m, v_m, \alpha_m, b_m)$
  Distill E^* into a round-by-round trace $H^* = (r_1^*, \ldots, r_L^*)$
  Suppress pointer samples that do not change the inspected region
  Retain dwell, redirection, and coarse-to-fine refinement
  $\mathcal{D} \leftarrow \mathcal{D} \cup \{(I_0, q, a^*, E^*, H^*)\}$
end for
return $\mathcal{D}$ ▷ the policy observes only (I_0, q)

Algorithm 2 HaPRL training with answer-gated process reward

Require: Policy π_θ, frozen judges J_{ans} and J_ϕ, dataset $\mathcal{D}$, group size G
for each training prompt $x = (I_0, q)$ with matched trace H^* and answer a^* **do**
  Sample G trajectories $\{\tau_i\}_{i=1}^G$ from π_θ with the crop tool
  for $i = 1$ to G **do**
    $r_{\text{out}}(\tau_i) \leftarrow J_{\text{ans}}(q, a(\tau_i), a^*)$
    if $r_{\text{out}}(\tau_i) = 0$ **then**
      $R_i \leftarrow 0$
    else
      $\mathbf{s}(\tau_i), p \leftarrow J_\phi(I_0, q, \tau_i, H^*)$
      Clip $\mathbf{s}(\tau_i)$ to $[0, 1]$ and normalize $\mathbf{w}^{(p)}$
      $r_{\text{proc}}(\tau_i) \leftarrow \langle \mathbf{w}^{(p)}, \mathbf{s}(\tau_i) \rangle$
      $R_i \leftarrow r_{\text{out}}(\tau_i)\, r_{\text{proc}}(\tau_i)$
    end if
  end for
  Compute group-relative advantages A_i from $\{R_i\}_{i=1}^G$
  Update θ with clipped GRPO and KL regularization toward π_{ref}
end for

A.2 Human-Trace Representation

For each annotated example, the raw event stream contains timestamps, cursor positions, hover and zoom events, selected regions, and the final response. We distill it offline into an ordered textual trace. Each retained round states the region under inspection, the normalized crop when available, and the evidence revealed by that view. Low-level pointer motion that does not change the inspected region is suppressed. This produces a compact semantic anchor that can be compared with the model rollout without requiring exact action or token alignment.

The process judge receives the original image, question, serialized model rollout, the matched human trace, and compact annotation metadata. The trace is identified by the same sample key as the rollout; it is never re-paired across examples except in the shuffled-reference ablation.

A.3 Structured Process Rubric

The judge returns one score in $[0, 1]$ for each of five dimensions. Target semantics measures whether the search follows the queried object, attribute, text, count, or relation. Evidence acquisition measures whether the observed views actually support the answer. Search progress rewards information gain and useful recovery while penalizing drift and repeated crops. Tool discipline covers valid source views, bounding boxes, and crop granularity. Communication discipline measures whether the reasoning is concise, consistent, and tied to visible evidence.

Table 6: **Task-adaptive weights over the five process dimensions.** Columns denote target semantics (Tar.), evidence acquisition (Evd.), search progress (Prog.), tool discipline (Tool), and communication discipline (Com.).

Profile	Tar.	Evd.	Prog.	Tool	Com.
General search	.20	.30	.25	.15	.10
OCR / text	.15	.40	.20	.15	.10
Count / relation	.20	.35	.25	.10	.10
Attribute	.25	.30	.20	.15	.10

The judge predicts the task profile from the question and annotations. Scores are clipped to $[0, 1]$, profile weights are normalized to sum to one, and the weighted total is recomputed outside the model response. This prevents a malformed or internally inconsistent judge output from directly setting the reward.

A.4 Reward Computation

For every sampled trajectory, training applies the following sequence, also listed in Algorithm 2:

1. Extract the terminal answer and obtain the binary semantic-match score r_{out} from the answer judge.
2. If $r_{\text{out}} = 0$, set $R_{\text{HaPRL}} = 0$ and skip process evaluation.
3. Otherwise, evaluate the rollout against its matched human trace, select the task profile, and compute r_{proc} from the structured rubric.
4. Set $R_{\text{HaPRL}} = r_{\text{out}} r_{\text{proc}}$, standardize rewards within the rollout group, and update the policy with GRPO.

This ordering makes the gate semantic rather than cosmetic: fluent reasoning, valid tool syntax, or close imitation of the human trace cannot earn reward when the final answer is wrong. Conversely, answer-correct rollouts retain a continuous preference signal that outcome-only supervision discards.

B Experimental Details

B.1 Benchmarks

VisualProbe (Lai et al., 2025) splits questions by search difficulty, defined by target size and distractor density. V*Bench QA (Wu & Xie, 2024) scores the correct option text as an open-ended answer rather than as multiple choice, which removes the option-elimination shortcut. HR-Bench (Wang et al., 2025c) pairs the same questions at 4K and 8K resolution, so the 8K split isolates the cost of searching a larger frame.

Visual token budget. Every rollout, in training and in evaluation, may take at most six tool rounds and hold at most six images in context. Each image is rescaled into the range 4×10^4 to 10^6 pixels. The budget is what makes the crop tool a real decision: magnifying a region costs one of six slots, so an agent that wastes rounds has fewer left for the target. All three agent conditions receive the identical budget, so no reported gap between them reflects one model seeing more pixels.

Metrics details. Answer accuracy is a binary semantic match produced by the answer judge, which compares the string inside `<answer>` against the reference and accepts paraphrases and formatting differences. VisualProbe is natively short-answer and HR-Bench natively multiple choice. V*Bench

is also multiple choice, but we score its correct option text as a free-form answer, which stops the policy from recovering the answer by eliminating distractors instead of searching.

B.2 Cold Start

Two filters reduce the 7,267-trajectory Mini-o3 cold-start corpus (Lai et al., 2025). The first keeps trajectories with at most six assistant turns (6,757 left), matching the tool-round cap used in RL. The second drops trajectories dominated by recovery phrases or repeated long sentences (6,057 left), both of which survive supervised training and reappear as degenerate loops during rollout.

Fine-tuning updates the language model with the vision tower and multimodal projector frozen. We use learning rate 5×10^{-6} for five epochs, sequence cutoff 32768, and per-device batch 1 with gradient accumulation 4, under DeepSpeed ZeRO-3 offload (Rajbhandari et al., 2020) in LLaMA-Factory (Zheng et al., 2024). Images use the same pixel range as RL.

Cold-start loss is a poor predictor of multi-turn health: checkpoints with lower loss frequently produce repeated crops and recovery loops. We therefore select the RL initialization by rollout quality on 20 held-out prompts, scoring answer rate, valid grounding rate, overlong rate, and mean response length.

