Towards On-Policy Data Evolution for Visual-Native Multimodal Deep Search Agents

Shijue Huang[α,β], **Hangyu Guo**[α,β], **Guanting Dong**[γ], **Chenxin Li**[δ]
Junting Lu[ϵ], **Xinyu Geng**[α,β], **Zhaochen Su**[α,β], **Zhenyu Li**[ζ]
Shuang Chen[δ], **Hongru Wang**[η], **Yi R. (May) Fung**[α,β †]
[α]The Hong Kong University of Science and Technology, [β]MMSense Lab
[γ]Renmin University of China, [δ]The Chinese University of Hong Kong
[ϵ]Peking University, [ζ]Tsinghua University, [η]University of Edinburgh
shijue.huang@connect.ust.hk yrfung@ust.hk

Abstract

Multimodal deep search requires an agent to solve open-world problems by chaining search, tool use, and visual reasoning over evolving textual and visual context. Two bottlenecks limit current systems. First, existing tool-use harnesses treat images returned by search, browsing, or transformation as transient outputs, so intermediate visual evidence cannot be re-consumed by later tools. Second, training data is usually built by fixed curation recipes that cannot track the target agent's evolving capability. To address these challenges, we first introduce a visual-native agent harness centered on an *image bank reference protocol*, which registers every tool-returned image as an addressable `<image:N>` reference and makes intermediate visual evidence reusable by later tools. On top of this harness, On-policy Data Evolution (ODE) runs a closed-loop data generator that refines itself across rounds from rollouts of the policy being trained. This per-round refinement makes each round's data target what the current policy still needs to learn. The same framework supports both diverse supervised fine-tuning data and policy-aware reinforcement learning data curation, covering the full training lifecycle of the target agent. Across 8 multimodal deep search benchmarks, ODE improves the Qwen3-VL-8B agent from 24.9% to 39.0% on average. At 30B, ODE raises the average score from 30.6% to 41.5%. Further analyses validate the effectiveness of image bank reuse, especially on complex tasks requiring iterative visual refinement, while rollout-feedback evolution yields more grounded SFT traces and better policy-matched RL tasks than static synthesis.[1]

1 Introduction

Recently, Multimodal Large Language Models (MLLMs) have witnessed a rapid emergence of agent capabilities, pushing their application boundary from static image-question answering toward open-world deep search [OpenAI, 2023, ByteDance Seed, 2026, Bai et al., 2025, Jiang et al., 2025, Li et al., 2025, Tao et al., 2026]. In this emerging setting, a model is expected to interact with search engines and a broad ecosystem of external tools in real time, gathering evidence to generate grounded answers. In practice, user information needs are becoming increasingly complex and open-ended, where shallow retrieval no longer suffices to capture their intent [Jiang et al., 2025, Li et al., 2025, Tao et al., 2026, Su et al., 2026]. To succeed, an agent must propagate fine-grained visual cues across turns, cross-verify evidence across modalities, and sustain coherent multi-step reasoning in noisy,

†Corresponding author.

[1]Our implementation is available at `https://github.com/JoeYing1019/ODE`.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

evolving environments. This makes multimodal deep search far more demanding, requiring not only perception but also tool use and cross-modal evidence integration over an expanding context.

Building strong multimodal deep search agents remains challenging for two reasons: **(1) Existing pipelines underutilize persistent visual state in tool-augmented search:** Early multimodal search agents augment MLLMs with image and text search to enable on-demand retrieval in open-world environments [Wu et al., 2025, Geng et al., 2026b], and subsequent works extend this paradigm with crop-conditioned image search, iterative query refinement, and increasingly complex multi-turn visual-textual exploration [Narayan et al., 2025, Hong et al., 2026, Huang et al., 2026]. However, many existing approaches still center visual reasoning and search around the original task image, rather than treating tool-produced visual outputs as new reusable evidence throughout the trajectory. **(2) Multimodal deep search data synthesis lacks closed-loop modeling of agent search behavior.** Recent works mainly rely on synthetic or semi-automatically constructed data. For instance, MMSearch-R1 [Wu et al., 2025], DeepMMSearch-R1 [Narayan et al., 2025], and WebWatcher [Geng et al., 2026b] obtain training data via semi-automated VQA curation, web-grounded task synthesis, and synthetic multimodal tool-use trajectories. These efforts mark important progress, but the generation recipe is usually fixed before scaling, making it difficult to use target-agent feedback to steer data toward the policy's learning frontier.

These observations suggest that progress in multimodal deep search depends on *jointly advancing the agent's interaction workspace and the way its training data are constructed.* Motivated by this, we seek to elicit multimodal deep search capability through a co-design along these two complementary axes. On the workspace side, instead of treating multimodal search as a fixed interaction over the original task image, we build a **Visual-Native Agent Harness** that unifies 9 core tools in a shared workspace: web search, image search, scholar search, visit (browsing), visual search (Google Lens), zoom-in, rotation, flip, and Python execution. At its core is an *image bank reference protocol*. It stores the original task image and every tool-returned image as reusable visual state, allowing later actions to operate on visual evidence produced by earlier steps. This turns multimodal search from single-image interaction into a chained visual workflow with evidence accumulation.

On the data side, built on our visual-native harness, we introduce **On-policy Data Evolution (ODE)**, which treats multimodal data construction as adaptive optimization rather than a fixed curation recipe. Instead of designing a synthesis pipeline once and then scaling it, ODE repeatedly generates candidate tasks, executes the target policy on them, and uses rubric-based trace analysis as feedback to revise the next round of data synthesis. In this sense, the rubric plays a role analogous to a *loss function*: it identifies whether the generated data is too easy, too brittle, insufficiently visual, poorly grounded, or otherwise misaligned with the agent's current training needs. The same evolution principle supports both supervised fine-tuning (SFT) and reinforcement learning (RL) with mode-specific objectives: ODE favors grounded, tool-effective, and diverse teacher trajectories for SFT, and seeks verifiable tasks near the policy's learning frontier for RL.

Experiments across eight challenging multimodal deep search benchmarks spanning MMBC, HLE-VL, BC-VL, MMSearch, VDR, MMSearch+, SimpleVQA, and FVQA show that the proposed framework substantially improves agent performance at both 8B and 30B scales when evaluated under the same visual-native harness. Further controlled analyses show that both parts of the framework matter: removing reusable image bank references weakens performance most on tasks that activate secondary image use, while replacing ODE with a static synthesis recipe yields lower SFT and RL gains under matched data budgets.

To summarize, our contributions are as follows:

- We introduce a **Visual-Native Agent Harness** for multimodal deep search, where search, browsing and visual manipulation operate over an image bank reference protocol that makes tool-produced visual evidence persistently reusable across the trajectory.
- We propose **On-policy Data Evolution (ODE)**, a closed-loop data construction framework that couples task synthesis, policy rollout, rubric-based trace analysis, and configuration optimization, and supports both SFT-style teacher-trace curation and RL training data generation.
- We validate the framework across eight multimodal deep search benchmarks. ODE improves Qwen3-VL from 24.9% to 39.0% at 8B and from 30.6% to 41.5% at 30B on average, verifying the effectiveness of visual-state reuse and data evolution against static synthesis.

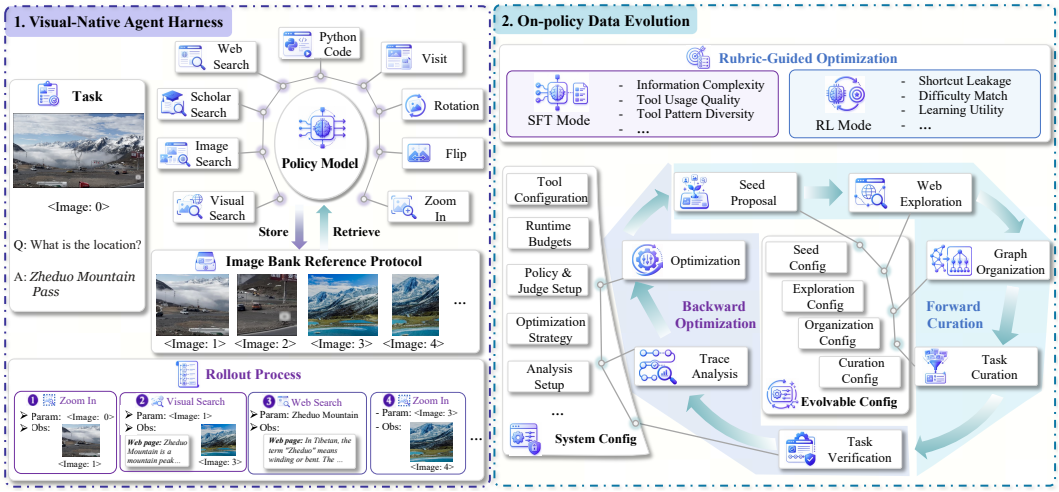


Figure 1: Overview of our framework. **Left:** The visual-native agent harness unifies 9 tools in a shared workspace and enables reusable visual state through the image bank reference protocol. **Right:** ODE constructs data with a closed loop over the harness: the forward pipeline synthesizes grounded tasks, and the backward pipeline uses rollout traces to refine the next generation configuration.

2 Method

Overview. In this section, we present the proposed framework, as illustrated in Fig. 1. To further unlock the multimodal deep search agent's ability, we first propose the visual-native agent harness (Section 2.1), which lets the agent reuse intermediate visual evidence through an image bank reference protocol that registers tool-returned images and makes them available to subsequent tool calls. Then, unlike static data-synthesis approaches, we propose On-policy Data Evolution (ODE, Section 2.2), a closed-loop data construction process that treats the data generator as the object being optimized. In each ODE epoch, ODE first uses the current generator configuration to synthesize a set of candidate tasks. The target policy then rolls out these tasks in the visual-native harness, producing interaction traces of tool calls, observations, and final answers. A mode-specific rubric scores these traces and revises the generator configuration, so that the next ODE epoch produces data better aligned with the SFT or RL objective. The data generator therefore evolves with policy feedback round by round, rather than being fixed by a static curation recipe.

2.1 Visual-Native Agent Harness

Multimodal deep search often requires iterative search, browsing, visual manipulation, and computation before answering. However, existing approaches typically organize visual reasoning around the original task image, so tool-returned images are not consistently exposed as reusable inputs to later tools. As a result, visual evidence cannot propagate across tool calls as naturally as textual evidence. To address this, our visual-native harness introduces an *image bank reference protocol*, shown in Fig. 1 (left), which assigns each initial or tool-returned image an addressable `<image:N>` reference and makes it available to subsequent tool calls.

Specifically, we represent a multimodal deep search task handled by the harness as $\mathcal{T} = (q, \mathcal{I}, a)$, where q is an open-world multimodal query that requires the agent to gather evidence and reason across modalities, $\mathcal{I}$ is the initial visual context loaded into the image bank, and a is the reference answer for verification. Each image in the bank carries an identifier of the form `<image:N>` that later tool calls reference. Starting from $(q, \mathcal{I})$, the policy model uses the nine tools shown in Fig. 1, covering web and scholarly retrieval, image and visual search, source browsing, image transformation, and Python-based computation. The Rollout Process in Fig. 1 illustrates this for the question *"What is the location?"*: the agent calls `zoom_in` on the input photo `<image:0>` to crop a mountain region into `<image:1>`, runs `visual_search` on `<image:1>` to retrieve a candidate name with a clearer photo `<image:3>`, follows up with `web_search` to verify the candidate, and zooms into `<image:3>` to decide the final answer is *"Zheduo Mountain Pass"*.

2.2 On-policy Data Evolution

2.2.1 Forward Curation

Building on the visual-native harness above, ODE represents the data generator with two configuration objects: a fixed *System Config*, which defines the execution environment and evaluation protocol, and an editable *Evolvable Config* $\mathcal{C}_t$, which carries the generator parameters adapted from rollout feedback across rounds. ODE initializes $\mathcal{C}_0$ by setting the seed, exploration, organization, curation, and optimization-strategy components. After round t, backward refinement revises these components to produce $\mathcal{C}_{t+1}$. We next illustrate the four forward stages controlled by $\mathcal{C}_t$, which together turn open-world evidence into a verifiable multimodal deep search task.

Seed Proposal. The seed proposer comes up with seeds, each consisting of an entity together with an associated image, that anchor downstream exploration. We also establish a balanced sampling mechanism to guide seed proposal across 11 topical domains, 4 capability-requirement profiles spanning perception-only, perception+search, perception+reasoning, and perception+search+reasoning tasks, and 4 difficulty levels. After dropping duplicates from earlier rounds, an LLM judge accepts a seed only if its image carries visual evidence such as labels, numbers, or dates, and its entity can be cross-checked against at least two independent web sources. This binds each image to a stable real-world entity and ensures that downstream tasks are grounded in verifiable evidence.

Web Exploration. For each retained seed, the explorer uses the harness's 9 tools to gather supporting evidence and organizes it into *anchors*, each focused on one related sub-topic such as an official source page or a map image returned by visual search. Each anchor records: (i) a textual, visual, or numerical fact, (ii) its source URL, (iii) any tool-returned image handle in the Image Bank, and (iv) its relation to the seed or to other anchors. The Exploration Config in $\mathcal{C}_t$ specifies the total and image-bearing anchor budgets.

Graph Organization. The graph organizer assembles the collected nodes for each seed into a multimodal evidence graph G. Each node contains text, image, or tool-derived observations, and edges capture source links, entity or event relations, and cross-modal dependencies. The organizer further enriches G with two kinds of new nodes. *Reasoning nodes* are produced by running `python_code` and `visit` over related observations to compute quantities or cross-verify facts across sources, and *perception nodes* are produced by running `zoom_in`, `rotation`, `flip`, and `visual_search` on existing images to expose fine-grained visual details. These enrichments make derived relations, calculations, and visual details explicit in G, providing richer evidence for task curation.

Task Curation. The curator selects a connected evidence cluster from G and turns it into a verifiable candidate task $(q, \mathcal{I}_0, a) \in \mathcal{D}_t^{\text{cand}}$, together with auxiliary annotations such as planned reasoning steps, capability requirements, and difficulty. It then refines the candidate question to increase reasoning depth, for example by requiring additional evidence, hiding obvious entity names, or removing shortcut clues while preserving the ground-truth answer. For difficulty control, the weights in $\mathcal{C}_t$ determine the curator's preference over difficulty levels, allowing backward refinement to adjust the difficulty distribution in later ODE epochs. Before a task enters $\mathcal{D}_t^{\text{cand}}$, validation checks that its image references resolve, its answer is unambiguous, and its question does not reveal the intended tool-use path.

