NEOS: LEARNING TO INTERLEAVE SEARCH AND VISUAL REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent multimodal deep search agents have advanced tool use and visual content retrieval, yet they still struggle to convert intermediate visual evidence into actionable guidance for subsequent search, limiting their effectiveness in long-horizon settings. To address this gap, we present **Neos**, a multimodal deep search framework that learns both *how to acquire evidence* and *how to use it to guide subsequent decisions*. To make this coupling learnable, Neos constructs training tasks with intermediate visual dependencies while filtering out textual shortcuts, and introduces a lightweight, trainable Multimodal Evidence Interface (MEI) that transforms retrieved content into goal-conditioned observations. At its core, an *agent–environment co-evolution* procedure treats MEI as a co-adaptive component of this loop. Specifically, reference-guided credit assignment uses verified successful trajectories to localize critical policy errors while preserving credit for useful decisions in failed searches, whereas rollout feedback selects informative local evidence-processing interactions for MEI supervised fine-tuning (SFT). By alternating policy reinforcement learning (RL) with MEI SFT, Neos allows the policy's evolving evidence requests to shape MEI training and the adapted MEI to reshape subsequent observations, closing the loop between evidence acquisition and evidence-conditioned action. Across ten benchmarks, experiments show that Neos-9B improves over its backbone under the same tool workflow by 17.4 percentage points on average, achieving leading performance among specialized multimodal deep search agents.

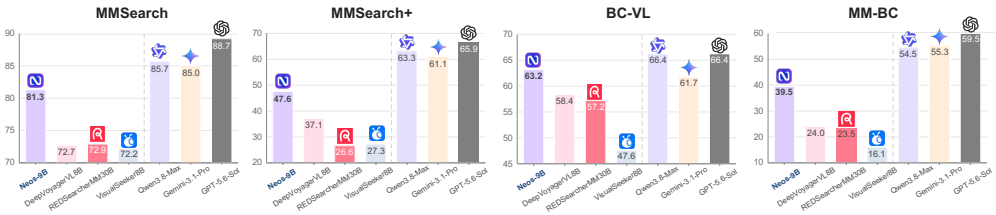


Figure 1: **Benchmark performance of Neos-9B.**

1 INTRODUCTION

Real-world information-seeking problems are often open-ended and grounded in both textual and visual evidence. Deep search agents have made substantial progress in text-centric settings by iteratively searching the web, browsing sources, and synthesizing grounded answers (Jin et al., 2025; Zheng et al., 2025; Wu et al., 2026; Li et al., 2025a; Team et al., 2025b;a; Li et al., 2026). Recent multimodal systems extend these agents with multimodal tool use, multi-turn visual interaction, scalable data synthesis, and visual context management (Hong et al., 2026; Chng et al., 2025; Zhang et al., 2026c;d; Yao et al., 2026; Huang et al., 2026a; Liu et al., 2026b; Du et al., 2026). Despite these advances, many systems fail to consistently leverage visual evidence during search. Yet such evidence is pivotal for guiding subsequent queries, interpreting retrievals, and determining the need for further inspection (Li et al., 2025b; Tao et al., 2026; Zeng et al., 2026). The central challenge is therefore to sustain an *evidence–decision loop*, in which search decisions determine which evidence is acquired, and newly acquired evidence guides subsequent decisions.

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

Existing multimodal deep search agents still exhibit three intertwined limitations: **I) Weak intermediate visual dependencies.** Training and evaluation tasks often do not require newly acquired visual evidence to influence later search. Synthesized tasks may confine vision to initial recognition or final attribute extraction (Wu et al., 2025; Zhang et al., 2026d), while captions, metadata, or indexed text can bypass intermediate visual reasoning (Zeng et al., 2026). Consequently, agents can succeed without internalizing the paradigm of using newly acquired visual evidence to guide subsequent decisions. **II) Ambiguous process credit.** Prior work provides fine-grained learning signals through advantage estimation, reward shaping, and process-aware optimization (Feng et al., 2026; Xia et al., 2026; Lu et al., 2026; Liu et al., 2026a; Yan et al., 2026a). However, these signals often lack an evidence-grounded reference for attributing trajectory-level outcomes to individual decisions. Final-answer supervision therefore remains too global to identify critical errors, distinguish them from downstream consequences, or preserve credit for useful decisions in failed searches. **III) Decoupled policy and evidence interface.** Retrieved content is commonly converted into observations by fixed, often proprietary auxiliary models (Zhang et al., 2026b). Beyond their inference cost, these interfaces cannot adapt to the policy's evolving evidence needs, preventing evidence presentation and search behavior from improving together.