B.3 Reinforcement Learning

Table 7 gives the full RL configuration; the two conditions differ only in the optimization total. Both call the answer judge and the process judge and log all five rubric dimensions. Process quality therefore stays observable in the outcome-only arm, where it contributes nothing to the gradient.

Table 7: RL configuration. The last row is the only difference between the two conditions.

Setting	Value
Algorithm	GRPO (Shao et al., 2024b)
Framework	verl (Sheng et al., 2025), vLLM rollout (Kwon et al., 2023)
Rollouts per prompt	8
Prompt batch size	48
PPO mini / micro batch	8 / 1 per GPU
Actor learning rate	5×10^{-7}
KL loss coefficient	3×10^{-3} (low-variance estimator)
Rollout temperature / top-p	1.0 / 1.0
Max prompt / response tokens	8192 / 8192
Max tool rounds	6
Max images per context	6
Image pixel range	4×10^4 to 10^6
Hardware	8 × H100 or 8 × H20

B.4 Judge

Both judges use a frozen Qwen3-VL-30B-A3B-Instruct served behind an OpenAI-compatible endpoint. The process judge is called at temperature 0 with structured JSON output. It receives the original image, the full multi-turn rollout rendered as text, and the human reference trace for the same question. Three call outcomes are logged: `vision` when the image is accepted, `text_fallback` when the image call fails and the judge is retried on text alone, and `failed` after three attempts. We serialize judge-heavy jobs and halt training when the fallback rate grows, since a degraded judge silently flattens the process reward.

The answer judge calls the same model through a separate text-only prompt and returns a binary semantic-correctness score. It is shared by both training conditions and all evaluations.

B.5 Process Annotation

Annotators answered 1,104 VisualProbe training questions in a browser interface over the full-resolution image. The interface logs a timestamped event stream, with event types covering canvas entry, hover movement, zoom start, and zoom motion, together with the cursor position and the region

under inspection. Annotators also record the bounding boxes they settled on as the evidence for their answer.

Raw event streams never reach the judge. Each stream is distilled into a short per-round textual trace naming the region the annotator inspected and what they found there. That distilled trace is the human reference the judge reads. Annotation is collected once per question and reused across every run in the paper, and it never enters the policy's context at training or inference time.

Annotation cost. A single annotator produced the whole collection. One question takes about 1.5 minutes on average, so the $1{,}104$ questions amount to roughly 27.6 person-hours. Because each trace is collected once and reused by every run reported in this paper, this cost is paid once for the entire study rather than per training run or per backbone. Appendix C.1 weighs it against cheaper reward references, including one that needs only the final evidence box and takes about ten seconds per question.

Ablation construction. The shuffled-label variant permutes the mapping from questions to reference traces, so each rollout is judged against a trace collected for a different question. The label distribution is unchanged and only the correspondence is destroyed. The noise variant perturbs 10% of the recorded dwell and hover events in each stream before distillation. The judge-only variant removes the reference trace from the judge prompt and leaves the rest of it, including the rubric and the task-profile weighting, intact.

C Additional Experiments

Three experiments separate the contribution of the human trace from three things that could be mistaken for it, i.e., (i) the contribution of any evidence specification, (ii) the contribution of a better cold start, and (iii) seed variance.

C.1 Cheaper Reward References

Anchoring the process reward to an ordered human trace costs 27.6 person-hours (Appendix B.5). Three cheaper references can occupy the same slot in the reward. A *hand-designed tool-use bonus* pays a fixed amount for every crop that is not a near-duplicate of an earlier one, and needs neither annotation nor a judge call. A *final evidence box* scores a rollout by how much of the box the annotator settled on its crops covered. A *teacher-generated evidence path* asks a strong vision-language model, shown the ground-truth answer, to write the region sequence it would have inspected, and judge reads that path exactly as it reads a human one.

Table 8 holds training identical and varies only the reference. Every substitute improves on the outcome-only reward, so part of the gain follows from anchoring the reward to an evidence specification of any kind. The substitutes then separate by how much of the search each one describes. The tool-use bonus, which describes none of it, recovers 1.74 points. The final box, which names where the evidence is, recovers 4.63 points. The teacher path, which names an ordering as well, reaches 39.85 with no human labour at all. The human trace adds a further 4.41 points over the best substitute.

What separates the human trace from the teacher path is where the ordering comes from. *A path written backwards from a known answer contains no rejected hypotheses*, so a rollout that inspects two plausible regions before converging is scored against a reference that went straight to the target. A human reference contains the same dead ends, and the judge can therefore distinguish productive exploration from redundant revisiting.

C.2 Cold-Start Dependence

Every backbone in Table 1 loses accuracy after supervised fine-tuning, which leaves open whether HaPRL improves visual search training or repairs that loss. We separate the two with a second cold start that does not degrade the backbone. Recipe B keeps the unfiltered $7{,}267$-trajectory corpus, masks the loss on turns beyond the six-round cap instead of discarding those trajectories, and stops

Table 8: Reward reference and what it specifies about the search. All arms train Qwen3-VL-4B for three epochs on the same 1,104 prompts and differ only in what the process reward is anchored to. Avg. is the macro average over the six evaluation splits. The best is **bold**.

Reward reference	Specifies	Judge	Avg.	Δ vs. HaPRL
None (Outcome-RL)	—	—	33.98	−10.28
Hand-designed tool-use bonus	crop novelty	—	35.72	−8.54
Rubric only, no reference	rubric dimensions	✓	36.41	−7.85
Final evidence box, coverage reward	location	—	38.61	−5.65
Teacher-generated evidence path	location, ordering	✓	39.85	−4.41
Human ordered trace (HaPRL)	location, ordering, dead ends	✓	**44.26**	—

Table 9: Cold-start recipe crossed with reward condition on Qwen3-VL-4B. Recipe A is the cold start used throughout the main text; Recipe B keeps the unfiltered corpus, masks the loss beyond the six-round cap, and stops at two epochs. The backbone scores 40.07 before any agent training. Entries are macro averages over the six evaluation splits.

Cold-start recipe	SFT	+Outcome-RL	+HaPRL	Δ (HaPRL − Outcome-RL)
A: filtered corpus, five epochs (ours)	31.36	33.98	44.26	+10.28
B: full corpus, over-turn masking, two epochs	39.84	43.15	**49.02**	+5.87

at two epochs rather than five. On Qwen3-VL-4B it lands at 39.84, within 0.23 of the 40.07 the backbone reaches before any agent training.

Table 9 runs both reward conditions from each cold start. The advantage of HaPRL over the outcome-only reward is 10.28 points from Recipe A and 5.87 points from Recipe B. Recovery from a weak initialization therefore accounts for 4.41 of the 10.28 points in the main table, and process supervision for the remaining 5.87, measured from an initialization with nothing to recover. The two effects add rather than overlap: Recipe B with HaPRL reaches 49.02, the highest score in this study and 4.76 points above the best Recipe A result.