2.2.2 Backward Optimization

Backward optimization evaluates whether the candidate tasks produced by forward exploration are useful for training and how the generator should change in the next ODE epoch. As shown in Fig. 1, it first verifies each task by rolling out the target policy in the visual-native harness, then analyzes the resulting traces, and finally uses rubric-guided optimization to update the generation configuration.

Task Verification. Each candidate $x_i = (q_i, \mathcal{I}_{0,i}, a_i) \in \mathcal{D}_t^{\text{cand}}$ is executed in the visual-native harness by the rollout model m_t. For SFT, m_t is a teacher model, and successful rollouts provide candidate demonstrations for distillation. For RL, m_t is the current policy, so the rollout measures whether the task is appropriate for the policy that will train on it. The execution produces a trace τ_i containing the message history, Image Bank references, final answer, and judge outcome.

Trace Analysis. Trace Analysis evaluates each rollout trace τ_i together with the construction record of x_i, including the seed image, explored sources, evidence graph, and task annotations. It returns a diagnosis δ_i containing rubric scores and stage-level fault localization. The shared rubric

dimensions assess whether the task is information-rich, visually grounded, non-leaky, and objectively verifiable. In addition, each mode has its own training-utility dimensions. The SFT rubric adds Step Appropriateness, Tool Usage Quality, and Tool Pattern Diversity to evaluate whether a trace is suitable as a teacher demonstration. The RL rubric adds Capability Requirement, Difficulty Match, and Learning Utility to evaluate whether a task provides a useful policy-optimization signal.

The diagnosis also routes each failure mode to the stage responsible for repair. Weak seed images or entity–image mismatch are attributed to Seed Proposal; topic drift or unsupported sources to Web Exploration; missing computations or visual transformations to Graph Organization; and leaked, ambiguous, or miscalibrated questions to Task Curation.

Rubric-Guided Optimization. The final optimization stage aggregates the per-trace diagnoses δ_i into an epoch-level signal Δ_t for updating the data generator. The goal is to make the next ODE epoch better match the mode-specific rubric objectives, rather than merely increasing rollout success on the current candidates. Concretely, Δ_t edits $\mathcal{C}_t$ into $\mathcal{C}_{t+1}$ by modifying the stage-specific Seed, Exploration, Organization, and Curation configs, together with the Optimization Strategy. Low seed grounding shifts the Seed Config toward entities with stronger image evidence and source support. Sparse or repetitive evidence adjusts the Exploration Config by changing search breadth, phase depth, or the share of image-bearing anchors. Missing computations or visual operations update the Organization Config with additional reasoning or perception enrichment. Leakage, ambiguous answers, or off-target difficulty revise the Curation Config through difficulty weights, enhancement prompts, and validation constraints. The Optimization Strategy records these stage-local edits together with aggregate rubric scores, recent score trends, batch health, tail failures, SFT diversity, and RL policy pass rate. The next forward pass uses $\mathcal{C}_{t+1}$, whose rollouts are analyzed again to produce $\mathcal{C}_{t+2}$ when the policy exposes new failure modes. Through this continued iteration, ODE moves SFT data toward diverse, high-quality demonstrations and RL data toward verifiable tasks at the policy's learning frontier.

We provide a full worked example of the ODE pipeline in Appendix A, including the round configuration, forward generation stages, rollout verification, trace analysis, rubric-guided optimization, and consecutive configuration updates across two ODE epochs.

2.3 Statistics of ODE-Curated Data

Figure 2 reports topic domain coverage and curator-annotated difficulty for the SFT training set and the two policy-specific RL task sets ODE-8B and ODE-30B-A3B curated by ODE for backbones of two different sizes, with the per-domain breakdown of the two RL sets and the planned reasoning-step distribution given in Appendix A.11.

Topical breadth is preserved. The SFT demonstration set covers all eleven topical domains, as shown in Fig. 2a. The ODE-8B and ODE-30B-A3B task sets cover the same domains, and their per-domain distributions remain balanced, with coefficients of variation around 0.05. Thus, policy-specific data evolution does not collapse topical coverage while adapting task difficulty.

Difficulty tracks policy capability. The Hard plus Expert share rises from 29.06% on the SFT set to 61.85% on ODE-8B and 93.67% on ODE-30B-A3B, while Easy tasks fall from 41.54% to 0.38% over the same progression, as shown in Fig. 2b. The pass-rate and difficulty-match signals collected by the analyzer in Section 2.2.2 push the curator toward each policy's learning frontier, so stronger policies receive proportionally harder tasks and almost no trivially easy ones.

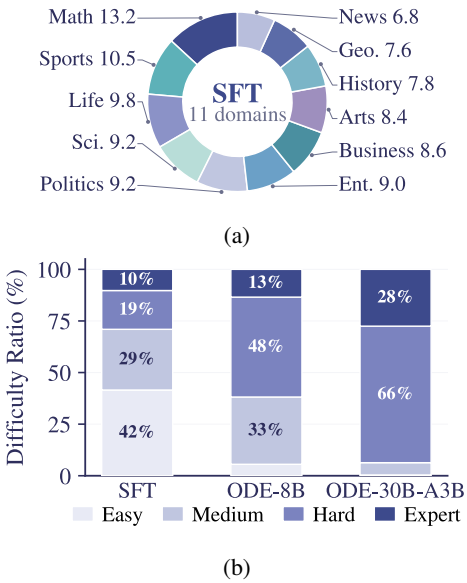


Figure 2: **Statistics of ODE-curated data.** **(a)** Topic domain coverage of the SFT demonstration set, in %. **(b)** Curator-annotated difficulty ratio across the three datasets.

Table 1: Main results. Avg denotes the average score over 8 benchmarks. Δ denotes the improvement over base model. The best results are highlighted in **bold**, and the second-best results are underlined.

Model	MMBC	HLE-VL	BC-VL	VDR	MMSearch	MMSearch+	SVQA	FVQA	Avg.
Direct Reasoning									
GPT-5	11.2	15.8	42.9	10.8	40.7	14.0	66.0	56.3	32.2
Claude-4-Sonnet	-	-	29.3	2.0	18.7	4.0	-	35.3	-
Claude-3.7-Sonnet	-	-	32.3	4.6	21.1	4.0	-	36.7	-
Gemini-2.5 Pro	10.3	**19.0**	43.1	8.0	39.8	14.5	72.7	60.7	33.5
Gemini-2.5 Flash	4.9	12.0	37.1	6.2	30.4	8.1	67.0	47.7	26.7
Qwen3-VL-8B	4.0	6.1	25.1	2.8	15.2	3.2	42.7	28.0	15.9
Qwen3-VL-30B	4.5	5.0	29.6	3.8	18.7	3.2	47.0	34.7	18.3
Agent Workflow									
GPT-5	-	18.1	**57.6**	17.6	62.7	**28.5**	67.3	62.0	-
Claude-4-Sonnet	-	-	48.6	13.6	67.2	23.1	-	69.0	-
Claude-3.7-Sonnet	-	-	50.4	**27.2**	63.7	17.2	-	67.3	-
Gemini-2.5 Pro	**13.8**	17.3	42.3	10.0	55.7	24.9	**74.3**	65.0	37.9
Gemini-2.5 Flash	11.6	8.5	38.1	7.8	53.3	14.5	67.3	54.7	32.0
Qwen3-VL-8B	6.2	5.0	24.3	5.0	48.7	9.0	51.3	45.3	24.4
Qwen3-VL-30B	6.2	4.4	29.3	4.4	49.3	6.8	53.0	45.3	24.8
Multimodal Deep Search Agent									
MMSearch-R1-7B	-	-	-	-	53.8	-	57.4	58.4	-
WebWatcher-7B	-	10.6	20.3	-	49.1	-	54.3	-	-
WebWatcher-32B	-	13.6	26.7	-	55.3	-	59.0	-	-
Visual-Native Agent Harness (Ours)									
Qwen3-VL-8B	7.6	6.1	26.1	4.2	47.3	10.0	53.0	44.7	24.9
+ ODE-SFT	8.5	8.2	39.6	19.2	61.7	21.3	68.3	62.0	36.1
+ ODE-RL	12.5	11.4	41.9	20.4	66.0	24.9	70.3	64.7	39.0
Δ	+4.9	+5.3	+15.8	+16.2	+18.7	+14.9	+17.3	+20.0	+14.1
Qwen3-VL-30B	7.1	8.5	32.8	11.0	57.0	10.0	63.7	54.3	30.6
+ ODE-SFT	10.3	9.9	43.1	24.0	65.7	26.7	70.3	66.3	39.5
+ ODE-RL	11.2	10.5	46.1	26.4	**69.7**	28.1	71.0	**69.3**	**41.5**
Δ	+4.1	+2.0	+13.3	+15.4	+12.7	+18.1	+7.3	+15.0	+10.9

3 Experiments

3.1 Experimental Setup

Datasets. We evaluate our approach on 8 multimodal deep search and related multimodal reasoning benchmarks: MM-BrowseComp (MMBC) [Li et al., 2025], HLE-VL [Center for AI Safety et al., 2026], BC-VL [Geng et al., 2026b], VDR [Zeng et al., 2026], MMSearch [Jiang et al., 2025], MMSearch+ [Tao et al., 2026], SimpleVQA (SVQA) [Cheng et al., 2025], and FVQA [Wang et al., 2017]. Details of these benchmarks are provided in Appendix B.4.

Baselines. We compare against proprietary and open-source multimodal models and agents under three evaluation settings. In the *Direct Reasoning* setting, models answer in a single pass without external retrieval or tool use. This group includes GPT-5 [OpenAI, 2026], Claude Sonnet 4 [Anthropic, 2025b], Claude 3.7 Sonnet [Anthropic, 2025a], Gemini-2.5 models [Gemini Team, 2025], and the Qwen3-VL-8B-Instruct and Qwen3-VL-30B-A3B-Instruct backbones [Bai et al., 2025]. In the *Agent Workflow* setting, models are equipped with a general multimodal deep search toolset, including web search, webpage browsing, image search, and image manipulation, following prior work [Huang et al., 2026, Narayan et al., 2025]. They are prompted to solve tasks through iterative reasoning and tool use. We also compare with recent dedicated multimodal deep search agents, including MMSearch-R1 [Wu et al., 2025] and WebWatcher [Geng et al., 2026b]. For training, we instantiate our framework with two Qwen3-VL backbones: Qwen3-VL-8B-Instruct and Qwen3-VL-30B-A3B-Instruct [Bai et al., 2025]. We refer to them as Qwen3-VL-8B and Qwen3-VL-30B for brevity. Further details on data construction, training, and evaluation are provided in Appendices B.1, B.2, and B.3.

3.2 Main Results

Tab. 1 reports the main results. Three-run inference statistics for the final RL checkpoints are reported in Appendix B.7. We highlight the following observations.

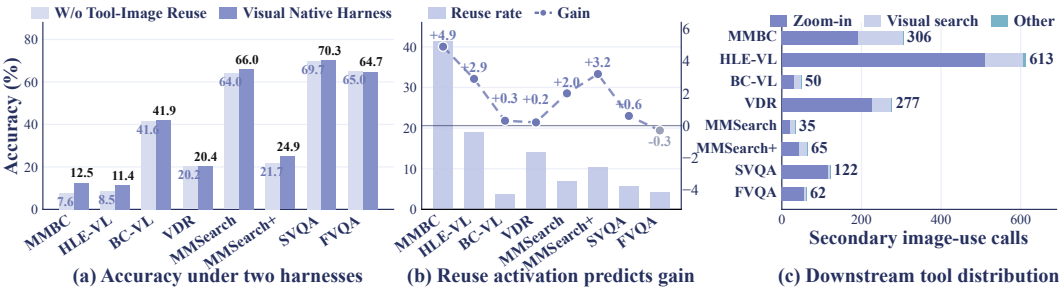


Figure 3: Visual-native harness ablation on ODE-8B-RL.

(1) ODE catalyzes multimodal deep search capability. Under the same visual-native harness, ODE improves the Qwen3-VL-8B agent from 24.9% to 39.0% average accuracy, and the Qwen3-VL-30B agent from 30.6% to 41.5%. The gains are not uniform score inflation: they are largest on benchmarks that require iterative evidence gathering and cross-modal grounding, such as VDR, MMSearch, MMSearch+, and FVQA. This suggests that ODE mainly improves the agent's ability to search, inspect, and integrate multimodal evidence over multiple steps.

(2) Tool access is not tool competence. Equipping Qwen3-VL backbones with a standard agent workflow improves over direct answering, but these tool-using baselines remain far below agents trained on ODE-curated data. This indicates that multimodal deep search is not unlocked by tool access alone: the model must learn when to search, when to inspect visual evidence, how to chain tools, and how to synthesize evidence into a grounded answer. ODE addresses this at the data level by curating trajectories that demonstrate these behaviors, so that subsequent SFT and RL optimize the model on the desired interaction patterns rather than relying on inference-time prompting alone.

(3) Reusable state strengthens the harness. Before ODE training, replacing the standard agent workflow with our visual-native harness already improves the Qwen3-VL-30B-Instruct agent's average accuracy across the eight benchmarks from 24.8% to 30.6%. The largest relative improvements appear on visually grounded and search-intensive benchmarks such as HLE-VL, VDR and MMSearch+. This supports the core harness design: tool-produced images should not be treated as transient observations, but as persistent visual state that enables multi-step evidence construction.

3.3 Visual-Native Harness Ablation

This analysis evaluates the effectiveness of the proposed visual-native agent harness. We compare ODE-8B-RL under two harnesses. The full harness keeps every tool-returned image as an addressable `<image:N>` reference, allowing later tools to consume it. The ablated harness still shows tool-returned images to the model, but removes these reusable references, so intermediate images cannot be passed into later image-consuming tools. As shown in Fig. 3, we make the following observations:

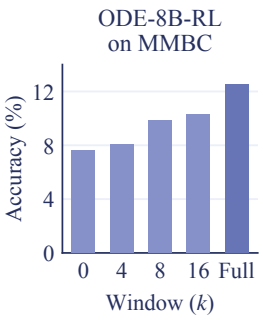
Reusable visual state improves performance. Fig. 3(a) shows that the full visual-native harness outperforms the ablated harness on key benchmarks. The effect is especially clear on MMBC, HLE-VL, and MMSearch+, where accuracy improves by $+4.9\%$, $+2.9\%$, and $+3.2\%$, respectively. Since the ablation keeps tool-returned images visible but removes their reusable references, these gains isolate the value of making intermediate visual evidence actionable across tool calls.