To address these limitations, we present **Neos**, a multimodal deep search framework that learns both how to acquire evidence and how to use it to guide subsequent decisions. Neos closes this evidence–decision loop through three complementary components: I) Evidence-dependent data synthesis. We construct tasks with intermediate visual dependencies and screen for textual shortcuts, producing questions and verified successful trajectories in which newly acquired visual evidence is necessary for later search. II) Reference-guided credit assignment. Using verified successful trajectories and the evidence available at each turn, our credit-assignment method localizes critical errors in evidence seeking and use, attenuates penalties for their downstream consequences, and preserves positive credit for useful decisions in failed searches. This provides targeted reinforcement-learning signals for both evidence acquisition and evidence-conditioned decision making. III) Agent–environment co-evolution. We embed a lightweight, trainable Multimodal Evidence Interface (MEI) into the search tools to transform retrieved content into goal-conditioned observations, while leaving search and on-demand visual inspection under policy control. After supervised initialization, Neos alternates policy reinforcement learning (RL) with MEI supervised fine-tuning (SFT). Rollout outcomes and shared process feedback select local evidence-processing interactions for MEI training, including useful interactions from failed searches. The policy's evolving evidence needs thereby shape MEI training, while the adapted MEI reshapes observations for subsequent policy learning.

Our contributions are fourfold:

- We introduce an **evidence-dependent synthesis pipeline** that produces multimodal search tasks in which newly acquired visual evidence is necessary for subsequent search decisions.
- We propose **reference-guided credit assignment**, which localizes critical policy errors and their downstream consequences, while preserving positive credit for productive and corrective decisions in failed searches.
- We develop a lightweight **Multimodal Evidence Interface (MEI)** and an **agent–environment co-evolution** procedure, allowing joint improvement of evidence presentation and the search policy.
- Across ten benchmarks, Neos-9B improves over its backbone by 17.4 points on average and achieves leading performance among open-source multimodal deep search agents.

2 Related Work

From multimodal understanding to active information seeking. MLLMs have advanced visual understanding and reasoning but remain limited by parametric knowledge and inference-time evidence. AVIS and Vision Search Assistant extend visual question answering with tools and web search (Hu et al., 2023; Zhang et al., 2024). Recent "thinking with images" methods integrate visual interaction into reasoning: DeepEyes learns image manipulation via RL (Zheng et al., 2026); Adaptive-CoF performs question-guided region selection and zooming, and MGPO learns multi-turn visual grounding from answer-level rewards (Zhang et al., 2025a; Huang et al., 2026c); RSAgent uses visual feedback for segmentation (He et al., 2025); PyVision and Thyme use generated Python

tools and executable code for visual computation (Zhao et al., 2025a; Zhang et al., 2025b); Mini-o3 extends exploratory visual reasoning to longer interactions (Lai et al., 2026); and MAPO aligns tool outputs with visual observations in policy optimization (Yang et al., 2026). These methods make visual inspection and manipulation part of reasoning, but multimodal deep search must iteratively acquire and verify external evidence to guide subsequent search.

Multimodal deep search agents. Multimodal agents learn multi-turn evidence acquisition through RL, e.g., MMSearch-R1 and WebWatcher (Wu et al., 2025; Geng et al., 2026b). ProMMSearchAgent uses process-oriented rewards, and OpenSearch-VL preserves useful prefixes via fatal-aware masking and advantage clamping (Yan et al., 2026b; Chen et al., 2026). Scalable supervision includes graph-based task construction and trajectory synthesis (Chu et al., 2026; Yao et al., 2026); VSearcher combines iterative-injection data synthesis with SFT-then-RL (Zhang et al., 2026c), while ODE adapts data generation from rollout feedback (Huang et al., 2026a). Evidence handling also differs: Vision-DeepResearch supports multi-turn, multi-entity, and multi-scale search (Huang et al., 2026b); LMM-Searcher separates image references from on-demand acquisition (Du et al., 2026); POINTS-Seeker renders older observations as images while keeping recent text (Liu et al., 2026b); DeepVoyager-VL constructs intermediate visual dependencies with helper-mediated observations and reusable visual state (Zhang et al., 2026b); and SimpleSearch-VL places webpage summarization and thumbnail verification within the policy (Dai et al., 2026).