C.3 Two-Stage Seed Variance

Table 4 reports one run per schedule, in which an outcome-only stage 2 adds 6.13 points after a process-trained stage 1 and 0.91 points after an outcome-only one. Table 10 repeats both schedules three times.

Across seeds the outcome-only path gains 0.88 ± 0.06 points in stage 2 and the process-initialized path gains 5.83 ± 0.26, a ratio of $6.6\times$. The largest spread in any row is 0.57 points, an order of magnitude below the 4.95-point difference between the two gains. Both schedules see the same 600 prompts under the same reward in stage 2 and differ only in their stage-1 initialization, and the 3.94-point gap they carry into stage 2 widens to 8.89 by the end of it. The three seeds share the prompt ordering and the 400/600 split, so these intervals cover optimization and rollout stochasticity rather than data partitioning.

D Qualitative Cases

The tables in Section 4 report what changed; the six cases below show what the change looks like inside a rollout. Each figure holds the question fixed and renders the condensed trajectory of all three trained conditions from the same backbone: the cold start, Outcome-RL, and HaPRL. Every round is shown with the reasoning that motivated it, the emitted crop, and the observation it returned, so the route to the answer can be read directly rather than inferred from a score.

These are six hand-picked examples chosen to span the task profiles of Section 3.2 (text recognition, spatial relation, and attribute recognition) and all three evaluation sources. They illustrate mechanisms already measured in aggregate and are not themselves evidence of frequency. Table 11 summarizes them.

Table 10: Three seeds of the two-stage schedule of Table 4 on Qwen3-VL-4B. Seed 1 is the run reported in the main text. Entries are macro averages over the six evaluation splits; the last column is the stage-2 gain of each row over its own stage-1 checkpoint. Mean ± SD over three seeds.

Training schedule	Seed 1	Seed 2	Seed 3	Mean ± SD	Stage-2 gain
Stage 1 only (400 prompts)					
Outcome-RL	32.63	32.05	32.55	32.41 ± 0.31	—
HaPRL	36.63	35.94	36.48	36.35 ± 0.36	—
Stage 1 → outcome-only stage 2 (600 prompts)					
Outcome-RL → Outcome-RL	33.54	32.86	33.47	33.29 ± 0.37	$+0.88 \pm 0.06$
HaPRL → Outcome-RL	42.76	41.62	42.16	$\mathbf{42.18 \pm 0.57}$	$+5.83 \pm 0.26$

Table 11: The six qualitative cases. Rounds and crops are counted from the rendered trajectories in Figures 4–9. In every case HaPRL reaches the correct answer with the shortest evidence path of the three conditions.

			Rounds / crops and verdict		
Fig.	Task profile	Source	+SFT	+Outcome-RL	+HaPRL
4	OCR / text	HR-Bench 4K-500	3 / 2 ×	6 / 5 ✓	4 / 3 ✓
5	OCR / text	HR-Bench 8K-39	4 / 3 ×	7 / 6 ×	3 / 2 ✓
6	OCR / text	VisualProbe Easy-128	4 / 3 ×	7 / 6 ×	2 / 1 ✓
7	OCR / relation	VisualProbe Hard-54	7 / 6 ×	6 / 5 ✓	2 / 1 ✓
8	Attribute	VisualProbe Medium-151	7 / 6 ×	7 / 6 ×	3 / 2 ✓
9	Attribute	V^* attributes-58	4 / 3 ×	4 / 3 ×	2 / 1 ✓
Mean rounds / crops			4.8 / 3.8	6.2 / 5.2	**2.7 / 1.7**
Correct			0 / 6	2 / 6	**6 / 6**

What the outcome-only rollouts do with their extra rounds. Outcome-RL uses the most rounds and the most crops of the three conditions and is correct twice. Both successes are the kind of rollout the argument of Section 1 predicts an outcome reward cannot penalize. In Figure 4 it overshoots the target twice, crops water and then rock, re-localizes, and arrives at the right word on the fifth crop. In Figure 7 it crops a nearly identical region twice in succession before confirming an answer it had already read. Neither trajectory is efficient and both receive full credit, because the only thing the reward can see is the string at the end. The four failures show where that indifference leads. In Figures 5 and 8 the policy repeats the same bottom-region or shelf crop across three consecutive rounds, exhausts its six-round budget, and emits no valid answer. In Figure 6 it chases several date-like lines and returns the truncated "March 3" instead of "March, 1935". In Figure 9 it zooms to the wrong flagpole twice and reports the colors of a flag that is not the queried one, which the cold start does as well: *repetition and distractor-chasing are exactly the behaviors a rubric can name and an outcome reward cannot.*

What the process-supervised rollouts do instead. HaPRL answers all six correctly using 2.7 rounds and 1.7 crops on average, fewer than either control. The pattern is consistent across profiles. In Figure 6 it crops the article header once and reads the date. In Figure 7 it crops the joint BEER/PARK sign region once, which makes the spatial relation legible in a single view rather than requiring the two separate crops the other conditions attempt. In Figure 8 it locates the shelf containing the queried word and then tightens onto it while keeping the word visible, and in Figure 9 it crops the roadside flag directly instead of the salient flagpole above it. This is what the process-quality numbers of Table 2 measure: not that the answer is right, *but that the view which justifies it was actually obtained, and obtained without spending rounds that reveal nothing new.*

Case Study: Fine-grained OCR through Visual Search



Question

What word is written on the hang glider?

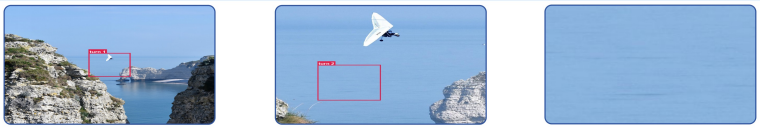
(A) Windy (B) Cloudy (C) Stormy (D) Sunny

Reference answer: (A) Windy

Task: OCR / Text

Sample Source: HR-Bench 4K-500

SFT 3rounds / 2crops — *Incorrect reading* — (D) Sunny — Wrong



Condensed rollout

T1: Centers on the small white glider.

<think> The hang glider is a small white object in the middle of the image. The hang glider is located in the center of the image, slightly to the right. ……I estimate its position to be within the coordinates [0.4, 0.4, 0.6, 0.6]. </think><grounding> {"bbox_2d": [0.4,0.4,0.6,0.6], "source": "original_image"}</grounding>

T2: Zooms the lower-left wing, but the text remains ambiguous.

<think> I have zoomed in on the hang glider in the previous step. The word appears to be written on the lower part of the hang glider's wing. It is still a bit small and blurry to read clearly. I will zoom in further on the specific area where the word is located. ……I estimate the coordinates for this new bounding box to be [0.2, 0.5, 0.5, 0.8] within observation 1. </think> <grounding>{"bbox_2d": [0.2,0.5,0.5,0.8], "source": "observation_1"}</grounding>

T3: Misreads the word as "SUNNY" and answers D.