Reuse activation explains the gains. Fig. 3(b) shows that benchmarks with higher secondary image-reuse rates tend to benefit more from the full harness. MMBC has the highest reuse rate and also the largest gain, while HLE-VL and MMSearch+ show the same trend. This supports the intended interpretation of the ablation: the performance gap comes from making intermediate images reusable as later tool inputs, rather than from image visibility alone.

Reused images support visual refinement. Fig. 3(c) shows that reused images are mainly consumed by zoom-in and visual search. This indicates that the harness enables tool-produced images to be inspected, cropped, and searched again in later steps. In other words, image bank reuse turns intermediate visual outputs into working evidence for subsequent tool use.

Benchmark	Gemini 2.5 Flash	Gemini 2.5 Pro	GPT-5	ODE 8B-RL	ODE 30B-A3B-RL
MMBC	8.04	12.50	23.20	12.5	11.16
HLE-VL	10.53	19.30	19.90	11.4	10.53
BC-VL	39.10	45.10	58.60	41.9	46.10
VDR	18.80	26.40	23.40	20.4	26.40
MMSearch	59.33	61.00	68.00	66.0	69.67
MMSearch+	17.65	29.41	31.20	24.9	28.05
SVQA	69.67	74.67	68.30	70.3	71.00
FVQA	61.33	69.33	62.70	64.7	69.33
Avg.	35.55	42.21	44.41	39.0	41.53

(a) Matched-harness comparison



(b) Image bank window

Figure 4: Further analysis of the visual-native harness.

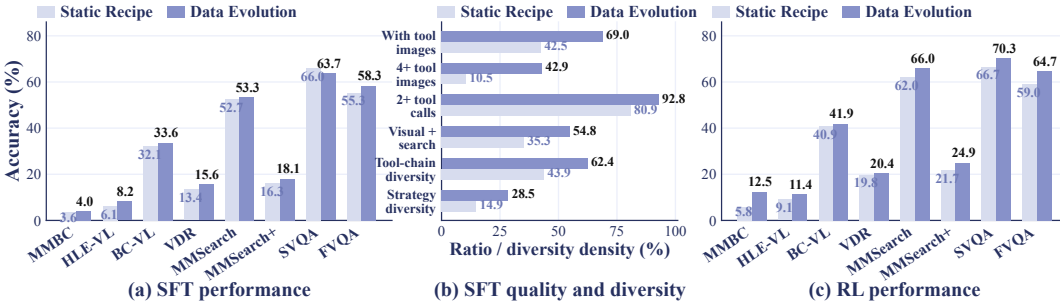


Figure 5: Static synthesis versus data evolution for agents based on Qwen3-VL-8B-Instruct.

3.4 Further Analysis of the Visual-Native Harness

To further assess the visual-native harness, we compare commercial models across workflows and use their performance under our harness as the basis for comparison with ODE (Fig. 4(a)). We then test whether restricting image reuse to recent tool outputs affects accuracy (Fig. 4(b)).

Matched-harness comparison. Switching from the standard agent workflow (Table 1) to our visual-native harness raises average accuracy across eight benchmarks from 32.0% to 35.55% for Gemini 2.5 Flash and from 37.9% to 42.21% for Gemini 2.5 Pro. GPT-5 scores higher on all seven benchmarks reported under both workflows. These gains demonstrate the effectiveness of our harness for multimodal search across the three commercial models tested. Under the same harness, our trained ODE-30B-A3B-RL is also competitive, averaging 41.53%, just 0.68 percentage points below Gemini 2.5 Pro.

Image bank window. We evaluate ODE-8B-RL on MMBC, the benchmark with the highest image reuse rate in Section 3.3. Accuracy rises as more tool-returned images remain available to later tools, increasing at each tested window size from 7.59% with reuse disabled to 12.50% with the full bank. Together with the reuse ablation in Section 3.3, this trend supports keeping images accessible throughout a search so that later tools can revisit earlier visual evidence.

3.5 Data Evolution vs. Static Synthesis

To verify whether the gains of ODE come from closed-loop data evolution rather than from scaling a fixed synthesis pipeline, we compare ODE with a static synthesis baseline that uses the initial ODE configuration and runs only the forward generation pipeline, without rollout-based analysis or configuration optimization. For SFT, we sample 2K traces from each source. For RL, we start from the same ODE-8B-SFT checkpoint and train with two 4K RL datasets, one produced by the evolved configuration and one produced by the static initial configuration. Fig. 5 reports both downstream performance and the SFT trace statistics.

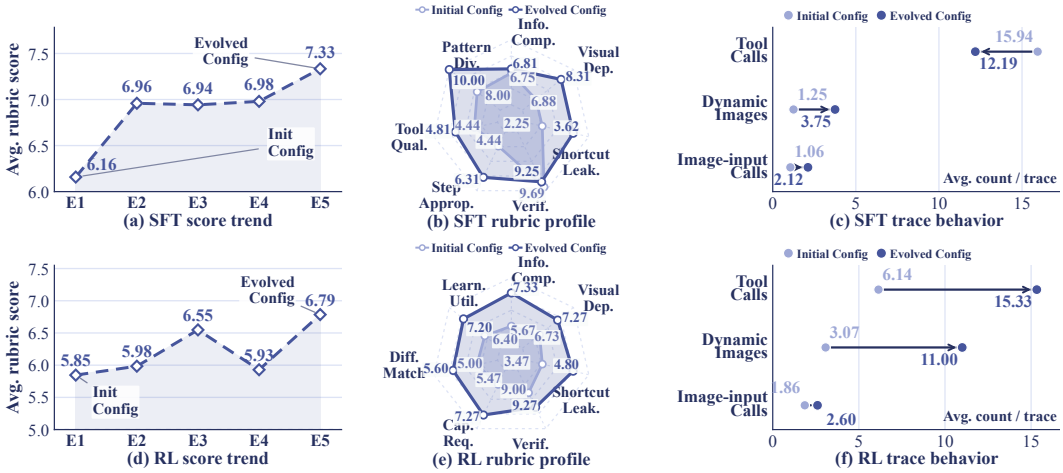


Figure 6: Mechanism analysis of ODE in SFT and 8B RL modes. Rubric scores are shifted by +5.

Evolution improves downstream SFT performance. Fig. 5(a) shows that SFT on evolved data outperforms the static recipe on most benchmarks, with clear gains on visually grounded and search-oriented evaluations such as HLE-VL, VDR, MMSearch+, and FVQA. This indicates that the forward pipeline alone can generate usable teacher traces, but feedback-driven evolution produces more effective imitation data under the same sample scale. The result supports the central role of ODE: the benefit is not merely from generating synthetic data, but from adapting the synthesis configuration using rollout feedback.

Evolved traces have higher quality and diversity. Fig. 5(b) explains why the evolved SFT data is more useful by comparing trace-level supervision patterns. *With tool images* measures the fraction of traces containing at least one intermediate tool-produced image beyond the original task image, while *4+ tool images* measures high-density visual supervision. *2+ tool calls* captures multi-step tool use, and *visual+search* measures whether a trace combines visual operations with search or browsing. Finally, tool-chain diversity counts distinct raw tool-call sequences, while strategy diversity groups these sequences into higher-level solving families. Compared with the static recipe, evolved traces contain more intermediate tool-produced images, a much larger fraction of high-density visual traces, more multi-step tool use, and more visual-search mixed strategies. The evolved subset also covers more distinct tool chains and broader abstract strategy families. Thus, ODE does not simply produce harder questions; it shifts the supervision distribution toward trajectories that demonstrate how to inspect visual evidence, combine tools, and solve tasks through richer agentic behavior.

Data evolution improves RL performance. Fig. 5(c) shows an even clearer pattern for RL. Starting from the same SFT checkpoint, RL on evolved data improves performance across the evaluated benchmarks, whereas RL on static data is weaker. This suggests that RL training benefits from tasks matched to the target policy's capabilities: a fixed initial recipe can generate tasks that are verifiable but poorly matched to the current policy's learning needs. ODE addresses this by using rollout feedback to move the generator toward tasks near the policy's learning frontier, making the resulting RL data more effective than static synthesis under the same data budget.

3.6 Mechanism Analysis of ODE

We further analyze what ODE changes during data construction. The goal is not only to show that evolved data performs better, but to understand how the closed-loop generator moves away from its initial configuration. We compare the initial and evolved configurations in both SFT and 8B RL modes. Fig. 6 reports rubric-score trends, rubric profiles, and rollout-level behavior statistics. E1–E5 denote evolution rounds. Here, *dynamic images* count tool-produced images acquired during rollout, excluding the original task image, and *image-input calls* count tool invocations that take an image reference as input. Agreement with independent trace analyzers is reported in Appendix B.6.

Evolution is mode-specific. The same ODE loop produces different changes for SFT and RL, matching their different data objectives. In SFT mode, the evolved configuration mainly improves

imitation-oriented dimensions such as visual dependency, step appropriateness, and tool-pattern diversity, while keeping verifiability high. This suggests that ODE does not simply make demonstrations longer or harder; it makes them better teacher traces. In RL mode, the improvements concentrate more on information complexity, capability requirement, difficulty match, and learning utility, indicating that the generator shifts toward tasks that are more suitable for policy improvement.

SFT traces become visually denser. The behavior statistics show that evolved SFT data uses fewer tool calls overall, but introduces more dynamic images and more image-input calls. This is an important distinction: the evolved demonstrations are not better because they are longer. They are better because more of the supervision is carried by intermediate visual evidence, and later tool calls are more likely to operate on those visual states. This matches the intended role of ODE for SFT: selecting trajectories that teach the model how to inspect, reuse, and integrate visual evidence rather than merely execute many tools.

RL tasks induce deeper search. For RL, evolution has a different behavioral effect. The evolved configuration induces rollouts with substantially more tool calls, more dynamic images, and more image-input calls. This indicates that ODE pushes the RL training task distribution toward examples that require active evidence gathering, rather than tasks solvable from the initial image or a single retrieval step. Together with the rubric improvements, this supports the central mechanism of ODE: rollout feedback steers the generator toward data that exposes the current policy's missing capabilities and provides a more useful training signal.

4 Related Work

Recent multimodal deep search agents extend MLLMs from static visual reasoning to interactive evidence gathering over web and tool environments. Early systems such as MMSearch [Jiang et al., 2025] and Vision Search Assistant [Zhang et al., 2024] equip MLLMs with image and web search, while newer benchmarks and agents emphasize harder settings that require fine-grained visual grounding, iterative retrieval, and provenance verification [Li et al., 2025, Tao et al., 2026, Geng et al., 2026b, Huang et al., 2026]. Recent training-oriented systems further improve multimodal search behavior with synthetic or semi-automated data and reinforcement learning. MMSearch-R1 [Wu et al., 2025] curates multimodal search VQA data for adaptive search, DeepMMSearch-R1 [Narayan et al., 2025] builds web-grounded search data with crop-conditioned image search, WebWatcher [Geng et al., 2026b] combines synthetic cold-start trajectories with RL, and Vision-DeepResearch [Huang et al., 2026] synthesizes long-horizon multi-tool trajectories under retrieval noise. Our approach differs from these systems in how it connects visual tool use with training data synthesis. In our visual-native harness, each tool-produced image has a persistent reference that later tools can use to retrieve it. ODE analyzes the resulting rollout traces and uses this feedback to adapt the synthesis recipe to the agent's current learning needs. See Appendix C for further discussion.

5 Conclusion

This paper presented a visual-native framework for multimodal deep search agents, combining a persistent image bank harness with On-policy Data Evolution (ODE). Across eight benchmarks, ODE-curated data consistently improves Qwen3-VL agents, increasing average accuracy from 24.9% to 39.0% at 8B and from 30.6% to 41.5% at 30B after SFT and RL. Further analyses show that reusable tool-generated images aid multi-step visual evidence gathering, while evolved data yields higher-quality and more diverse teacher traces than static synthesis. These findings highlight multimodal deep search as a co-design problem spanning the workspace, data generator, and training policy, and point to larger-scale on-policy evolution as a promising direction.

Acknowledgements

This research was financially supported by the Open Research Fund from Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ), under Grant No. GML-KF-26-06. This work was also partially supported by the National Natural Science Foundation of China No. 625B2178. The authors declare no competing interests.

References

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting reinforce style optimization for learning from human feedback in llms, 2024. URL https://arxiv.org/abs/2402.14740.

Anthropic. Claude 3.7 Sonnet System Card. Technical report, Anthropic, February 2025a. URL https://www-cdn.anthropic.com/9ff93dfa8f445c932415d335c88852ef47f1201e.pdf.

Anthropic. System Card: Claude Opus 4 & Claude Sonnet 4. Technical report, Anthropic, May 2025b. URL https://www-cdn.anthropic.com/6d8a8055020700718b0c49369f60816ba2a7c285.pdf.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

ByteDance Seed. Seed2.0 model card: Towards intelligence frontier for real-world complexity. https://lf3-static.bytednsdoc.com/obj/eden-cn/lapzild-tss/ljhwZthlaukjlkulzlp/seed2/0214/Seed2.0%20Model%20Card.pdf, 2026. Model card.

Center for AI Safety, Scale AI, and HLE Contributors Consortium. A benchmark of expert-level academic questions to assess AI capabilities. *Nature*, 649:1139–1146, 2026. doi: 10.1038/s41586-025-09962-4. URL https://arxiv.org/abs/2501.14249.

Shuang Chen, Kaituo Feng, Hangting Chen, Wenxuan Huang, Dasen Dai, Quanxin Shou, Yunlong Lin, Xiangyu Yue, Shenghua Gao, and Tianyu Pang. Opensearch-vl: An open recipe for frontier multimodal search agents, 2026. URL https://arxiv.org/abs/2605.05185.

Xianfu Cheng, Wei Zhang, Shiwei Zhang, Jian Yang, Xiangyuan Guan, Xianjie Wu, Xiang Li, Ge Zhang, Jiaheng Liu, Yuying Mai, Yutao Zeng, Zhoufutu Wen, Ke Jin, Baorui Wang, Weixiao Zhou, Yunhong Lu, Tongliang Li, Wenhao Huang, and Zhoujun Li. Simplevqa: Multimodal factuality evaluation for multimodal large language models, 2025. URL https://arxiv.org/abs/2502.13059.