3 METHOD

3.1 PROBLEM SETTING AND AGENT WORKFLOW

We formulate multimodal deep search as an *evidence–decision loop*. Given a question q and initial images $\mathcal{I}_0$, the policy π_θ interacts with a tool environment $\mathcal{E}_\phi$ to produce an answer. At turn t, history h_t contains the inputs and preceding action–observation pairs,

$$a_t \sim \pi_\theta(\cdot \mid h_t), \qquad o_t \sim \mathcal{E}_\phi(\cdot \mid h_t, a_t), \tag{1}$$

with $h_{t+1} = h_t \oplus (a_t, o_t)$. The interaction ends when the policy produces a final answer.

Neos discovers candidate evidence through `TextSearch`, `ImageSearch`, and `ReverseImageSearch`, and accesses source pages via `WebVisit`. These tools return text and image URLs, while `FetchImage` and `CropImage` expose selected images and regions on demand (Du et al., 2026). Loaded images and crops remain accessible through reusable identifiers in an episode-local visual register (Zhang et al., 2026b). To present retrieved multimodal content as policy observations, Neos incorporates a trainable Multimodal Evidence Interface (MEI) within `WebVisit`, `ImageSearch`, and `ReverseImageSearch` to present search results as goal-conditioned observations. The parameters ϕ govern this interface within $\mathcal{E}_\phi$, while the underlying web and search services remain fixed (Section 3.2).

3.2 MULTIMODAL EVIDENCE INTERFACE

Online content becomes useful for decision-making only after the agent identifies evidence relevant to its current search goal. Existing search pipelines often delegate this processing to large or proprietary auxiliary models (Wu et al., 2025; Zhang et al., 2026b), incurring repeated inference latency and serving cost. When these models remain fixed, evidence processing cannot adapt to the evolving policy's needs. MEI addresses these limitations through a lightweight, trainable vision–language model, g_ϕ, that adapts evidence presentation to the policy's current search objective.

Embedded in `WebVisit`, `ImageSearch`, and `ReverseImageSearch`, MEI is implemented by a lightweight vision–language model. Conditioned on the policy's local goal and the returned material, it extracts relevant page evidence, filters and ranks image candidates, and assesses visual matches. Thus, evidence is selected and summarized for the current search objective rather than presented as an undifferentiated collection of retrieved material. It inspects candidate images internally, but returns structured text and image references as ordinary tool observations. The agent does not invoke MEI as a separate model and retains control over query formulation and on-demand visual inspection. This separates learning what evidence to present from learning how to act on it.

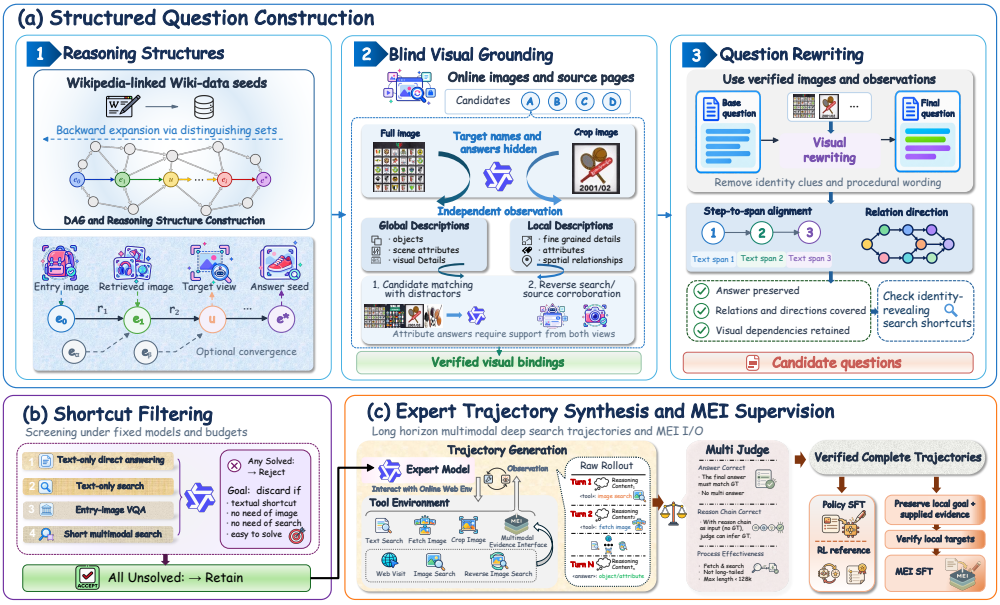


Figure 2: **Data synthesis pipeline.** Reasoning structures with visual checkpoints undergo blind visual grounding, question rewriting, and shortcut screening. Verified successful searches provide policy demonstrations, success references, and local MEI training interactions.