<think> The user wants to identify the word written on the hang glider. I have zoomed in twice, and now in observation 2, the word is much clearer. I can now read the word on the hang glider's wing. The word is "SUNNY". </think>**<answer> D </answer>**

Outcome-only RL 6rounds / 5crops — *Correct after repeated retry* — (A) Windy — Correct



Condensed rollout

T1: Initial crop is too high.

<think> First, I need to locate the hang glider in the image. I can see a white object in the sky, which appears to be a hang glider, located in the upper-middle part of the image. ……I need to zoom in on the hang glider to read the word. </think><grounding>{"bbox_2d": [0.4,0.2,0.6,0.5], "source": "original_image"}</grounding>

T2: Moves too low and captures an object in the water.

<think> However, looking at observation 1, the hang glider is not in the center of the image. It seems my initial bounding box was too high. ……I need to adjust my bounding box. </think> <grounding>{"bbox_2d": [0.4,0.5,0.7,0.9], "source": "original_image"}</grounding>

T3: Re-localizes above the rock and finds the glider.

<think> Looking at observation 2, I can see a small white object in the water, not the hang glider. This means my bounding box is still incorrect. The hang glider is actually higher up than I thought. ……My previous bounding boxes were too low. I need to aim higher. </think><grounding>{"bbox_2d": [0.3,0.1,0.6,0.5], "source": "original_image"}</grounding>

T4: Zooms the wing and tentatively reads 'Windy'.

<think> Looking at observation 3, I can see a small white object in the sky, which is the hang glider. The text on it is still not clear. I need to zoom in further on the hang glider itself. ……I will select a bounding box that focuses on the hang glider. </think><grounding> {"bbox_2d": [0.4,0.4,1.0,1.0], "source": "observation_3"}</grounding>

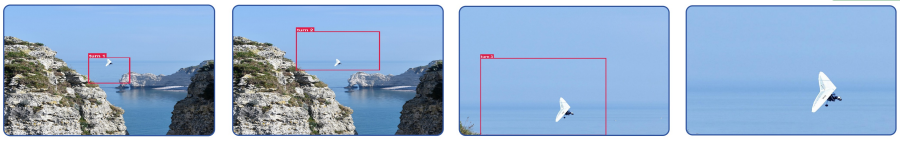
T5: Tight crop confirms Windy.

<think> Looking at observation 4, I can see the hang glider more clearly. There is some text written on the wing. …… Let me zoom in one more time to be sure. The text is located on the left wing of the hang glider. I will select a bounding box that focuses on the text. </think><grounding>{"bbox_2d ": [0.0,0.2,0.8,0.8], "source": "observation_4"}</grounding>

T6: Answer.

<think> Looking at observation 5, I can see the text on the hang glider more clearly. The word is "Windy". ……This matches option (A) in the question. I am confident in this answer and no further grounding is needed. </think><answer> A </answer>

HaPRL 4rounds / 3crops — *Correct with a shorter evidence path* — (A) Windy — Correct



Condensed rollout

T1: Initial center crop captures only part of the wing.

<think> The hang glider is a small white object in the upper-middle part of the image. The text on it is not legible in the original image due to its small size. ……Therefore, I need to zoom in on the hang glider to read the word. </think><grounding>{"bbox_2d": [0.4,0.4,0.6,0.6], "source": "original_image"}</grounding>

T2: Re-centers on the upper-middle region and obtains the full glider.

<think> However, looking at observation 1, the hang glider is more towards the top of the image, not the center. I need to adjust my bounding box based on the original image…….A better bounding box would be centered more on the glider. </think><grounding>{"bbox_2d": [0.3,0.2,0.7,0.5], "source": "original_image"}</grounding>

T3: Targets the wing text, reads 'Windy', and answers A.

<think> Observation 2 shows a much better view of the hang glider. I can now see the entire glider, including the text on its wing. To read it clearly, I need to zoom in further on the wing where the text is located.……I will select a bounding box that focuses on this specific area. </think><grounding>{"bbox_2d": [0.1,0.4,0.7,1.0], "source": "observation_2"}</grounding>

T4: Answer

<think> Observation 3 shows a clear view of the text on the wing. I can now read the word. The word is "Windy". This matches option (A) from the given choices. I am confident in this answer and no further grounding is needed. </think><answer> A </answer>

Figure 4: **Fine-grained OCR through visual search** (HR-Bench 4K-500). The cold start zooms twice, misreads the word on the wing as "SUNNY", and answers the wrong option. Outcome-RL needs five crops, two of which land on water and rock, before reading the word correctly. HaPRL re-centers once on the upper-middle region, targets the wing text, and answers in four rounds.

Case Study: Fine-grained Device-brand OCR



Question

Which brand is mentioned on the black rectangular device located at the left of the image?

(A) Sony (B) Apple (C) Samsung (D) Insignia

Reference answer: (D) Insignia

Task: OCR / Text

Sample Source: HR-Bench 8K-39

SFT 4rounds / 3crops — *Incorrect reading* — (A) Sony — Wrong



Condensed rollout

T1: Localizes a candidate device on the left.

<think> I can see several electronic devices. On the left, there is a black device with some text on it. To read the brand name, I need to zoom in on this specific device. I will select a bounding box that covers the left portion of the image where this device is located. </think><grounding> {"bbox_2d": [0.0, 0.4, 0.4, 0.8], "source": "original_image"}</grounding>

T2: Nested zoom still leaves the brand blurry.

<think> In the previous step, I cropped the left part of the image. Now I need to examine `observation_1` to see if the brand is visible. The cropped image shows a black device, but the text on it is not clear enough to read.....</think> <grounding>{"bbox_2d": [0.2, 0.2, 0.8, 0.8], "source": "observation_1"}</grounding>

T3: Over-zooms the text region.

<think> I have zoomed in twice, and now in `observation_2`, I have a closer view of the device. However, the text is still not legible. I can see some characters, but they are too small and blurry to be identified with certainty. I need to zoom in one more time..... </think><grounding>{"bbox_2d": [0.3, 0.3, 0.7, 0.7], "source": "observation_2"}</grounding>

T4: Misreads the brand as SONY and answers A.

<think> I have zoomed in three times, and now in `observation_3`, I have a very close-up view of the device. The text is still a bit blurry, but I can make out some letters. It appears to be "SONY". </think>**<answer> (A) Sony </answer>**

Outcome-only RL 7rounds / 6crops — *Max-round failure* — Wrong



Condensed rollout

T1: Starts from the left-side device.