Ethan Chern, Zhulin Hu, Steffi Chern, Siqi Kou, Jiadi Su, Yan Ma, Zhijie Deng, and Pengfei Liu. Thinking with generated images, 2025. URL https://arxiv.org/abs/2505.22525.

Guanting Dong, Junting Lu, Junjie Huang, Wanjun Zhong, Longxiang Liu, Shijue Huang, Zhenyu Li, Yang Zhao, Xiaoshuai Song, Xiaoxi Li, Jiajie Jin, Yutao Zhu, Hanbin Wang, Fangyu Lei, Qinyu Luo, Mingyang Chen, Zehui Chen, Jiazhan Feng, Ji-Rong Wen, and Zhicheng Dou. Agent-world: Scaling real-world environment synthesis for evolving general agent intelligence, 2026. URL https://arxiv.org/abs/2604.18292.

Gemini Team. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities, 2025. URL https://arxiv.org/abs/2507.06261.

Xinyu Geng, Xuanhua He, Sixiang Chen, Yanjing Xiao, Fan Zhang, Shijue Huang, Haitao Mi, Zhenwen Liang, Tianqing Fang, and Yi R. Fung. DeepSearch-World: Self-distillation for deep search agents in a verifiable environment. *CoRR*, abs/2607.07820, 2026a. doi: 10.48550/arXiv.2607.07820. URL https://arxiv.org/abs/2607.07820.

Xinyu Geng, Peng Xia, Zhen Zhang, Xinyu Wang, Qiuchen Wang, Ruixue Ding, Chenxi Wang, Jialong Wu, Kuan Li, Yida Zhao, Huifeng Yin, Yong Jiang, Pengjun Xie, Fei Huang, Huaxiu Yao, Yi R. Fung, and Jingren Zhou. Webwatcher: Breaking new frontiers of vision-language deep

research agent. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=8jsaazdAb3.

Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. Webvoyager: Building an end-to-end web agent with large multimodal models, 2024a. URL https://arxiv.org/abs/2401.13919.

Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Hongming Zhang, Tianqing Fang, Zhenzhong Lan, and Dong Yu. Openwebvoyager: Building multimodal web agents via iterative real-world exploration, feedback and optimization, 2024b. URL https://arxiv.org/abs/2410.19609.

Jack Hong, Chenxiao Zhao, ChengLin Zhu, Weiheng Lu, Guohai Xu, and Xing Yu. Deepeyesv2: Toward agentic multimodal model. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=yDKawwfJ5O.

Wenxuan Huang, Yu Zeng, Qiuchen Wang, Zhen Fang, Shaosheng Cao, Zheng Chu, Qingyu Yin, Shuang Chen, Zhenfei Yin, Lin Chen, Zehui Chen, Xu Tang, Yao Hu, Shaohui Lin, Philip Torr, Feng Zhao, and Wanli Ouyang. Vision-deepresearch: Incentivizing deepresearch capability in multimodal large language models, 2026. URL https://arxiv.org/abs/2601.22060.

Dongzhi Jiang, Renrui Zhang, Ziyu Guo, Yanmin Wu, Jiayi Lei, Pengshuo Qiu, Pan Lu, Zehui Chen, Guanglu Song, Peng Gao, Yu Liu, Chunyuan Li, and Hongsheng Li. Mmsearch: Unveiling the potential of large models as multi-modal search engines. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/04e6525463afaa460b5a2e6868b18f18-Abstract-Conference.html.

Guanyu Jiang, Zhaochen Su, Xiaoye Qu, and Yi R. Fung. XSkill: Continual learning from experience and skills in multimodal agents. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, pages 53078–53104. PMLR, 2026. URL https://proceedings.mlr.press/v306/jiang26aq.html.

Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Chong Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Ruslan Salakhutdinov, and Daniel Fried. Visualwebarena: Evaluating multimodal agents on realistic visual web tasks, 2024. URL https://arxiv.org/abs/2401.13649.

Shilong Li, Xingyuan Bu, Wenjie Wang, Jiaheng Liu, Jun Dong, Haoyang He, Hao Lu, Haozhe Zhang, Chenchen Jing, Zhen Li, Chuanhao Li, Jiayi Tian, Chenchen Zhang, Tianhao Peng, Yancheng He, Jihao Gu, Yuanxing Zhang, Jian Yang, Ge Zhang, Wenhao Huang, Wangchunshu Zhou, Zhaoxiang Zhang, Ruizhe Ding, and Shilei Wen. Mm-browsecomp: A comprehensive benchmark for multimodal browsing agents, 2025. URL https://arxiv.org/abs/2508.13186v1.

Shilong Liu, Hao Cheng, Haotian Liu, Hao Zhang, Feng Li, Tianhe Ren, Xueyan Zou, Jianwei Yang, Hang Su, Jun Zhu, Lei Zhang, Jianfeng Gao, and Chunyuan Li. Llava-plus: Learning to use tools for creating multimodal agents, 2023. URL https://arxiv.org/abs/2311.05437.

Weiwen Liu, Xu Huang, Xingshan Zeng, Xinlong Hao, Shuai Yu, Dexun Li, Shuai Wang, Weinan Gan, Zhengying Liu, Yuanqing Yu, Zezhong Wang, Yuxian Wang, Wu Ning, Yutai Hou, Bin Wang, Chuhan Wu, Xinzhi Wang, Yong Liu, Yasheng Wang, Duyu Tang, Dandan Tu, Lifeng Shang, Xin Jiang, Ruiming Tang, Defu Lian, Qun Liu, and Enhong Chen. Toolace: Winning the points of llm function calling, 2025a. URL https://arxiv.org/abs/2409.00920.

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

Arindam Mitra, Luciano Del Corro, Guoqing Zheng, Shweti Mahajan, Dany Rouhana, Andres Codas, Yadong Lu, Wei ge Chen, Olga Vrousgos, Corby Rosset, Fillipe Silva, Hamed Khanpour, Yash Lara, and Ahmed Awadallah. Agentinstruct: Toward generative teaching with agentic flows, 2024. URL https://arxiv.org/abs/2407.03502.

Kartik Narayan, Yang Xu, Tian Cao, Kavya Nerella, Vishal M. Patel, Navid Shiee, Peter Grasch, Chao Jia, Yinfei Yang, and Zhe Gan. Deepmmsearch-r1: Empowering multimodal llms in multimodal web search, 2025. URL https://arxiv.org/abs/2510.12801.

OpenAI. Gpt-4v(ision) system card. https://cdn.openai.com/papers/GPTV_System_Card.pdf, 2023. System card.

OpenAI. Update to GPT-5 System Card: GPT-5.2. https://deploymentsafety.openai.com/gpt-5-2, December 2025. OpenAI Deployment Safety Hub.

OpenAI. Openai gpt-5 system card, 2026. URL https://arxiv.org/abs/2601.03267.

Akshara Prabhakar, Zuxin Liu, Ming Zhu, Jianguo Zhang, Tulika Awalgaonkar, Shiyu Wang, Zhiwei Liu, Haolin Chen, Thai Hoang, Juan Carlos Niebles, Shelby Heinecke, Weiran Yao, Huan Wang, Silvio Savarese, and Caiming Xiong. Apigen-mt: Agentic pipeline for multi-turn data generation via simulated agent-human interplay, 2025. URL https://arxiv.org/abs/2504.03601.

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, 2024. URL https://arxiv.org/abs/2402.03300.

Zhaochen Su, Jincheng Gao, Hangyu Guo, Zhenhua Liu, Lueyang Zhang, Xinyu Geng, Shijue Huang, Peng Xia, Guanyu Jiang, Cheng Wang, Yue Zhang, Yi R. Fung, and Junxian He. Agentvista: Evaluating multimodal agents in ultra-challenging realistic visual scenarios, 2026. URL https://arxiv.org/abs/2602.23166.

Xijia Tao, Yihua Teng, Xinxing Su, Xinyu Fu, Jihao Wu, Chaofan Tao, Ziru Liu, Haoli Bai, Rui Liu, and Lingpeng Kong. MMSearch-plus: Benchmarking provenance-aware search for multimodal browsing agents. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=VGYgG2GH0d.

Tongyi DeepResearch Team, Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, Guangyu Li, Guoxin Chen, Huifeng Yin, Jialong Wu, Jingren Zhou, Kuan Li, Liangcai Su, Litu Ou, Liwen Zhang, Pengjun Xie, Rui Ye, Wenbiao Yin, Xinmiao Yu, Xinyu Wang, Xixi Wu, Xuanzhong Chen, Yida Zhao, Zhen Zhang, Zhengwei Tao, Zhongwang Zhang, Zile Qiao, Chenxi Wang, Donglei Yu, Gang Fu, Haiyang Shen, Jiayin Yang, Jun Lin, Junkai Zhang, Kui Zeng, Li Yang, Hailong Yin, Maojia Song, Ming Yan, Minpeng Liao, Peng Xia, Qian Xiao, Rui Min, Ruixue Ding, Runnan Fang, Shaowei Chen, Shen Huang, Shihang Wang, Shihao Cai, Weizhou Shen, Xiaobin Wang, Xin Guan, Xinyu Geng, Yingcheng Shi, Yuning Wu, Zhuo Chen, Zijian Li, and Yong Jiang. Tongyi deepresearch technical report, 2025. URL https://arxiv.org/abs/2510.24701.

Peng Wang, Qi Wu, Chunhua Shen, Anton van den Hengel, and Anthony Dick. Fvqa: Fact-based visual question answering, 2017. URL https://arxiv.org/abs/1606.05433.

Jinming Wu, Zihao Deng, Wei Li, Yiding Liu, Bo You, Bo Li, Zejun Ma, and Ziwei Liu. Mmsearch-r1: Incentivizing lmms to search, 2025. URL https://arxiv.org/abs/2506.20670.

Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal llms, 2023. URL https://arxiv.org/abs/2312.14135.

Shidong Yang, Ziyu Ma, Tongwen Huang, Yiming Hu, Yong Wang, and Xiangxiang Chu. Coevolve: Training llm agents via agent-data mutual evolution, 2026. URL https://arxiv.org/abs/2604.15840.

Yu Zeng, Wenxuan Huang, Zhen Fang, Shuang Chen, Yufan Shen, Yishuo Cai, Xiaoman Wang, Zhenfei Yin, Lin Chen, Zehui Chen, Shiting Huang, Yiming Zhao, Xu Tang, Yao Hu, Philip Torr, Wanli Ouyang, and Shaosheng Cao. Vision-deepresearch benchmark: Rethinking visual and textual search for multimodal large language models, 2026. URL https://arxiv.org/abs/2602.02185.

Kangning Zhang, Wenxiang Jiao, Kounianhua Du, Yuan Lu, Weiwen Liu, Weinan Zhang, and Yong Yu. Looptool: Closing the data-training loop for robust llm tool calls, 2025. URL https://arxiv.org/abs/2511.09148.

Zhixin Zhang, Yiyuan Zhang, Xiaohan Ding, and Xiangyu Yue. Vision search assistant: Empower vision-language models as multimodal search engines. *arXiv preprint arXiv:2410.21220*, 2024.

Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. Gpt-4v(ision) is a generalist web agent, if grounded, 2024a. URL https://arxiv.org/abs/2401.01614.

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E. Gonzalez, Clark Barrett, and Ying Sheng. Sglang: Efficient execution of structured language model programs, 2024b. URL https://arxiv.org/abs/2312.07104.

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.

Appendix

A Implementation Details of On-policy Data Evolution **16**

A.1 Round Configuration $\mathcal{C}_t$ . . . 16
A.2 Mode-Specific Trace Rubrics . . . 17
A.3 Stage 1: Seed Proposal . . . 18
A.4 Stage 2: Web Exploration . . . 18
A.5 Stage 3: Graph Organization . . . 19
A.6 Stage 4: Task Curation. . . . 20
A.7 Stage 5: Rollout and Stage 6: Backward Analysis . . . 21
A.8 Optimizer Update to $\mathcal{C}_{t+1}$ . . . 22
A.9 Round $t{+}1$ Forward Under $\mathcal{C}_{t+1}$ . . . 22
A.10 Round $t{+}1$ Backward and Update to $\mathcal{C}_{t+2}$ . . . 23
A.11 Additional Statistics of ODE-Curated Data . . . 23

B More on Experimental Setup **24**

B.1 Data Construction . . . 24
B.2 Training setup. . . . 24
B.3 Evaluation Setup . . . 24
B.4 Benchmark Details . . . 25
B.5 Train/Evaluation Overlap Audit . . . 26
B.6 Agreement with Independent Trace Analyzers . . . 27
B.7 Repeated-Inference Evaluation . . . 27

C More on related work **28**

C.1 Multimodal Web and Deep Search Agents . . . 28
C.2 Visual Tool Use and Thinking with Images . . . 28
C.3 Agentic Data Synthesis and Data Evolution . . . 28

D Limitations **29**

E Broader Impacts **29**

A Implementation Details of On-policy Data Evolution

Algorithm 1 summarizes the On-policy Data Evolution loop described in Section 2. This appendix traces two consecutive rounds from a real run. We first lay out the round's frozen System Config and starting Evolvable Config $\mathcal{C}_t$ (Appendix A.1), and the seven-dimension trace rubric the analyzer scores against (Appendix A.2). We then follow a single seed through the four forward stages of round t, namely seed proposal, web exploration, graph organization, and task curation (Appendices A.3–A.6), with the actual images decoded back from the trace. We rollout the curated task with the policy under training, score it with the rubric, and read off the per-stage diagnoses, which the optimizer aggregates into four targeted edits that produce $\mathcal{C}_{t+1}$ (Appendix A.8). To show the loop actually closes, we then run round $t{+}1$ on $\mathcal{C}_{t+1}$ with a new seed and walk through its rubric scoring and the resulting update to $\mathcal{C}_{t+2}$ (Appendix A.10), where the optimizer rolls back one of its round-t edits in response to the new failure mode, showing that ODE's edits are responses to the policy's current weak point rather than monotone refinements along a fixed direction.