3.3 Evidence-Dependent Task Synthesis

Our data pipeline is designed to instantiate the *evidence–decision loop* at the task level: an agent must acquire intermediate visual evidence, and that evidence must constrain subsequent search and reasoning. The presence of images alone does not guarantee this property. Tasks may restrict vision to initial recognition or final attribute extraction, while textual clues can reveal identities intended to be inferred visually. We therefore construct tasks in which intermediate visual evidence supplies entity bindings for later reasoning steps, and combine structural checks, blind visual grounding, and shortcut screening before collecting verified searches (Figure 2).

Reasoning structures and identifiability. To make intermediate evidence decision-relevant, we construct reasoning structures in which visual checkpoints are required for later relational inference. Starting from Wikipedia-linked Wikidata entities, we recursively expand *distinguishing sets*: neighboring relational constraints that jointly identify an entity within a knowledge-graph (KG) snapshot $\mathcal{K}$. The resulting DAG connects prerequisites to identified entities while preserving the direction of each KG fact. Reasoning structures comprise paths or convergent subgraphs with a visual entry anchor and at least one intermediate visual checkpoint; converging branches have disjoint entity sets before their shared entity. These checkpoints connect visual entity identification to subsequent relational reasoning, making newly acquired visual evidence a prerequisite for later steps. To check that the specified evidence bindings identify an unambiguous target, we evaluate

$$\mathcal{U}_{\mathcal{K}}(C,B)=\left\{u \;\middle|\; \exists \mathbf{z},\ z_{\text{out}}=u,\ \bigwedge_{(v,r,w)\in C}(z_v,r,z_w)\in\mathcal{K},\ \bigwedge_{v\in B} z_v=b_v\right\}. \tag{2}$$

Here, C denotes relational constraints, B denotes bindings at the entry and checkpoints, $\mathbf{z}$ assigns entities to graph variables, and z_{out} denotes the final target entity. We retain structures satisfying $\mathcal{U}_{\mathcal{K}}(C,B)=\{u^\star\}$, ensuring that the seed $u^\star$ is uniquely identified within $\mathcal{K}$ conditional on the specified bindings. This check establishes structural identifiability; the following stages assess whether the required evidence is visually grounded and whether textual bypasses remain.

Blind visual grounding. To ensure that the required bindings are genuinely obtainable from visual evidence, an observer model describes full images and crops independently, without target identities or expected answers. These descriptions are matched against candidates containing distractors, with the target unmarked. Bindings require distinctive visual evidence or compatible observations