<think> I can see several electronic devices in the image. On the left side, there is a black device with some text on it. I need to zoom in on this device to read the brand name. The device is located in the left half of the image..... </think><grounding>{"bbox_2d":[0.0, 0.4, 0.5, 0.8], "source": "original_image"}</grounding>

T2: Pushes toward the bottom of the crop.

<think> I cropped the left half of the image. In observation 1, I can see a black rectangular device at the bottom. It has some text on it. I need to zoom in on this device to read the brand name. The device is located at the bottom of observation 1..... </think> <grounding>{"bbox_2d": [0.2, 0.5, 1.0, 1.0], "source": "observation_1"}</grounding>

T3: Repeats the same bottom-region search.

<think> In the previous step, I cropped the bottom part of observation 1. In observation 2, I can see a black rectangular device at the bottom. It has some text on it. I need to zoom in on this device to read the brand name. .....</think><grounding>{"bbox_2d": [0.0, 0.5, 1.0, 1.0], "source": "observation_2"}</grounding>

T4: Repeats the same bottom-region search again.

<think> I cropped the bottom part of observation 2. In observation 3, I can see a black rectangular device at the bottom. It has some text on it. I need to zoom in on this device to read the brand name. The device is located at the bottom of observation 3.....</think><grounding> {"bbox_2d": [0.0, 0.5, 1.0, 1.0], "source": "observation_3"}</grounding>

T5: Abandons the crop and guesses another device text.

<think> However, in observation 4, I still cannot clearly see the brand name. The brand name seems to be on a different part of the device. Let's go back to observation 1. In observation 1, there is a black device with the brand name "DIN360" written on it.....</think><grounding>{"bbox_2d ": [0.2, 0.2, 0.8, 0.8], "source": "observation_1"}</grounding>

T6: Jumps to an unrelated top-right device.

<think> In the previous step, I cropped the center-left part of observation 1. In observation 5, I can see a black rectangular device with some text on it. The text is still a bit blurry, but it seems to be "DIN360"..... </think><grounding>{"bbox_2d ":[0.5, 0.0, 1.0, 0.6], "source": "original_image"}</grounding>

T7: Uses the final crop budget without a valid answer.

<think> In the previous step, I cropped the top-right region of the original image. In observation 6, I can see a black rectangular device at the top. It has some text on it. I need to zoom in on this device to read the brand name.....</think><grounding>{"bbox_2d ": [0.0, 0.0, 1.0, 0.5], "source": "observation_6"}</grounding>

HaPRL 3rounds / 2crops — *Correct and grounded* — (D) Insignia — Correct



Condensed rollout

T1: Covers the correct left-side device.

<think>I can see several electronic devices in the image. On the left side, there is a black device with some text on it. I need to zoom in on this device to read the brand name. The device is located in the left half of the image. This should capture the device in question.</think><grounding>{"bbox_2d": [0.0, 0.3, 0.5, 0.8], "source": "original_image"}</grounding>

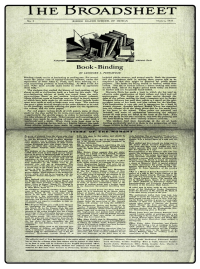
T2: Recognizes the vertical INSIGNIA text and tightens once.

<think> In Observation 1, I can see a black rectangular device on the left. The brand name is written vertically on the device. It seems to be "INSIGNIA". The device is located in the upper left part of Observation 1. I will select a bounding box around this device to get a clearer view of the text. This should isolate the device and make the text readable.</think><grounding>{"bbox_2d": [0.1, 0.1, 0.6, 0.7], "source": "observation_1"}</grounding>

T3: Confirms INSIGNIA and answers D.

<think> In Observation 2, I can clearly see the brand name written vertically on the black rectangular device. The text reads "INSIGNIA". Therefore, I can confidently provide the final answer. </think>**<answer> (D) Insignia </answer>**

Figure 5: **Fine-grained device-brand OCR** (HR-Bench 8K-39). The cold start over-zooms until the text is illegible and guesses "Sony". Outcome-RL repeats the same bottom-region crop three times, abandons it, jumps to an unrelated device, and exhausts its round budget without a valid answer. HaPRL covers the left-side device, recognizes the vertical INSIGNIA text, and answers in three rounds.



Case Study: Historical-date OCR

Question

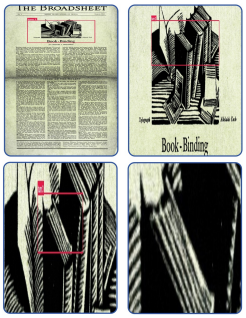
What is the displayed date of the article?

Reference answer: March, 1935

Task: OCR / Text

Sample Source: Visual probe-easy-128

SFT 4rounds / 3crops — *Incorrect reading* — FRIDAY, 20 APRIL 2023 — Wrong



Condensed rollout

T1: Targets the newspaper header.

<think> I can see the title of the newspaper is "THE BROADSHEET". Below the title, there is some text which likely contains the date. I will zoom in on the area just below the main title to read the text clearly. This area is located in the upper part of the image..... </think><grounding> {"bbox_2d": [0.2, 0.1, 0.8, 0.25], "source": "original_image"}</grounding>

T2: Moves to a different date-like line.

<think> I have zoomed in on the area below the title "THE BROADSHEET". In observation 1, I can see some text, but it's still a bit blurry and I can't make out the exact date. I need to zoom in further to read the text clearly. ..... </think> <grounding>{"bbox_2d": [0.2, 0.1, 0.8, 0.4], "source": "observation_1"}</grounding>

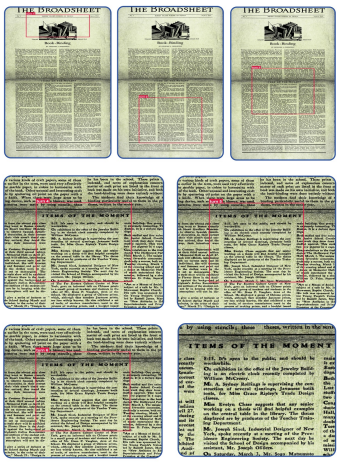
T3: Zooms irrelevant publication text.

<think> I have zoomed in twice, but the text is still not clear enough to read the date. Looking at observation 2, the text is located in the center of the image. I will select a bounding box that focuses on the text. The bounding box will be approximately [0.3, 0.2, 0.7, 0.6] in observation 2. This should give me a very clear view of the date. </think><grounding>{"bbox_2d": [0.3, 0.2, 0.7, 0.6], "source": "observation_2"}</grounding>

T4: Answers FRIDAY, 20 APRIL 2023.