Algorithm 1: On-policy Data Evolution

Require: Mode M, rubric R_M, config $\mathcal{C}_0$, rounds T, fixed System Config and π_{ref}
π_{ref}: teacher for SFT; pre-RL target checkpoint for RL

for $t = 0, \ldots, T-1$ **do**
  $S_t \leftarrow \textsc{SeedProposal}(\mathcal{C}_t)$
  $A_t \leftarrow \textsc{WebExploration}(S_t, \mathcal{C}_t)$
  $G_t \leftarrow \textsc{GraphOrganization}(A_t, \mathcal{C}_t)$
  $\mathcal{D}_t^{\text{cand}} \leftarrow \textsc{TaskCuration}(G_t, \mathcal{C}_t)$
  for each $x_i \in \mathcal{D}_t^{\text{cand}}$ **do**
    $\tau_i \leftarrow \textsc{Rollout}(\pi_{\text{ref}}, q_i, \mathcal{I}_{0,i})$
    $\delta_i \leftarrow \textsc{TraceAnalysis}(\tau_i, x_i, R_M)$
  end for
  $\Delta_t \leftarrow \textsc{Aggregate}(\{\delta_i\})$
  $P_t \leftarrow \textsc{ProposeEdits}(\mathcal{C}_t, \Delta_t)$
  $\mathcal{C}_{t+1} \leftarrow \textsc{AuditUpdate}(\mathcal{C}_t, P_t, \Delta_t)$
end for
Select and freeze best audited config $\mathcal{C}^*$
$\mathcal{D}_{\text{scale}} \leftarrow \textsc{ForwardGenerate}(\mathcal{C}^*)$
if $M = \text{SFT}$ **then**
  $\mathcal{U}_{\text{scale}} \leftarrow \textsc{Inputs}(\mathcal{D}_{\text{scale}})$
  $\boldsymbol{\tau}_{\text{scale}} \leftarrow \textsc{Rollout}(\pi_{\text{ref}}, \mathcal{U}_{\text{scale}})$
  $\mathcal{D}_{\text{train}} \leftarrow \textsc{RetainSFT}(\mathcal{D}_{\text{scale}}, \boldsymbol{\tau}_{\text{scale}})$
  $\pi_\theta \leftarrow \textsc{SFTTrain}(\mathcal{D}_{\text{train}})$
else
  $\pi_\theta \leftarrow \textsc{RLTrain}(\mathcal{D}_{\text{scale}})$
end if

A.1 Round Configuration $\mathcal{C}_t$

A round begins with two configuration objects, the System Config and the Evolvable Config $\mathcal{C}_t$. The System Config bundles every component that ODE deliberately freezes for the duration of an evolution run, so that backward refinement is comparing rounds under matched conditions. The Evolvable Config holds the four-stage generator parameters that the optimizer is allowed to edit between rounds. We give a sample of each below.

System Config (frozen for the run)

```
mode : rl
rollout_model : the policy currently under training, queried through the visual-native harness
judge : LLM judge that compares the rollout's final answer to the reference and emits a success or
failure label
analyzer : LLM analyzer that scores each rollout against the round's rubric and returns the
per-rollout diagnosis δ_i
rubric : RL rubric, 7 dimensions, weights [1.0, 1.2, 1.2, 1.0, 1.0, 2.0, 1.6]
tool_environment : nine harness tools (web_search, web_fetch, image_search, visual_search,
```

```
link_reader, zoom_in, rotation, flip, python_code)
seed_type : entity_with_image
sampling_axes : 11 domains, 4 ability profiles, 4 difficulty levels
```

$\mathcal{C}_t$, the round's Evolvable Config (key numerical fields)

```
seed_proposer.max_steps = 8
explorer.max_steps = 10
explorer.params.number_of_anchors = 6
explorer.params.max_nodes_per_phase = 2
explorer.params.image_ratio = 0.5
graph_organizer.complexity.reasoning_ratio = 0.33
graph_organizer.complexity.perception_ratio = 0.40
graph_organizer.complexity.reasoning_max_steps = 5
graph_organizer.complexity.perception_max_steps = 4
curator.few_shot_difficulty_weights = {easy: 0.05, medium: 0.15, hard: 0.50, expert: 0.30}
```

The key prompt fields used by the loop are embedded in the corresponding stage cases below. The forward-stage prompts belong to $\mathcal{C}_t$ and evolve with the numerical fields: the optimizer can append rejection rules, swap clauses, or rephrase strategy hints based on the stage diagnoses returned by backward refinement.

A.2 Mode-Specific Trace Rubrics

The analyzer scores each rollout along a seven-dimension rubric. Every dimension takes an ordinal score from -5 to $+5$ with a short textual justification, and the overall score $s(\delta_i)$ is the weighted average of the seven dimension scores using the weights listed below. The analyzer also returns a stage-level attribution field that names which of `seed_proposer`, `explorer`, `graph_organizer`, or `curator` is responsible for any observed failure. Four dimensions, namely *Information_Complexity*, *Visual_Dependency*, *Shortcut_Leakage*, and *Verifiability*, are shared across modes because they describe the task itself regardless of how it is consumed. The remaining three dimensions are mode-specific.

SFT Rubric: Teacher Trace Value

The SFT rubric favours teacher trajectories that are useful for distillation, rather than merely correct final answers. **Information_Complexity** (weight 1.0). Rewards traces teaching multi-source search, cross-reference, and synthesis. Rejects common-knowledge answers and one-query lookups. **Visual_Dependency** (weight 1.2). Rewards traces where visual interpretation is necessary and visible in the reasoning process. Rejects text-only tasks and irrelevant visual context. **Shortcut_Leakage** (weight 1.5). Rewards traces whose key cues must be actively discovered. Rejects examples where the answer is exposed by the question, an obvious image cue, or the first search snippet. **Verifiability** (weight 1.0). Rewards objective answers with a clear evidence-to-answer chain. Rejects subjective, ambiguous, or weakly judgeable conclusions. **Step_Appropriateness** (weight 1.2). Rewards traces whose number and type of steps match the task difficulty and ability profile. Rejects demonstrations that teach miscalibrated effort. **Tool_Usage_Quality** (weight 1.5). Rewards purposeful tool selection, chaining, verification, and stopping behaviour. Rejects circular calls, redundant tools, or missing critical operations. **Tool_Pattern_Diversity** (weight 3.0). A batch-level metric computed from complete tool-call sequences, encouraging diverse strategy types such as search-only, visual-search, computation-assisted, and multi-step visual workflows.

RL Rubric: Task Quality and Policy Suitability

The RL rubric favours tasks that are judgeable, evidence-supported, and useful for optimizing the current policy. **Information_Complexity** (weight 1.0). Rewards tasks requiring multi-source evidence integration. Rejects tasks answerable by memorization, common knowledge, or a single obvious query. **Visual_Dependency** (weight 1.2). Rewards tasks whose solving process depends on visual information from the input or tool-returned images. Rejects text-only solutions or misleading visual context. **Shortcut_Leakage** (weight 1.2). Rewards tasks whose key cues require effort to uncover and verify. Rejects tasks where the answer is leaked by the prompt, obvious image content, or search snippets. **Verifiability** (weight 1.0). Rewards tasks with unique, objective answers verifiable against authoritative evidence. Rejects subjective or under-specified tasks. **Capability_Requirement** (weight 1.0). Rewards tasks requiring multi-step reasoning, tool planning, and integration of visual and textual evidence. Rejects tasks solvable without meaningful tools or reasoning. **Difficulty_Match** (weight 2.0). Rewards tasks near the policy's learning zone. The analyzer maps each rollout to a five-stage difficulty diagnosis, ranging from tasks already solved with little effort, through under-challenging tasks and productive matches, to tasks beyond the policy's current capability or made hard by brittle evidence and infrastructure noise. The optimizer reads this distribution when steering batch-level difficulty. **Learning_Utility** (weight 1.6). Rewards tasks and traces with clear correctness signal, attributable errors, transferable strategies, and useful capability gaps. Rejects chaotic, misleading, or non-transferable traces.

A.3 Stage 1: Seed Proposal

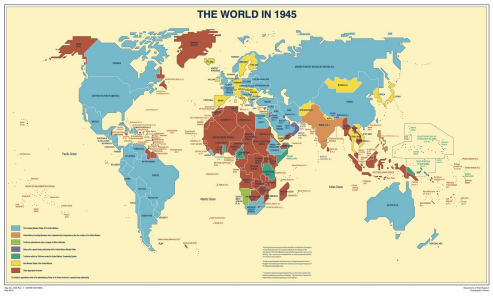


Figure 7: **Seed image $\mathcal{I}_0$.** The seed proposer samples an `entity_with_image` pair anchored on *United Nations Map No. 4135 Rev. 3, "The World in 1945"* (May 2010), domain `geography`.

Seed Record

Entity. United Nations Map No. 4135 Rev. 3: The World in 1945 (May 2010). **Domain.** `geography`. **Visual potential.** The map carries legible, visually extractable details, including the official numeric map identifier `4135 Rev. 3`, publication date `May 2010`, and the thematic legend categories for 1945 global territorial statuses, all of which require direct visual inspection rather than keyword recognition. **Information depth.** The seed supports multi-hop reasoning, such as cross-referencing the map number and revision against the UN cartographic archive, or validating the 1945 boundary classifications against independent historical geographic datasets, and rejects trivial single-snippet solutions. **Search queries.** `United Nations map "Map No." "Rev." jpg scale bar; 3 May 2010 United Nations cartographic section world in 1945`. **Source URLs.** Cornell University Press *UN Support for Decolonization* page; UN file repository at `un.org/id/file/59483`. **Judge decision.** Accepted. The judge confirms that the artifact carries information-bearing image content and supports multi-hop verifiable tasks.

Seed Proposal Prompt Fields

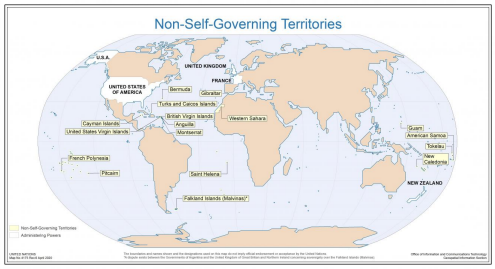
`strategy`**.** The seed proposer is instructed to find `entity_with_image` seeds for deep search tasks. A good seed must pair a real entity with an information-bearing image and enough factual surface area to support multi-hop verification and computation. The prompt prioritizes labeled maps, museum placards, technical diagrams, archival documents, posters, charts, timelines, and other images whose labels, dates, quantities, coordinates, or legends can be visually extracted. It also requires domain diversity across the seed batch and rejects decorative photos, common-knowledge trivia entities, paywalled images, and unstable social posts. `default_requirement`**.** The acceptance checklist requires `web_search`, `image_search`, and at least one `visual_search` call. The image must be relevant, high-resolution, and readable under zoom, and the entity or visible facts must be supported by at least two independent sources, preferably authoritative institutions such as museums, universities, governments, publishers, or recognized organizations. The seed must also support multi-hop lookup plus at least one reasoning operation, rather than a single-snippet answer. `seed_prompt`**.** The agent is asked to propose a small set of diverse multimodal seeds. For each candidate it searches for a factual entity, locates an information-bearing image, verifies that the image contains readable content, and finds an independent corroborating source. The required output is a YAML-style record containing `entity`, `entity_type`, `image_url`, `image_source_page`, `supporting_sources`, `why_visual`, `multi_hop_potential`, and `rejection_risks`.

A.4 Stage 2: Web Exploration

The explorer expands the seed into a small information network of six anchors, visiting twelve URLs over a single exploration phase. Anchors include the cartographic baseline, the September 1948 UN snapshot, the institutional pathway from the Trusteeship Council to the C-24 Special Committee on Decolonization, the modern Non-Self-Governing-Territories list, and territory-specific maps. Each anchor records textual facts, source URLs, and tool-returned image identifiers. Two of the anchor images are reproduced below.

Explorer Record

Topic. UN cartography of post-WWII territorial status. **Visited URLs.** 12 (UN Geospatial Information Section, UN Charter texts, Trusteeship Council documents, NSGT roster, Western Sahara reference page, Britannica, and Wikipedia event pages). **Anchors collected.** (1) UN Map No. 4135 Rev. 3, baseline 1945 world; (2) UN Map of UN Members and Dependent Territories, September 1948; (3) UN Trusteeship Council; (4) Special Committee on Decolonization (C-24); (5) UN Map No. 4175 Rev. 6, NSGT global map (April 2020); (6) UN Map No. 3175 Rev. 5, Western Sahara



(a) The contemporary NSGT global map, *UN Map No. 4175 Rev. 6 (April 2020)*.



(b) The territory-specific UN reference map for Western Sahara, *UN Map No. 3175 Rev. 5 (Jan 2020)*.

Figure 8: **Tool-returned anchor images** from the explorer. Each is appended to the image bank under a fresh `<image: N>` identifier and remains available to later stages and to the rollout policy.

(January 2020). **Quantitative facts attached.** The 11 "original" UN Trust Territories administered under Chapter XII; the September 1948 snapshot showing 10 Trust Territories already administered (Italian Somaliland's trusteeship begins only in 1950); 17 NSGTs listed in the 2020 NSGT roster.

Web Exploration Prompt Fields

`strategy`**.** The explorer is instructed to expand the seed into a compact knowledge graph for multimodal deep search QA generation. Each anchor must add a new direction, such as a related artifact or version, upstream cause, downstream impact, comparable peer entity, geographic context, dataset or report, or historical milestone. The prompt favors anchors with dates, measurements, counts, money, coordinates, legend structures, classification tables, competing claims, or cross-modal details that appear in images but not plainly in text. `quality_requirements`**.** Every node must be built from fresh tool use rather than memory. The prompt requires at least two distinct tool calls per node, a clear definition, filled detailed-explanation fields, six to ten key facts, at least three numeric or date facts when available, and at least two independent sources. For image-bearing nodes, the record must state exactly what can be extracted visually and why the image is useful downstream. `exploration_process_prompt`**.** The loop starts from the seed or previous node, identifies a promising next anchor, uses `web_search` to locate authoritative pages and alternatives, uses `link_reader` to extract precise details, and uses `image_search` plus `visual_search` when the anchor should satisfy the image-evidence target. Each node is written with title, type, definition, detailed explanation, key facts, images, sources, and source-support notes before the explorer chooses the next anchor. The prompt explicitly prefers fewer high-quality facts over long vague summaries.

A.5 Stage 3: Graph Organization

The graph organizer assembles the anchors into a multimodal evidence graph G and enriches it with reasoning and perception nodes that test cross-source consistency and surface fine-grained visual details for the curator to anchor on.