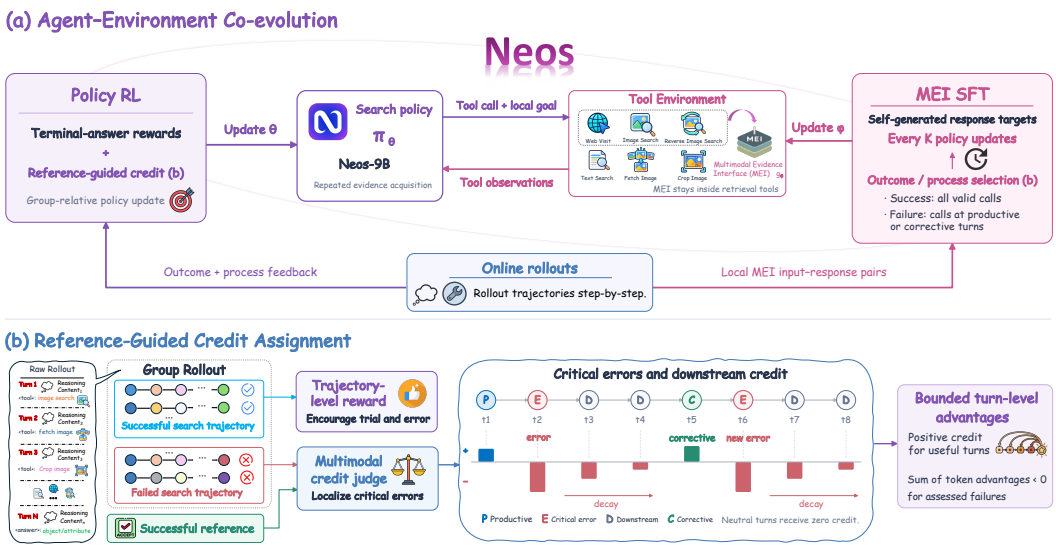


Figure 3: **Agent–environment co-evolution.** Policy RL alternates with MEI SFT on outcome- and process-filtered online interactions. Reference-guided credit supports policy optimization and identifies useful local interactions for MEI updates.

corroborated by reverse search or source pages; visual attribute answers require support from both full-image and crop observations.

Question rewriting and filtering. Even with evidence-dependent structures, a question can still be solvable through leaked names or text-only shortcuts. We verbalize each reasoning structure, then use a multimodal rewriter conditioned on verified images, crops, and observations to remove revealing names and procedural wording while preserving the answer and dependencies. An independent check aligns each step with a span in the final question to verify coverage and relation direction, while targeted text searches screen for wording that exposes hidden identities. We then apply four sequential probes—text-only answering, text-only search, entry-image VQA, and multimodal search—and reject questions solved by any probe, with limited budgets for the search probes. The text-only and entry-image probes screen for routes that bypass intermediate visual acquisition, while the budget-limited multimodal probe additionally filters task difficulty.

Trajectory curation. For retained questions, an expert model conducts long-horizon searches, producing trajectories that follow the desired evidence–decision loop. Successful trajectories serve as demonstrations and success references for credit assignment, while local tool interactions provide MEI training inputs. The resulting supervision connects task construction to both search-policy learning and evidence-interface training. Further details appear in Appendix A.

3.4 Reference-Guided Credit Assignment

An incorrect final answer does not invalidate all preceding decisions; a failed search may still acquire relevant evidence or correct earlier mistakes. Although process-oriented rewards and prefix-preserving updates provide finer-grained supervision (Yan et al., 2026b; Chen et al., 2026), uniformly propagating trajectory-level penalties still obscures turn-level contributions. Neos instead uses verified successful trajectories as references to identify the role of each turn.

Supervised initialization. The agent and MEI are initialized on their respective curated examples using a masked autoregressive objective normalized over target tokens. Policy targets comprise reasoning, tool calls, and final answers, while MEI targets comprise structured evidence responses. Conditioning content, including tool observations, and padding are excluded from the loss.

Reference-guided assessment. For each question, we sample G rollouts under $\pi_{\theta_{\text{old}}}$ and $\mathcal{E}_\phi$, retaining complete groups containing both successes and failures. Binary answer-correctness rewards r_i define group-relative advantages $A_i = (r_i - \bar{r})/(\text{std}(\mathbf{r}) + \varepsilon_{\text{std}})$, where $\bar{r} = G^{-1}\sum_j r_j$. For failed trajectories, a credit judge assesses recorded actions and observations against a verified successful

reference. The reference grounds assessment in a demonstrated successful investigation, rather than prescribing a required sequence of actions. The judge identifies critical policy errors, their downstream consequences, and productive or corrective decisions while respecting the evidence available at each turn. For successful trajectories, alternative valid search routes are permitted.

Turn-level credit. We focus negative credit on critical errors and attenuate penalties on their downstream consequences. Productive and corrective decisions receive positive credit, so useful evidence acquisition and recovery can be reinforced even within failed searches, rather than merely exempted from penalties. For an assessed failure, we set $b_i = |A_i|$ and assign

$$w_{i,t} = \begin{cases} -\max(\kappa_{\mathrm{e}}, s_{i,t}c_{i,t}), & \text{critical error,} \\ -\lambda_{\mathrm{d}} \max(\kappa_{\mathrm{d}}, s_{i,t}c_{i,t})\gamma^{d_{i,t}-1}, & \text{downstream consequence,} \\ \lambda_{\mathrm{c}} c_{i,t}, & \text{corrective,} \\ \lambda_{\mathrm{p}} c_{i,t}, & \text{productive,} \\ 0, & \text{neutral.} \end{cases} \tag{3}$$

Here, $s_{i,t}$ and $c_{i,t}$ denote severity and confidence. Unannotated turns following an active error inherit its severity and confidence, with $d_{i,t} \geq 1$ measuring elapsed turns; otherwise, they are neutral. Corrective turns stop propagation, and subsequent critical errors restart it. Each policy token k receives $\widetilde{A}_{i,k} = b_i w_{i,t(k)}$. Additional engineering techniques of our agentic RL are provided in Appendix B.