<think> I have zoomed in three times, and in observation 3, I can finally see the text more clearly. The text says "THE BROADSHEET" and below it, there is a line that says "FRIDAY, 20 APRIL 2023". I can now clearly read the date. I am confident in this answer and no further grounding is needed. </think>**<answer> FRIDAY, 20 APRIL 2023 </answer>**

Outcome-only RL 7rounds / 6crops — *Incorrect after repeated search* — March 3 — Wrong



Condensed rollout

T1: Searches the upper newspaper region.

<think> I can see the title of the newspaper is "THE BROADSHEET". Below the title, there is some text which likely contains the date. I need to zoom in on that area to read it clearly. ……</think><grounding>{"bbox_2d": [0.2, 0.05, 0.8, 0.2], "source": "original_image"}</grounding>

T2: Moves across multiple header fragments.

<think> I have zoomed in on the top part of the image, but I cannot see any date in observation 1. The text is still too small and blurry to read. I will examine the area below the illustration of the books. ..... </think> <grounding>{"bbox_2d": [0.2, 0.6, 0.8, 0.9], "source": "original_image"}</grounding>

T3: Chases another date-like line.

<think> I have zoomed in on the lower part of the image, but I still cannot see any clear date in observation 2. The text is too small and blurry. I will examine the area in the middle of the image……</think><grounding>{"bbox_2d": [0.2, 0.4, 0.8, 0.7], "source": "original_image"}</grounding>

T4: Repeats the header search.

<think> I have zoomed in on the middle part of the image in observation 3. I can now see some text more clearly. ……I will crop the region from 20% to 80% of the width and from 40% to 70% of the height of observation 3. </think><grounding> {"bbox_2d": [0.2, 0.2, 0.8, 0.8], "source": "observation_3"}</grounding>

T5: Repeats the header search again.

<think> I have zoomed in on the middle part of the image in observation 3. I can now see some text more clearly. ..... I will crop the region from 20% to 80% of the width and from 40% to 70% of the height of observation 3.</think><grounding>{"bbox_2d ": [0.2, 0.2, 0.8, 0.8], "source": "observation_3"}</grounding>

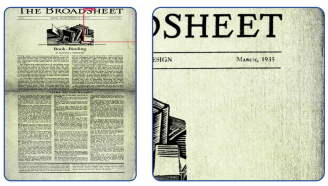
T6: Over-zooms a partial date.

<think> I have zoomed in on the middle part of the image in observation 3. I can now see some text more clearly. ..... I will crop the region from 20% to 80% of the width and from 40% to 70% of the height of observation 3.</think><grounding>{"bbox_2d ": [0.2, 0.2, 0.8, 0.8], "source": "observation_3"}</grounding>

T7: Returns the incomplete March 3.

<think> I have zoomed in on the middle part of the image in observation 3. I can now see some text more clearly. I can spot the date "March 3" in the text. ……</think>**<answer> March 3 </answer>**

HaPRL 2rounds / 1crops — *Correct with one crop* — March, 1935 — Correct



Condensed rollout

T1: Directly crops the article's displayed header.

<think> I can see the text "MARCH, 1935" at the top right of the image, which is likely the date. The text is located in the upper right quadrant of the image. I will select a bounding box that covers this area to get a clearer view.</think><grounding>{"bbox_2d": [0.6, 0.0, 1.0, 0.2], "source": "original_image"}</grounding>

T2: Reads MARCH, 1935 and answers.

<think> In the previous step, I zoomed in on the top right corner of the image where I suspected the date was located. The observation shows the text "MARCH, 1935" clearly. The image is clear enough to read the text, so no further grounding is needed. I can now provide the final answer. </think>**<answer> MARCH, 1935 </answer>**

Figure 6: **Historical-date OCR** (VisualProbe Easy-128). The cold start zooms into irrelevant publication text and reports a date that is not in the image. Outcome-RL chases several date-like lines across six crops and returns the truncated "March 3". HaPRL crops the article's displayed header directly and reads "March, 1935" with one crop.