Organized Graph

Core themes. (1) UN cartographic documentation of territorial status, 1945 to 2020. (2) The shift from colonial dependencies and trust territories to the NSGT framework. (3) UN institutional oversight of decolonization, from the Trusteeship Council to the C-24 Special Committee. (4) Territory-specific case documentation (Western Sahara). **Edge structure.** Node 1 (1945 baseline map) provides the territorial-status framework that later UN maps extend. Node 2 (1948 UN snapshot) *supports* Node 3 by depicting Trust Territories under the Trusteeship Council. Node 3 *explains* the governance mechanism behind Node 2 and is functionally extended by Node 4 (C-24). Node 4 *exemplifies* the modern NSGT pathway and is realized as the cartographic product Node 5. Node 6 (Western Sahara) *supports* Node 5 through territory-specific cartographic detail. **Reasoning enrichment.** A consistency check between (a) the count of Trust Territories visible on the 1948 map and (b) the canonical roster of 11 originals attaches a derived quantity "$10/11 \approx 90.9\%$" to the edge connecting Node 2 and Node 3, with provenance to the Trusteeship Council page. The check fires under `graph_organizer.complexity.reasoning_ratio` = 0.33 in $\mathcal{C}_t$. **Perception enrichment.** A `zoom_in` pass over the 1948 map legend exposes the shading category for Trust Territories and is added back as a new perception node, and a second `zoom_in` over the Western Sahara map legend extracts the administering-power annotation. The pass fires under `graph_organizer.complexity.perception_ratio` = 0.40 in $\mathcal{C}_t$.

Graph Organization Prompt Fields

`organization_strategy`**.** The organizer is asked to analyze the relationships among collected information nodes and turn them into a coherent knowledge graph. The prompt receives each node's anchor, node type, core information, connections, and available images, then asks for themes, edges, source relations, entity or event relations, and cross-modal dependencies. `complexity.reasoning_strategies_prompt`**.** The reasoning agent is asked to create a reasoning-enhanced node through analysis or computation. It can call `python_code` and `visit`, must use at least one tool, and should add analytical value beyond restating the original node. The output schema records the reasoning type, process, tools used, findings, computation when applicable, source URLs, and the connection back to the original anchor. `complexity.perception_strategies_prompt`**.** The perception agent is asked to create a perception-enhanced node by operating on an image. It can call `zoom_in`, `flip`, `rotation`, and `visual_search`. The prompt emphasizes choosing transformations that reveal new information, such as small labels, legends, angled text, logos, species, or landmarks, and it requires the resulting visual finding to be documented as a new graph node rather than an informal note. `complexity.enhancement_requirements`**.** Both enhancement paths must produce nodes that are useful for later task curation. The enhanced node should have a clear definition, explanation, key facts, provenance, and explicit connections so that the curator can ground questions in the enriched graph.

A.6 Stage 4: Task Curation



Figure 9: **Curated task image** for the worked example. The image is the September 1948 UN snapshot, selected from the evidence graph as the visual anchor of the curated question. It is registered into the image bank as $\mathcal{I}_0$ before rollout.

Curated Task Record

Question. "In the world map shown (dated 'as of September 1948'), use the legend to identify which shaded regions count as UN-administered trust territories (not ordinary dependencies/colonies), and count how many distinct trust territories are actually depicted. Then compare that count to the full set of 'original' trust territories administered under the UN's trusteeship system by the UN organ responsible for trust territories. Be careful to exclude the Horn-of-Africa territory that is often confused with similarly named 'Somaliland' regions but whose trusteeship agreement started later (in 1950, so it would not appear as a trust territory on a September 1948 map). What percent of the original trust-territory set were already designated as trust territories by September 1948? Round to the nearest whole percent." **Ground truth.** `91%`. **Difficulty.** `hard` (sampled under $\mathcal{C}_t$.curator.few_shot_difficulty_weights with weight 0.50). **Ability profile.** `perception_reasoning`. **Visual dependency.** The map legend is required to distinguish trust territories from other dependencies, and the small on-map labels are required to count the distinct trust territories depicted. The count cannot be obtained reliably without reading the map. **Evidence chain.** The 1948 UN map's trust-territory legend category corresponds to 10 distinct labeled trust territories. The UN Trusteeship Council administered 11 original trust territories in total. The missing one relative to the originals is the former Italian-administered Somali territory whose trusteeship began only in 1950 and therefore does not appear on a September 1948 map, so the share is $10/11 \approx 90.9\%$, rounding to 91%. **Planned reasoning steps.** Nine steps mixing image_processing (read effective date, identify legend category, scan and label trust territories), web_search (retrieve the original-set count and the Somaliland exclusion), and calculate (form the percentage and round). Curator complexity-enhancement rewrote an earlier easier draft ("what percentage of original trust territories were already designated by 1948?") into the present form by introducing the explicit Somaliland-exclusion check and removing the verbatim total.

Task Curation Prompt Fields

`difficulty_control_prompt`**.** The curator is instructed to avoid trivial tasks, avoid brittle or unsolvable tasks, and prefer hard-but-learnable examples. The desired task should combine at least one key image fact, one web-only fact, and one synthesis, comparison, or computation step. Expert difficulty is encouraged only when the evidence path remains clean and judgeable. `strategy`**.** The task-generation prompt turns the evidence graph into diverse QA tasks. It requires

short, verifiable answers such as entities, numbers, dates, or names, and it discourages sentence-length ground truth. It also enforces multi-step tool use, specialized knowledge beyond common facts, diverse ability profiles, and question phrasing that asks what to find rather than how to find it. `quality_requirements_prompt`. The final self-check covers answer format, answer uniqueness, objective verifiability, complete evidence chain, natural image references, absence of numbered image indices, no tool-name or strategy leakage, specialized knowledge complexity, and batch diversity. The prompt bans questions that say "zoom in", "search for", "use OCR", "calculate by", or otherwise reveal the solving procedure. `complexity_enhancement.requirements_prompt`. The enhancement pass preserves the ground-truth answer while increasing reasoning depth. It keeps image references natural, avoids technical image tokens, removes strategy leakage, targets longer hard or expert reasoning chains, and ensures the enhanced question remains answerable from the graph evidence. `complexity_enhancement.strategy_prompt`. The rewrite strategies include removing obvious clues, replacing direct entity names with relationship-based descriptions, adding intermediate lookups, adding comparisons or computations, converting single-hop questions into multi-hop questions, and preserving visual dependency when the task is multimodal.

A.7 Stage 5: Rollout and Stage 6: Backward Analysis

The candidate task is rolled out by the policy under training. The verifier finds the final answer incorrect, and the analyzer then scores the rollout along the rubric and attributes observed failures back to specific forward stages.

Backward Analysis Prompt Fields

```
rubric_prompt: You are a professional data quality evaluator. Score the Agent execution trace
based on the rubric. For each dimension, read the question, ground truth, visual materials, complete
reasoning steps, tool calls, observations, and outcome, then return a score and explanation.
scoring_principles: evaluate QA quality through task design and execution process. Synthesize
evidence from the question, ground truth, visual materials, rollout steps, tool calls, observations,
and final answer.
Difficulty_Match rule: classify the rollout into the five-stage difficulty diagnosis from too simple
to too difficult or noisy. Do not reward long traces, repeated retries, or external failures by
themselves. High scores require productive struggle with concrete intermediate progress.
diagnosis_prompt: after rubric scoring, diagnose root causes using the forward construction record.
Attribute each issue to seed_proposer, explorer, graph_organizer, or curator, report severity, affected
dimensions, and a suggested config-side repair.
```

Rubric Scores (overall 3.82, cognitive friction `optimal`)

Information_Complexity (4.0). Multi-source synthesis of legend-based classification on the 1948 map plus the canonical 11-original Trusteeship set, with cross-checks for the 1950 Somaliland exclusion. **Visual_Dependency (5.0).** The depicted-count of trust territories on the September 1948 map is recoverable only by visual interpretation of shaded regions and labels. Text-only sources can give the original total (11) but not the map's depicted count. **Shortcut_Leakage (3.0).** The original-set total is one snippet away from any UN page, but the depicted count still requires careful map reading. The final percentage is not directly leaked. **Verifiability (5.0).** Single objective numeric target, rounded to whole percent. **Capability_Requirement (5.0).** Demands precise zoom and crop to read the legend and small territory labels, correct entity classification (trust vs. mandate vs. dependency), external verification of the originals list, and arithmetic with rounding. **Difficulty_Match (3.0, tag `[good_match]`).** Challenging but achievable. The agent failed, but on correctable issues (missed visual items, misclassification of South-West Africa as a trust territory), placing the task inside the policy's productive learning zone. **Learning_Utility (3.0).** Failure mode is clearly attributable to visual extraction quality and behaviour, namely prioritizing accurate legend reading over speculative web search and validating each shaded region against the legend before counting. The trace yields transferable training signal.

Stage Diagnoses (analyzer output, abridged)

seed_proposer (severity moderate). The seed did not robustly lock identity and provenance to a single canonical map artifact (map number, revision, date), so the downstream pipeline could drift to a different UN-themed 1948 map than the proposed seed. The seed also did not enforce a high-resolution image requirement, which contributed to later misclassifications when low-resolution copies were used. *Affected dimensions.* Difficulty_Match, Learning_Utility, Visual_Dependency, Verifiability.
explorer (severity severe). The explorer collected only six anchors with thin per-anchor evidence and did not produce zoom-validated visual extractables for the legend or small territory labels. *Affected dimensions.* Information_Complexity, Visual_Dependency, Capability_Requirement.
graph_organizer (severity moderate). The reasoning enrichment recorded the $10/11$ derivation but did not run a perception enrichment that would have surfaced the legend's shading categories as separate nodes. *Affected dimensions.* Visual_Dependency, Capability_Requirement.
curator (severity moderate). The question is well-formed, but its evidence chain depends on a low-resolution count that the rollout policy plausibly miscounts. *Affected dimensions.* Verifiability, Difficulty_Match.

A.8 Optimizer Update to $\mathcal{C}_{t+1}$

The optimizer aggregates the per-rollout diagnoses across the round's batch into a single Δ_t, applies the implied edits to the corresponding fields of $\mathcal{C}_t$, and writes out $\mathcal{C}_{t+1}$. The next-round config differs from $\mathcal{C}_t$ at exactly the four numerical fields below. The string-valued strategy and requirement prompts are updated in parallel by appending the analyzer's suggested rejection rules verbatim.

Optimizer Prompt Fields

```
role: You are a professional AI training data system architect. Based on rubric score analysis and
diagnosis feedback, generate precise, structured changes to the stage configuration.
inputs: rubric score table, dimension trends, bottom-tail failure modes, batch health, policy pass
rate, current stage config, available field paths, programmatic issue aggregation, stable fault
signals, cross-stage root-cause signal, and stage-local rubric responsibility.
actions: update_param for numerical or boolean parameters, append_text for new prompt clauses,
replace_text for exact-substring edits, and rewrite_text for consolidating an entire prompt field.
constraints: numerical edits are small, prompt edits must be surgical, patch sets should be compact,
parser noise is ignored, current-config faults are fixed before weak-reference drift, schema contracts
are preserved, and stable recurring failures are preferred over case-specific rules.
reviewer_prompt: a conservative reviewer filters proposed changes, keeping only edits directly
supported by recurring fault signals, compatible with the config contract, not case-specific
overfitting, and not unnecessary prompt bloat.
```

Diff $\mathcal{C}_t \to \mathcal{C}_{t+1}$ (numerical fields)

```
seed_proposer.max_steps : 8 → 10
explorer.params.image_ratio : 0.50 → 0.40
graph_organizer.complexity.reasoning_max_steps : 5 → 6
graph_organizer.complexity.perception_max_steps : 4 → 5
```

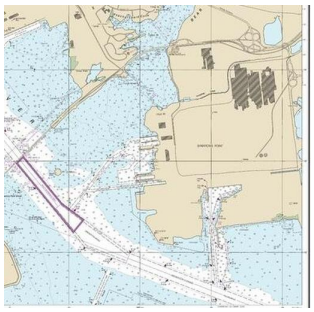
The four edits map cleanly onto the four stage diagnoses above. (1) Raising `seed_proposer.max_steps` from 8 to 10 gives the seed proposer enough budget to satisfy the new identity-and-provenance lock the optimizer appended to its `default_requirement`, addressing the seed-stage drift. (2) Lowering `explorer.params.image_ratio` from 0.50 to 0.40 trades off raw image-fetch frequency for tighter visual-evidence quality, paired with a separate prompt edit that requires a `visual_search` validation per quantitative anchor, addressing the explorer's thin per-anchor evidence. (3,4) Raising both reasoning and perception `max_steps` by one in the graph organizer gives complexity enhancement room to attach legend-category nodes and run an extra cross-source consistency check, addressing the missing perception enrichment. The curator field set is left unchanged, because the curator's diagnosis is a downstream consequence of the explorer and graph-organizer issues that the optimizer now addresses upstream. The batch-level pass-rate signal also leaves `curator.few_shot_difficulty_weights` unchanged, since the `[too_hard]` share is below the threshold that would trigger a difficulty-weight shift.

A.9 Round $t{+}1$ Forward Under $\mathcal{C}_{t+1}$

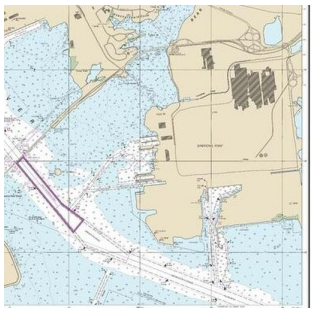
To show that the loop actually closes, we walk through the next round on $\mathcal{C}_{t+1}$. The same optimizer that produced $\mathcal{C}_{t+1}$ now drives a new seed through the four forward stages, and the rubric scores expose a different failure mode that the next update will then address.

Round $t{+}1$ Forward (compact)

Seed. `entity_with_image` pair. Entity *NOAA Nautical Chart 12281: Baltimore Harbor, 57th Edition (November 2018)*, domain `geography`. The seed proposer now lists the chart's image-visible identifiers explicitly (chart number `12281`, edition `57th Ed.`, publication date `Nov. 2018`, cancellation date `Mar 6, 2024`) and pairs them with two source URLs that include the original NOAA-hosted PDF. **Explorer.** 13 visited URLs over a single phase, surfacing 7 anchors. Anchors include the chart itself, the federally authorized Baltimore Harbor channel network, the USACE maintenance dredging program, the Seagirt Marine Terminal, double-stack intermodal rail, the FY2025 contract opportunity record on `sam.gov`, and the USACE district news release announcing the FY2025 work. **Graph.** Core themes are NOAA charting for Baltimore Harbor, federal navigation channel maintenance, and the Port of Baltimore cargo capacity. Edge structure links Node 1 (Chart 12281) to Node 3 (Baltimore Harbor Federal Navigation Channels) which is *maintained by* Node 4 (USACE program), specialized to Node 7 (FY2025 fiscal year implementation). Reasoning enrichment attaches the derived *average cost per cubic yard* to the edge between the contract value ($33.5 M) and the dredge quantity (2.3 M cy). **Curator.** The curator emits the dredging task, with difficulty `expert` and ability profile `perception_search_reasoning`, both raised relative to round t.