Policy optimization. We rescale positive and negative credit into bounded advantages $\widehat{A}_{i,k}$, keeping their sum negative for each assessed failure while preserving the uniform advantage A_i for successful trajectories. This reinforces useful local decisions while keeping failed trajectories negative in aggregate. Let $\mathcal{M}_i$ index policy-generated tokens and $\rho_{i,k}(\theta)$ denote the likelihood ratio between π_θ and $\pi_{\theta_{\mathrm{old}}}$ on the recorded prefix. Using the dual-clipped surrogate ℓ_{dc}, we optimize

$$\mathcal{L}_{\mathrm{RL}}(\theta) = -\frac{\sum_i \sum_{k \in \mathcal{M}_i} \ell_{\mathrm{dc}}\left(\rho_{i,k}(\theta), \widehat{A}_{i,k}\right)}{\sum_i |\mathcal{M}_i|}. \tag{4}$$

Outcome rewards remain unchanged, and credit estimates are held fixed during optimization. The SFT objective, credit rescaling, and dual-clipping definitions appear in Appendix D. The same process labels also select useful local MEI interactions, connecting decision-level credit with evidence-interface adaptation in Section 3.5.

3.5 Agent–Environment Co-evolution

Optimizing either component while holding the other fixed limits adaptation: a fixed policy exposes MEI to a static distribution of evidence requests, whereas a fixed MEI cannot adjust its observations to the policy's evolving needs. Neos instead co-evolves the two components, allowing evidence presentation and search behavior to adapt to each other and improve jointly (Figure 3).

Outcome- and process-filtered supervision. Since the evidence environment must adapt to the policy's evolving evidence requests, its supervision is drawn from online agent–MEI interactions rather than static demonstrations alone. After supervised initialization, we record each agent–MEI interaction, including inputs to MEI, its generated response, the corresponding policy turn and the outcome. Correct trajectories contribute all valid MEI calls; failed trajectories selected for agent optimization contribute only calls labeled *productive* or *corrective*. This filtering allows MEI to learn from evidence-processing interactions that remain useful even when the final search fails.

Alternating adaptation. We collect agent rollouts asynchronously under a fixed MEI version and update the agent after each. After every K agent updates, MEI is fine-tuned from its latest weights on newly retained calls collected since its previous update. The evolving agent determines which evidence-processing interactions become available for MEI training, and the updated MEI changes the observations used for subsequent agent decisions. This alternating procedure creates a reciprocal adaptation loop: policy updates reshape MEI's training signal, and MEI updates reshape the policy's decision context, while the underlying web and tool APIs remain unchanged.

Table 1: Comparison of multimodal search methods across 10 benchmarks. The best and second-best results are **bolded** and underlined. Green Δ rows report absolute score improvements over Qwen3.5-9B under our workflow for the SFT and RL models, respectively.