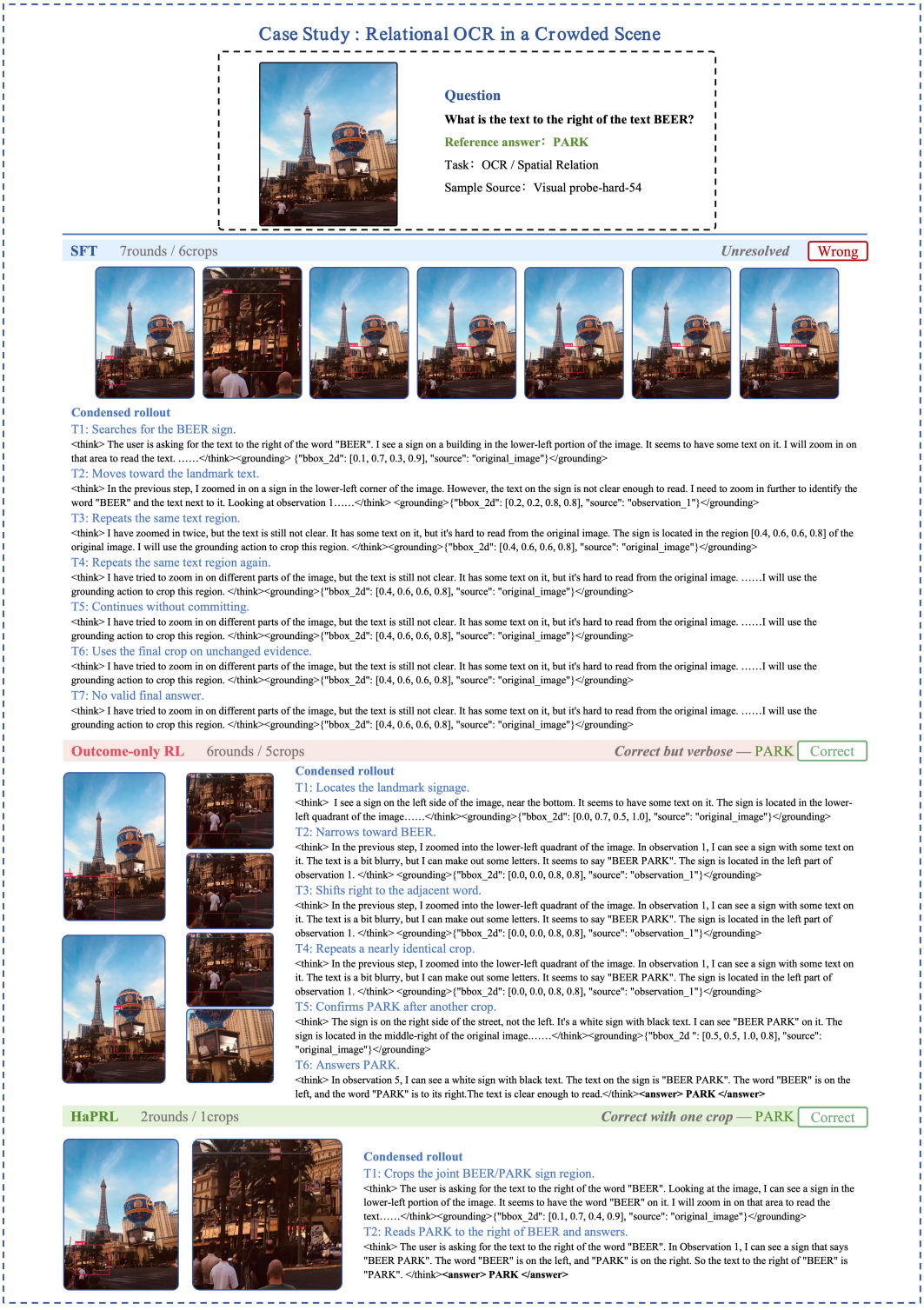


Figure 7: **Relational OCR in a crowded scene** (VisualProbe Hard-54). The question asks for the text to the right of BEER, so the two words must be legible in one view. The cold start re-crops the same unreadable region four times and returns no valid answer. Outcome-RL reaches the right answer but repeats a nearly identical crop and spends a further round confirming what it had already read. HaPRL crops the joint sign region once, which makes the relation readable immediately.

Case Study: Text-color Recognition in a Dense Shelf



Question

What color is the word "MILK"?

Reference answer: red

Task: Attribute

Sample Source: Visual probe-medium-151

SFT 7rounds / 6crops *Target lost* Wrong



Condensed rollout

T1: Searches a broad shelf region.

<think> I can see many products on the shelves. On the left side of the aisle, there are several shelves with different products. The shelf above it has blue cans. The shelf above that has red cans. The shelf above that has orange boxes.....</think><grounding> {"bbox_2d": [0.1, 0.4, 0.3, 0.6], "source": "original_image"}</grounding>

T2: Moves to another shelf segment.

<think> The user is asking for the color of the word "MILK". I have zoomed in on the orange boxes on the left side of the aisle, but I cannot find the word "MILK" in this area. The text on the orange boxes is blurry and I cannot read it clearly..... </think> <grounding>{"bbox_2d": [0.8, 0.0, 1.0, 0.2], "source": "original_image"}</grounding>

T3: Loses the MILK label after zooming.

<think> I have zoomed in on the green boxes on the right side of the aisle, but I cannot find the word "MILK" in this area. The text on the green boxes is blurry and I cannot read it clearly. I will now examine the products on the left side of the aisle again..... </think> <grounding>{"bbox_2d": [0.1, 0.3, 0.3, 0.5], "source": "original_image"}</grounding>

T4: Retries a different aisle region.

<think> The text on the blue cans is blurry and I cannot read it clearly. I will now examine the products on the right side of the aisle again. The text on these cans is also blurry, but it might contain the word "MILK"..... </think> <grounding>{"bbox_2d": [0.8, 0.2, 1.0, 0.4], "source": "original_image"}</grounding>

T5: Repeats an unhelpful shelf crop.

<think> The text on the blue cans is blurry and I cannot read it clearly. I will now examine the products on the right side of the aisle again. The text on these cans is also blurry, but it might contain the word "MILK"..... </think> <grounding>{"bbox_2d": [0.0, 0.8, 0.2, 1.0], "source": "original_image"}</grounding>

T6: Final crop still lacks the target.

<think> The text on the blue cans is blurry and I cannot read it clearly. I will now examine the products on the right side of the aisle again. The text on these cans is also blurry, but it might contain the word "MILK"..... </think> <grounding>{"bbox_2d": [0.8, 0.5, 1.0, 0.7], "source": "original_image"}</grounding>

T7: Concludes the color cannot be determined.

</think>**<answer> I cannot determine the color of the word 'MILK' as it is not visible in the image.</answer>**

Outcome-only RL 7rounds / 6crops *Max-round failure* Wrong



Condensed rollout

T1: Scans the shelves for MILK.

<think> I can see many products on the shelves. On the left side of the aisle, there are several rows of shelves. The shelf above that has blue cans. The shelf above that has bottles of lemon juice. The shelf above that has red cans. The shelf above that has green boxes.....</think><grounding>{"bbox_2d": [0.0, 0.4, 0.5, 0.8], "source": "original_image"}</grounding>

T2: Moves to a candidate product row.

<think> I have zoomed into the left side of the aisle, but I don't see the word "MILK" in this cropped image. The green boxes have some text, but it's not clear.Let's look at the right side of the aisle.There are some yellow boxes on the top shelf on the right side.....</think> <grounding>{"bbox_2d": [0.7, 0.0, 1.0, 0.4], "source": "original_image"}</grounding>

T3: The crop loses the target label.

<think> I have zoomed into the top right corner of the aisle, but I don't see the word "MILK" in this cropped image. Let's look at the bottom shelf on the right side of the aisle. The region is approximately [0.2, 0.5, 0.6, 1.0]..... </think><grounding>{"bbox_2d": [0.2, 0.5, 0.6, 1.0], "source": "original_image"}</grounding>

T4: Backtracks to another shelf.

<think> I have zoomed into the bottom right corner of the aisle, but I don't see the word "MILK" in this cropped image. The boxes have some text on them, but it's not clear what it says. On the left side, there are red cans with "Cola",..... </think><grounding> {"bbox_2d": [0.6, 0.0, 1.0, 0.2], "source": "original_image"}</grounding>

T5: Repeats a broad shelf search.

<think> I have zoomed into the top right corner of the aisle again, but I still don't see the word "MILK". Let's look at the blue boxes on the right side of the aisle again. The text is a bit blurry. Let me try to zoom in on the blue boxes on the right side of the aisle.....</think><grounding>{"bbox_2d ": [0.5, 0.4, 1.0, 0.8], "source": "original_image"}</grounding>

T6: Uses the last crop without readable evidence.

<think> I have zoomed into the right side of the aisle, focusing on the blue boxes of pasta. I can see the word "Spaghetti" and "Thin Spaghetti" on the boxes. I also see some text that says "AL DENTE" and "Source of Protein & Fiber".....</think><grounding>{"bbox_2d ": [0.1, 0.4, 0.5, 0.8], "source": "original_image"}</grounding>

T7: No valid final answer.

<think> The text on the boxes is still not very clear. </think><grounding>{"bbox_2d ": [0.2, 0.0, 0.6, 0.4], "source": "original_image"}</grounding>

HaPRL 3rounds / 2crops *Correct evidence path* — red Correct



Condensed rollout

T1: Locates the shelf containing MILK.

<think> The user is asking for the color of the word "MILK". I can see many products on the shelves. On the left side of the aisle, there are bottles of "Lemon Juice" and "Cola". On the right side, there are boxes of "Spaghetti",..... </think><grounding>{"bbox_2d": [0.3, 0.3, 0.7, 0.7], "source": "original_image"}</grounding>

T2: Tightens while keeping the word visible.