Round $t{+}1$ exploration anchor 4: zoomed segment of the Seagirt Marine Terminal access channel from the NOAA chart, surfaced by the explorer's tool-returned image pass.



Round $t{+}1$ task image $\mathcal{I}_0$: the curated chart excerpt showing the purple-outlined deep-draft channel reach adjacent to the Seagirt Marine Terminal.

Figure 10: **Round $t{+}1$ visual artifacts**, produced under the updated $\mathcal{C}_{t+1}$. The graph organizer has higher reasoning and perception step budgets in this round, and the curator anchors the question on a fine-grained channel reach rather than a coarse legend category.

A.10 Round $t{+}1$ Backward and Update to $\mathcal{C}_{t+2}$

The round $t{+}1$ candidate is rolled out by the same policy and scored by the same rubric. The new failure mode is informative.

Round $t{+}1$ Rubric Scores (overall 2.60, cognitive friction `optimal`)

Information_Complexity (5.0). Multi-step disambiguation among similarly named USACE FY2025 dredging efforts plus official-source verification of contract value and dredge quantity, then cost-per-yard arithmetic. **Visual_Dependency (5.0).** The chart's federal-channel styling and the adjacent terminal label are required to fix the program identity. **Verifiability (5.0).** Single objective dollars-and-cents target. **Capability_Requirement (5.0).** Demands chart reading, strategic web retrieval, disambiguation, and arithmetic. **Difficulty_Match (-3.0, tag `[too_hard]`).** The task landed *outside* the policy's productive learning zone. The agent never reliably extracted the terminal name and never reached the FY2025 contract record, looping on zoom and visual-search attempts and ending without the required numeric computation. **Learning_Utility (3.0).** The failure is correctable and clearly attributable, so the trace still yields transferable training signal.

Diff $\mathcal{C}_{t+1} \to \mathcal{C}_{t+2}$ (numerical fields)

```
explorer.params.max_nodes_per_phase :  2 → 1
explorer.params.image_ratio :  0.40 → 0.50   (rolled back)
graph_organizer.complexity.reasoning_max_steps :  6 → 7
graph_organizer.complexity.perception_max_steps :  5 → 6
```

The `[too_hard]` tag dominates the round's batch-level signal. The optimizer reads this as a request to slow down the explorer (`max_nodes_per_phase` drops from 2 to 1, forcing a deeper traversal of fewer per-phase nodes) and to give the graph organizer more enrichment headroom (both `reasoning_max_steps` and `perception_max_steps` go up by one). The optimizer also *rolls back* the round-t edit on `explorer.params.image_ratio` from 0.40 to 0.50, since the round-$t{+}1$ failure traces back to insufficient image-bearing evidence per anchor rather than to fetch volume. This rollback illustrates the on-policy character of ODE. Edits are not monotone refinements of a fixed direction. They are responses to whichever failure mode the policy currently exposes, and the optimizer is free to revisit a prior decision once the rollouts under that decision show it was the wrong move.

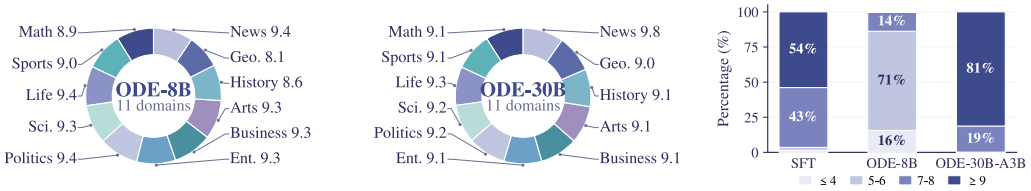
A.11 Additional Statistics of ODE-Curated Data

This appendix complements Fig. 2 in Section 2.3 with the ODE-8B and ODE-30B topic domain donuts and the planned reasoning-step distribution.

Topical breadth on the RL task sets. The ODE-8B and ODE-30B shown in Fig. 11(a, b) span the same eleven domains as the SFT demonstration set in Fig. 2a. ODE-30B confines per-domain shares to a narrow 8.43%–10.03% band with a coefficient of variation of 0.05, and ODE-8B falls within a

comparable band. The forward exploration stage therefore preserves topical coverage across SFT and policy-specific RL data construction, even though the difficulty distribution is allowed to shift between them.

Planned reasoning-step distribution tracks policy capacity. The reasoning-step buckets in Fig. 11(c) read from left to right as a clear depth ladder. ODE-8B concentrates at 5–6 steps with 70.58% of tasks in that bucket, ODE-30B pushes out to ≥ 9 steps with 81.22%, and the SFT demonstration set sits at the deep end with an average of 8.47 steps inherited from the teacher. The curator's planned-step field therefore tracks each retention's intended trajectory depth, scaling back to shorter plans when the targeted policy cannot sustain long ones and lengthening them when the policy can.



(a) (b) (c)

Figure 11: **Additional statistics of ODE-curated data.** Topic domain donuts for the two RL task sets and the planned reasoning-step distribution across the SFT demonstration set and the two RL task sets.

B More on Experimental Setup

B.1 Data Construction

We use ODE to curate both SFT and RL training data. GPT-5.2 [OpenAI, 2025] is used for most generation, analysis, and optimization stages, and also serves as the SFT rollout policy; in RL mode, the policy being trained is used for task-verification rollouts. We initialize the evolvable configuration with GPT-5.2 and set the maximum number of evolution steps to 5, with each step using 32 curated tasks and verified traces for rubric-guided configuration updates. We then freeze the selected configuration for large-scale synthesis, yielding 8,855 filtered SFT examples and two RL datasets of 4,000 examples for each model size.

B.2 Training setup.

We instantiate our agent with two Qwen3-VL backbones: Qwen3-VL-8B-Instruct and Qwen3-VL-30B-A3B-Instruct [Bai et al., 2025]. We first perform SFT on data curated by the SFT mode of ODE. For both backbones, SFT uses a maximum sequence length of 64k tokens, a global batch size of 64, a learning rate of 2×10^{-5}, and 2 training epochs. Starting from the SFT checkpoints, we further refine the agents with reinforcement learning. Following Huang et al. [2026], Chen et al. [2026], we use Group Relative Policy Optimization (GRPO) [Shao et al., 2024] with the leave-one-out trick [Ahmadian et al., 2024]. RL is conducted in our visual-native agent harness with asynchronous SGLang [Zheng et al., 2024b] rollouts, sampling 6 responses per prompt. We use a batch size of 96, an actor learning rate of 2×10^{-6}, a clip ratio of 0.28, and no KL regularization. All training experiments were conducted on NVIDIA H20 GPUs.

B.3 Evaluation Setup

All models and agents are evaluated under the same decoding and interaction budget. We use temperature 0.6 and top-p 0.95, and allow at most 50 LLM calls, 8,192 tokens per turn, and 16,000 tokens in total. To ensure consistent assessment across models and settings, we evaluate all predictions with the same LLM-as-judge verifier.

The verifier computes the terminal answer-level reward for each trajectory. Given a task instance, we first extract the candidate final answer from the model response. The judge is then given the question,

the reference answer, the extracted candidate answer, and the full model response as auxiliary context. It is instructed to assess correctness with respect to the reference answer, emphasizing semantic equivalence rather than exact surface-form matching. The criteria accept paraphrases, standard abbreviations, harmless formatting variations, entity-name variants, and mathematically equivalent numeric expressions, while rejecting answers that are ambiguous, incomplete, contradictory, unrelated, or that mention the reference answer only in an invalid context. The judge returns a structured JSON object containing a binary correctness decision, an equivalence category, and a short rationale. We use the binary decision as the final trajectory reward.

Prompt for LLM-as-judge

You are a precise final-answer judge. Your job is to decide whether the candidate final answer matches the correct answer. Be conservative, but grade like a careful human evaluator rather than a brittle string matcher. Do not require the candidate to copy the reference wording if it clearly gives the same answer.
[Question] question
[Correct Answer] {reference_answer}
[Candidate Final Answer] {candidate_answer}
[Full Model Response For Context] {full_response}
Task: Determine whether the candidate final answer is correct.
Decision rules: 1. Judge the candidate final answer itself, not the quality or completeness of the explanation. 2. A short answer can still be fully correct. Do NOT require supporting reasoning, derivations, citations, or extra context. 3. If the candidate final answer exactly matches the correct answer after normalizing case, whitespace, simple punctuation, commas, or trailing zeros, you MUST mark it correct. 4. First identify what kind of answer the question asks for: a named concept/entity, a definition, a purpose/ function, a mechanism/explanation, a date/date range, a number, an identifier, a location, or a list/set of items. 5. Accept direct semantic equivalence when the candidate gives the same answer in a different surface form. 6. Accept term-definition equivalence if the candidate uniquely names the same concept as the reference definition, or uniquely gives the defining description of the referenced term. 7. Accept a more specific but non-contradictory answer when it still directly answers the question. 8. Accept harmless category or wording variation when it preserves the same core meaning. 9. Accept a correct answer embedded inside a full sentence if the sentence directly answers the question and is not contradicted by other content. 10. For event questions, accept a scene description if it clearly identifies the same event. 11. For cause, reason, or purpose questions, the stated cause must match; a related but different explanation is wrong. 12. For broad class questions, a correct subtype is acceptable. 13. For numbers, dimensions, and formulas, accept mathematically equivalent forms and reasonable decimal approximations when they clearly refer to the same value. 14. For dates or date ranges, every required boundary must match. If the reference is only a year, a full date within that year is acceptable. 15. For named entities, a different entity is wrong even if it is similar, related, or from the same family. 16. Accept standard abbreviated company names and obvious person-name variants when they unambiguously identify the same entity. 17. For open-set questions signaled by wording such as "some examples" or "for example", accept alternate valid examples. 18. For closed-set questions asking for an exact set of items, extra incorrect items or missing required items are wrong. 19. If the candidate gives only a related topic, broad discussion, or background context instead of the answer itself, mark no. 20. If the candidate answer is ambiguous, hedged, contradictory, missing required specificity, or not an answer, mark no. 21. If the candidate says it could not solve the task, refuses to answer, gives no answer, or only shows tool traces or search queries, mark no. 22. If the candidate merely mentions the reference phrase inside a negated statement, failure trace, or quoted query, mark no.
Calibration examples: - Correct: reference='A set of edges without common vertices.' candidate='matching' - Correct: reference='Thought experiments.' candidate='used to explore philosophical questions about perception and reality' - Correct: reference='Saxbys coffee shop' candidate='The Saxbys location at the University of Pennsylvania...' - Correct: reference='Barcelona vs Inter Milan Champions League match' candidate='The image shows Lamine Yamal celebrating during the Barcelona vs Inter Milan Champions League semi-final.' - Correct: reference='2025' candidate='April 17, 2025' - Correct: reference='A snake' candidate='Gary, a blue pit viper' - Correct: reference='Anker Innovations' candidate='Anker' - Correct: reference='Clem Delangue' candidate='Clément Delangue' - Correct: reference='log(2)/log(3)' candidate='0.6309' - Wrong: reference='April 22 - 29' candidate='April 23 to April 29, 2025' - Wrong: reference='The HIVE Evo' candidate='The HIVE - Modular Hex Drawers' - Wrong: reference='precautionary checks after a gruelling bout' candidate='severe dehydration from weight cut' - Wrong: reference='1500 light-years' candidate='1375 light-years' - Wrong: reference='the Ocean's trilogy' candidate='Ocean's Eleven' - Wrong: reference='Ex Machina' candidate='About Time' - Wrong: reference='Canvas art prints' candidate='giclee prints' - Wrong: reference='Martha' candidate='None of the characters... Martha'
Return ONLY a single-line JSON object with no markdown fences and no extra text: {"correct":"yes"|"no","equivalence":"exact"|"format"|"semantic"|"wrong"|"missing"|"ambiguous","reason":"one short sentence"}

B.4 Benchmark Details

We provide the detailed evaluation benchmark below. Unless otherwise specified, we evaluate on the released benchmark questions with our unified agent harness and LLM-based answer judge.

MMBC. MM-BrowseComp (MMBC) [Li et al., 2025] is a multimodal browsing benchmark designed to test whether agents can retrieve and reason over web evidence that may appear in images or videos rather than text alone. Its questions are hand-crafted to require multi-hop multimodal browsing, and each item includes fine-grained reasoning requirements for checking multimodal dependency. We evaluate on the released MMBC evaluation set.

HLE-VL. HLE-VL is the visual-language subset of Humanity's Last Exam [Center for AI Safety et al., 2026], an expert-level academic benchmark with broad subject coverage and questions designed to be unambiguous, verifiable, and difficult to answer through shallow retrieval. We use HLE-VL to measure whether multimodal agents can combine visual interpretation with specialized academic reasoning.

BC-VL. Introduced by WebWatcher [Geng et al., 2026b], BC-VL extends BrowseComp-style hard browsing tasks to the visual domain. The benchmark contains long, entity-obfuscated multimodal questions that require cross-modal inference, web search, browsing, and planning rather than direct perception alone. Following prior work, we evaluate on the full BC-VL split.

VDR. VDR-Bench [Zeng et al., 2026] evaluates multimodal deep-research agents under long-horizon visual and textual search. It emphasizes multi-turn, multi-entity, and multi-scale evidence gathering, making it especially relevant for testing whether an agent can combine visual retrieval, textual search, and iterative reasoning. We use the test-mini split.

MMSearch. MMSearch [Jiang et al., 2025] evaluates whether large multimodal models can act as multimodal search engines. It contains manually curated queries spanning news and rare-knowledge domains, requiring models to retrieve external evidence rather than answer from parametric knowledge alone. We evaluate on all VQA instances in MMSearch.

MMSearch+. MMSearch+ [Tao et al., 2026] is a provenance-aware multimodal browsing benchmark designed to require fine-grained visual cue extraction, iterative image-text retrieval, and cross-validation under retrieval noise. We evaluate on the single-image subset, following the setting used in prior multimodal deep search evaluations.

SimpleVQA. SimpleVQA [Cheng et al., 2025] evaluates factuality in multimodal question answering. Its examples focus on short, factual visual questions where the answer should be grounded in reliable visual or world knowledge. We randomly sample 300 examples for evaluation.

FVQA. FVQA [Wang et al., 2017] is a fact-based visual question answering benchmark where answering requires external factual knowledge in addition to image understanding. Each question is associated with supporting facts, making it useful for evaluating knowledge-grounded visual reasoning. We randomly sample 300 examples for evaluation.

Following prior evaluation practice where applicable [Geng et al., 2026b, Huang et al., 2026], we use the test-mini split of VDR, the full split of BC-VL, all VQA instances in MMSearch, MMBC, HLE-VL, and the single-image subset of MMSearch+. For SimpleVQA and FVQA, we randomly sample 300 instances from each benchmark.

B.5 Train/Evaluation Overlap Audit

Seed entities were selected without using benchmark questions, answers, or entity lists. Web exploration also blocked Hugging Face and benchmark source-project URLs. We audited the SFT dataset and both RL datasets against the eight evaluation sets, comprising 2,586 questions and 2,371 images. Table 2 summarizes the checks. All 51 near-duplicate image candidates identified by pHash were false positives upon manual review. Our audit found no evidence of benchmark contamination in the ODE training data.

Table 2: Train/evaluation overlap audit across all three training datasets and eight evaluation sets.

Check	Method	Candidates	Confirmed
URLs	Exact intersection	0	0
Questions/answers	Exact + normalized	0	0
Near-duplicate questions	Jaccard ≥ 0.70, ≥ 0.80	0	0
Exact images	Decoded-image SHA-256	0	0
Near-duplicate images	pHash ≤ 8 + manual review	51	0

B.6 Agreement with Independent Trace Analyzers

GPT-5.5 and Claude Opus 4.8 independently rescored all archived ODE-SFT and ODE-RL-8B traces using the same mode-specific rubrics. Table 3 reports Pearson correlations with the original GPT-5.2 scores. Overall correlations range from 0.790 to 0.837, supporting agreement in ODE's internal trace diagnostics across analyzers. Agreement across independent analyzers suggests that ODE's trace diagnostics do not depend strongly on the choice of scoring model.

Table 3: Pearson correlations with GPT-5.2 for independent rescoring of archived traces. Dashes mark dimensions not applicable to the corresponding mode.

	GPT-5.5		Claude Opus 4.8	
Rubric dimension	SFT	RL-8B	SFT	RL-8B
Overall	0.816	0.837	0.790	0.814
Information Complexity	0.868	0.836	0.893	0.734
Visual Dependency	0.832	0.827	0.786	0.800
Shortcut Leakage	0.819	0.711	0.748	0.717
Verifiability	0.857	0.706	0.786	0.796
Step Appropriateness	0.890	—	0.748	—
Tool Usage Quality	0.895	—	0.768	—
Capability Requirement	—	0.831	—	0.699
Difficulty Match	—	0.725	—	0.696
Learning Utility	—	0.767	—	0.707

B.7 Repeated-Inference Evaluation

We performed two additional independent inference runs for each final RL checkpoint, giving three runs including the original evaluation reported in Table 1. Table 4 reports the mean and standard deviation across these three runs. The eight-benchmark average accuracy is $38.83 \pm 0.17\%$ for ODE-8B-RL and $41.78 \pm 0.23\%$ for ODE-30B-A3B-RL. For both ODE models, average benchmark performance remains stable across repeated inference runs with the checkpoints held fixed.

Table 4: Accuracy (%) as mean $\pm$ standard deviation over three inference runs of fixed checkpoints. Avg. summarizes the average accuracy over all eight benchmarks.

Benchmark	ODE-8B-RL	ODE-30B-A3B-RL
MMBC	12.05 ± 0.45	11.46 ± 0.51
HLE-VL	11.21 ± 0.61	11.41 ± 0.77
BC-VL	41.52 ± 0.39	46.69 ± 0.81
VDR	20.53 ± 0.42	25.60 ± 0.69
MMSearch	65.11 ± 1.02	69.22 ± 0.39
MMSearch+	24.43 ± 0.46	27.45 ± 0.69
SVQA	70.55 ± 0.39	72.00 ± 0.88
FVQA	65.22 ± 0.69	70.44 ± 0.96
Avg.	38.83 ± 0.17	41.78 ± 0.23

C More on related work

C.1 Multimodal Web and Deep Search Agents

Interactive multimodal agents have been studied in two adjacent settings. In visual web navigation, agents use rendered webpages, screenshots, accessibility trees, or browser actions to complete real-world web tasks. SeeAct [Zheng et al., 2024a] studies GPT-4V as a generalist web agent and highlights the importance of grounding visual plans into executable browser actions. WebVoyager [He et al., 2024a] builds an end-to-end multimodal web agent for real websites, and OpenWebVoyager [He et al., 2024b] further explores iterative real-world exploration, feedback, and optimization for open multimodal web agents. VisualWebArena [Koh et al., 2024] provides a realistic evaluation environment for web tasks that require both visual perception and browser interaction.

Multimodal deep search agents focus more directly on collecting and verifying external evidence through search, browsing, and visual tools. MMSearch [Jiang et al., 2025] and Vision Search Assistant [Zhang et al., 2024] connect MLLMs to multimodal search pipelines. MMSearch-R1 [Wu et al., 2025] studies adaptive image and text search through reinforcement learning. DeepMMSearch-R1 [Narayan et al., 2025] extends this direction with multi-turn query refinement and crop-conditioned image search. WebWatcher [Geng et al., 2026b] combines synthetic cold-start trajectories with reinforcement learning for vision-language deep research, and Vision-DeepResearch [Huang et al., 2026] scales long-horizon visual-textual search with multiple tools. Together, these works show that multimodal search benefits from richer interaction environments, stronger grounding, and training data that demonstrates multi-step evidence gathering. Our visual-native harness builds on this direction by making the multimodal search workspace explicit: search, browsing, visual manipulation, and computation operate over a shared image bank rather than over isolated tool outputs.

C.2 Visual Tool Use and Thinking with Images

A complementary line of work studies how multimodal models can actively manipulate, revisit, or construct visual information during reasoning. LLaVA-Plus [Liu et al., 2023] integrates external visual tools into multimodal assistants, while XSkill [Jiang et al., 2026] uses past trajectories to guide tool selection on new tasks. V* [Wu and Xie, 2023] introduces LLM-guided visual search for high-resolution and visually crowded images, emphasizing selective visual attention and region-level grounding. More recently, Visual-ARFT [Liu et al., 2025b] trains LVLMs to use web browsing and code-based image manipulation tools through reinforcement fine-tuning, while DeepEyes [Zheng et al., 2025] incentivizes interleaved visual and textual reasoning through reinforcement learning. Thinking with Generated Images [Chern et al., 2025] explores a related but distinct direction, where models generate and refine intermediate visual states as part of reasoning. These works collectively move multimodal reasoning from passive image understanding toward active visual computation. This trend is closely aligned with multimodal deep search, where useful visual evidence is often discovered, transformed, or localized over multiple tool calls rather than provided only in the original input image. This motivates our treatment of intermediate visual observations as reusable working memory, so that visual evidence discovered or transformed in one step can continue to support later search and reasoning.

C.3 Agentic Data Synthesis and Data Evolution

Synthetic data has become a major driver for training tool-using and agentic language models. AgentInstruct [Mitra et al., 2024] uses agentic flows to generate diverse post-training data from raw documents and code. ToolACE [Liu et al., 2025a] constructs function-calling data through self-evolution, multi-agent dialogue generation, and verification. APIGen-MT [Prabhakar et al., 2025] generates multi-turn agent data from task blueprints and simulated human-agent interaction. These methods show that agentic synthesis can produce scalable supervision for tool use, function calling, and multi-turn interaction. Recent work has also begun to close the loop between agent behaviour and data construction [Geng et al., 2026a]. LoopTool [Zhang et al., 2025] uses model-aware probing, label verification, and error-driven data expansion to improve LLM tool calling. CoEvolve [Yang et al., 2026] studies mutual evolution between agent policies and data in language-agent environments. Similarly, Agent-World [Dong et al., 2026] proposes a self-evolving training arena that dynamically synthesizes real-world environments and tasks to identify capability gaps, enabling the continuous co-evolution of agent policies and tool environments. Tongyi DeepResearch [Tongyi DeepResearch

Team et al., 2025] demonstrates the importance of large-scale automatic data synthesis for long-horizon research agents. These works motivate the broader view that agent training data should adapt to model behaviour, especially when the target capability involves long-horizon interaction rather than single-turn prediction. ODE extends this adaptive-data perspective to multimodal deep search, where rollout feedback must capture not only answer correctness but also visual grounding, tool-use structure, and policy-specific learning value.

D Limitations

This work has three main limitations. First, ODE introduces an additional pre-computation stage before large-scale training data synthesis. In our experiments, each evolution epoch uses only 32 rollout traces for feedback, but the repeated generate–rollout–analyze–update loop still consumes wall-clock time and API resources because it invokes web tools, visual tools, LLM-based analysis, and configuration optimization. This cost is amortized once the evolved configuration is frozen for large-scale synthesis, but improving the efficiency of the evolution loop remains important. Future work could study cheaper proxy feedback, smaller analyzer models, or more autonomous optimization of the data-generation recipe. Second, RL training uses a terminal answer-level binary reward from the LLM-as-judge verifier. Although ODE uses trace-level analysis during data construction to diagnose visual grounding, tool-use structure, and difficulty match, the policy optimization signal itself is still based on final answer correctness. Future work could explore evidence-level or step-level rewards that more directly credit intermediate evidence gathering, image reuse, and tool-use decisions during reinforcement learning. Third, this paper instantiates data evolution in multimodal deep search. The core idea of adapting a data generator from policy rollouts is more general, and may also benefit other agentic settings such as text-only deep search or GUI agents. Extending ODE to these domains would require domain-specific harnesses, rubrics, and retention rules, and we leave this broader generalization to future work.

E Broader Impacts

This work may improve multimodal deep search agents for tasks such as shopping assistance, travel planning, and troubleshooting from user photos, where grounded evidence from images and the web is useful. It also offers a research insight that multimodal deep search is best understood as a co-design problem spanning the interaction workspace and the data-construction loop, rather than as a pure model-scaling problem. In particular, our results suggest that reusable visual evidence and policy-aware data evolution are both important for training stronger agents, which may inform future work on multimodal tool use, agent training, and benchmark design. At the same time, stronger multimodal deep search agents could be misused for large-scale scraping, privacy-invasive image identification, or generating plausible but unverified summaries. To mitigate these risks, we emphasize rate-limited tool access, evidence-citation requirements, filtering of personally identifying images when applicable, and human oversight for high-stakes use cases.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Section 2 introduce both contributions claimed in the paper, the visual-native agent harness with image bank reuse and On-policy Data Evolution (ODE), and the experimental gains reported in the abstract are exactly the numbers verified in Section 2 and the Experiments section across the same eight benchmarks.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss limitations in Appendix D, including the pre-computation cost of ODE, the use of terminal answer-level rewards for RL, and the current instantiation of data evolution in multimodal deep search.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper does not contain formal theoretical results, theorems, or proofs; the contributions are an agent harness design, a closed-loop data construction framework, and empirical evaluation.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The harness design (nine tools, image bank protocol), the ODE forward and backward stages, the rubric weights, the data construction protocol, the SFT/RL training hyperparameters, and the evaluation protocol with the LLM-as-judge prompt are all described in Section 2 and Appendices A, B.1, B.2, and B.3, including a fully worked end-to-end ODE round example.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We will release the ODE codebase, the visual-native agent harness, the curated SFT and RL data, and trained model checkpoints under an open-source license upon publication. The Appendix already provides the full forward/backward pipeline configuration, prompts, and hyperparameters needed to reproduce the data construction and training.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Appendices B.1, B.2, and B.3 specify the SFT/RL data sizes, sequence length, batch size, learning rate, number of epochs, RL algorithm (GRPO with leave-one-out), rollout sampling, decoding parameters, evaluation budget, and the LLM-as-judge protocol used for all reported numbers.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Appendix B.7 reports mean and standard deviation over three independent inference runs of the fixed ODE-8B-RL and ODE-30B-A3B-RL checkpoints across all eight benchmarks. These statistics capture inference-time variability, not training-seed variability. We do not claim statistical significance for the single-run ablations.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix B.2 reports the GPU type, optimizer, batch size, learning rate, sequence length, and number of epochs for SFT, the GRPO setup, rollout sampling and clip ratio for RL, and the asynchronous SGLang serving configuration; we additionally describe the evaluation interaction budget (LLM call cap, per-turn and total token caps) in Appendix B.3.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conforms with the NeurIPS Code of Ethics. The work does not involve human subjects, uses only publicly available benchmarks and pre-trained models under their respective licenses, and we preserve anonymity in all materials submitted for review.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss positive and negative societal impacts in the Appendix E section, including practical assistance applications, research insights on workspace and data-loop co-design, and risks such as scraping, privacy-invasive image identification, and unverified summaries, together with mitigations.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: The released ODE-curated SFT and RL data are derived from publicly accessible web sources and are checked by an LLM judge for verifiability and shortcut leakage; we plan to release the data and model checkpoints under an open-source license, exclude personally identifying images, and accompany the release with a model card that describes the intended scope (multimodal deep search research) and known failure modes.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All used assets, including the Qwen3-VL-8B-Instruct and Qwen3-VL-30B-A3B-Instruct backbones [Bai et al., 2025], the eight evaluation benchmarks (MMBC, HLE-VL, BC-VL, VDR, MMSearch, MMSearch+, SimpleVQA, FVQA), GRPO [Shao et al., 2024], and SGLang [Zheng et al., 2024b], are cited in the main text and appendix and used in accordance with their original licenses and intended research use.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We introduce three new assets: the visual-native agent harness, the ODE-curated SFT demonstration set, and two ODE-curated RL task sets (ODE-8B and ODE-30B-A3B). Section 2 and Appendices A, B.1, and A.11 document the construction pipeline, prompts, rubrics, sizes, domain coverage, and difficulty distribution; documentation will accompany the public release.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing or research with human subjects; all data construction and evaluation are performed automatically through LLM-driven curation, judging, and policy rollout.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve human subjects research and therefore does not require IRB approval or equivalent review.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: LLMs are a core component of our methodology: Qwen3-VL serves as the trained policy, and a frozen LLM (GPT-5.2) acts as the seed proposer, web explorer, graph organizer, task curator, SFT teacher, trace analyzer, configuration optimizer, and final-answer judge. Their roles, prompts, and rubric weights are documented in Section 2 and Appendices A, B.1, and B.3.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.