<think> The user is asking for the color of the word "MILK". In this new observation, I can see more products on the shelves. On the left side of the shelves, there are some yellow boxes with the word "MILK" written on them. The word "MILK" is written in red color..... </think><grounding>{"bbox_2d": [0.1, 0.1, 0.5, 0.5], "source": "observation_1"}</grounding>

T3: Identifies the red lettering and answers red.

<think> I have zoomed in on the yellow boxes in observation 1. In observation 2, I can clearly see the word "MILK" written in red on the yellow boxes. The text is inside a heart shape. The color is definitely red. I am confident in this answer and no further grounding is needed. **</think><answer> red </answer>**

Figure 8: **Text-color recognition on a dense shelf** (VisualProbe Medium-151). Both controls lose the target among repeated shelf crops and end without a usable answer, the cold start concluding the color cannot be determined. HaPRL first locates the shelf holding the queried word, then tightens while keeping the word visible, and reads its color in three rounds.

Case Study: Small-object Color Disambiguation



Question

What is the color of the flag?

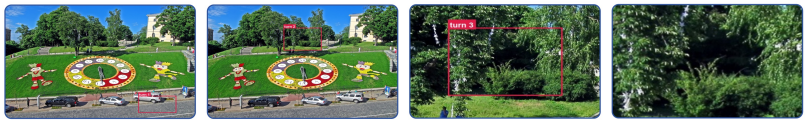
(A) red and yellow (B) blue and yellow (C) blue and red (D) green and white

Reference answer: (B) blue and yellow

Task: Attribute

Sample Source: Vstar-direct_attributes-58

SFT 4rounds / 3crops — *Incorrect target* — (C)blue and red **Wrong**



Condensed rollout

T1: Mistakes a bottom-right object for the flag.

<think> I can see a small flag on the right side of the road, near a silver car. It appears to have two horizontal stripes. To determine the colors accurately, I need to zoom in on that specific area. The flag is located in the bottom right corner of the image. </think><grounding> {"bbox_2d": [0.7, 0.8, 0.9, 0.95], "source": "original_image"}</grounding>

T2: Re-localizes to a different pole.

<think> In the previous step, I zoomed in on the bottom right corner of the original image, where I thought the flag was located. However, after examining `observation_1`, I realize that my initial assumption was incorrect. Looking at the original image again…… </think> <grounding>{"bbox_2d": [0.4, 0.2, 0.6, 0.4], "source": "original_image"}</grounding>

T3: Zooms a three-stripe candidate.

<think> I have successfully located the flag in the previous step. Now, in `observation_2`, I can see the flag more clearly. It appears to be a vertical tricolor flag with three stripes. From top to bottom, the colors seem to be blue, white, and red. To be absolutely sure…… </think><grounding> {"bbox_2d": [0.2, 0.2, 0.8, 0.8], "source": "observation_2"}</grounding>

T4: Answers blue/red (C).

<think> The user wants to know the color of the flag. I have successfully zoomed in on the flag in `observation_3`. The flag is clearly visible and I can now confidently identify its colors. From left to right, the colors are blue, white, and red. The question asks for the color of the flag, and I have identified it as blue, white, and red. However, the options provided are pairs of colors. I am confident in this answer and no further grounding is needed. </think>**<answer> (C) blue and red </answer>**

Outcome-only RL 4rounds / 3crops — *Incorrect target* — (C)blue and red **Wrong**



Condensed rollout

T1: Searches an upper-right flagpole.

<think> I can see a flagpole on the right side of the image, near the top of the grassy hill, next to the yellow building. It's quite small, so I'll need to zoom in to see the colors clearly. The flagpole is located in the upper right quadrant of the image……</think><grounding>{"bbox_2d": [0.7, 0.1, 1.0, 0.4], "source": "original_image"}</grounding>

T2: Jumps to a left-side pole.

<think> I zoomed in on the upper right quadrant of the image where I thought the flagpole was. However, in observation 1, I don't see a flag. It seems my initial guess for the location of the flagpole was incorrect. Looking at the original image again, I can see a flagpole on the far left side of the image, partially obscured by a tree. I'll use the bounding box [0.0, 0.3, 0.3, 0.7] to focus on that area.</think> <grounding>{"bbox_2d": [0.0, 0.3, 0.3, 0.7], "source": "original_image"}</grounding>

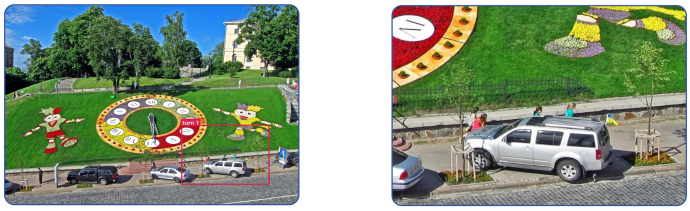
T3: Zooms the wrong flag candidate.

<think> In observation 2, I can see a flagpole on the left side of the image, and there is a flag at the top of it. The flag appears to have two colors. I need to zoom in further to identify the colors accurately.</think><grounding>{"bbox_2d": [0.0, 0.0, 0.5, 0.5], "source": "observation_2"}</grounding>

T4: Answers blue/red (C).

<think> The user is asking for the color of the flag. In the previous step, I zoomed in on the flag in observation 2. In observation 3, I can see the flag more clearly. It appears to be a vertical flag with two colors. I am confident in this answer and no further grounding is needed. </think>**<answer> (C) blue and red </answer>**

HaPRL 2rounds / 1crops — *Correct with one crop* — **(B) blue and yellow** Correct



Condensed rollout

T1: Directly crops the relevant roadside flag.

<think> The user is asking about the color of a flag in the image. Below the clock, there are cars and a road. On the right side of the road, near a silver car, I can see a small flag. It appears to have two horizontal stripes. To determine the colors accurately, I need to zoom in on that specific area. The flag is located in the lower right portion of the image. </think><grounding>{"bbox_2d": [0.6, 0.6, 0.9, 0.9], "source": "original_image"}</grounding>

T2: Sees blue/yellow and answers B.

<think> The user is asking about the color of the flag. In the previous step, I zoomed into the lower right corner of the image, where I thought the flag was located. In Observation 1, I can see a flag more clearly. I am confident in this observation and can provide the final answer.</think>**<answer> (B) blue and yellow </answer>**

Figure 9: **Small-object color disambiguation** (V* direct_attributes-58). Both controls fix on a salient flagpole rather than the queried flag and report the colors of the wrong object, arriving at the same incorrect option by different routes. HaPRL crops the roadside flag directly and answers in two rounds. A wrong target reached fluently is the failure the rubric's target-semantics dimension is meant to price.