Model	MMSearch	MMSearch+	BC-VL	MM-BC	VisBrowse	BC-V^3	VDR	SimpleVQA	LiveVQA	FVQA	Overall
Direct Answer											
GPT-5.6-Sol	**68.3**	21.5	**47.4**	**16.8**	**36.7**	**22.7**	**19.2**	69.3	**73.0**	**63.7**	**43.9**
Gemini-3.1-Pro	62.3	**24.1**	41.6	11.8	24.2	18.7	16.0	64.7	64.3	57.7	38.5
Kimi-K3	61.3	17.0	40.1	11.8	30.8	15.3	11.4	70.0	72.7	61.0	39.1
Qwen3.8-Max	53.0	15.8	41.0	14.0	28.4	11.6	8.8	**72.7**	68.7	59.0	37.3
Qwen3.5-35B-A3B	26.7	4.5	25.3	8.5	17.8	6.3	2.6	55.3	56.3	34.3	23.8
Qwen3.5-9B	22.3	1.9	24.8	6.8	17.2	5.0	2.2	52.7	54.7	28.3	21.6
Agentic Workflow											
GPT-5.6-Sol	**88.7**	**65.9**	**66.4**	**59.5**	**70.4**	55.7	**44.6**	**87.0**	**94.0**	**88.7**	**72.1**
Gemini-3.1-Pro	85.0	61.1	61.7	55.3	63.3	51.3	40.2	83.0	88.7	81.3	67.1
Kimi-K3	82.3	65.0	64.6	56.5	68.0	53.3	33.8	85.0	91.0	82.0	68.1
Qwen3.8-Max	85.7	63.3	**66.4**	54.5	66.9	**56.3**	39.0	85.6	93.0	86.7	69.7
Qwen3.5-35B-A3B	71.0	28.1	51.4	29.0	42.0	30.7	19.4	71.7	83.7	65.0	49.3
Qwen3.5-9B	66.3	24.8	43.6	22.8	33.7	22.3	17.0	66.7	74.7	61.3	43.3
Multimodal deep search Agents											
MMSearch-R1-7B	53.8	–	–	–	–	4.0	–	57.4	48.4	58.4	–
WebWatcher-32B	55.3	–	27.0	–	–	8.7	–	59.0	58.7	–	–
DeepEyesV2-7B	63.7	–	–	–	–	–	–	59.4	–	60.6	–
SenseNova-MARS-32B	74.3	–	–	–	–	–	–	74.1	60.8	72.6	–
Vision-DeepResearch-30B-A3B	69.6	28.5	53.7	–	–	–	**37.8**	–	77.6	74.2	–
REDSearcher-MM-30B-A3B	72.9	26.6	57.2	23.5	–	–	–	–	79.3	–	–
MM-DeepResearch-32B	69.0	–	43.0	–	–	–	–	67.6	68.0	70.1	–
POINTS-Seeker-8B	70.8	25.2	44.4	–	–	–	–	68.8	77.7	71.2	–
LMM-Searcher-30B-A3B	72.3	34.8	–	30.1	48.3	–	–	–	–	–	–
OpenSearch-VL-30B-A3B	68.7	–	41.1	–	–	–	33.5	74.9	67.4	73.2	–
Visual-Seeker-8B	72.2	27.3	47.6	16.1	34.7	–	–	–	–	–	–
SearchEyes-9B	69.2	–	42.3	–	–	–	28.3	75.4	66.1	74.2	–
SimpleSearch-VL-8B	77.1	32.5	52.1	–	–	–	–	76.6	75.2	76.8	–
UniBrowse-35B-A3B	75.2	34.1	57.8	28.2	–	–	–	–	–	76.7	–
DeepVoyager-VL-8B	72.7	37.1	58.4	24.0	47.3	32.3	35.0	76.3	82.7	82.7	54.9
Neos-9B (Ours, SFT)	79.7	44.8	60.4	37.8	49.1	35.3	30.2	**82.3**	87.7	79.7	58.7
Δ_{SFT} vs. base workflow	**+13.4**	**+20.0**	**+16.8**	**+15.0**	**+15.4**	**+13.0**	**+13.2**	**+15.6**	**+13.0**	**+18.4**	**+15.4**
Neos-9B (Ours, RL)	**81.3**	**47.6**	**63.2**	**39.5**	**53.3**	**37.0**	31.4	81.3	**88.7**	**83.3**	**60.7**
Δ_{RL} vs. base workflow	**+15.0**	**+22.8**	**+19.6**	**+16.7**	**+19.6**	**+14.7**	**+14.4**	**+14.6**	**+14.0**	**+22.0**	**+17.4**

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Benchmarks. We evaluate Neos on ten benchmarks spanning factual visual question answering, multi-step information search, and reasoning over online visual evidence. The evaluation suite comprises MMSearch (Jiang et al., 2024), SimpleVQA (Cheng et al., 2025), LiveVQA (Fu et al., 2025), FVQA (Wu et al., 2025), BrowseComp-VL (Geng et al., 2026b), MM-BrowseComp (Li et al., 2025b), MMSearch-Plus (Tao et al., 2026), VDR-Bench (Zeng et al., 2026), BrowseComp-V^3 (Zhang et al., 2026a), and VisBrowse-Bench (Zhang et al., 2026e).

Baselines. We evaluate three categories of baselines, covering proprietary and open-source MLLMs as well as specialized multimodal search agents (Table 1). *Direct Answer* measures model capability using only the input question and images, without external tools. *Agentic Workflow* allows the same models to gather textual evidence and inspect images through our tool framework, assessing the benefit of iterative search without additional model training. *Multimodal deep search Agents* compares against existing systems specifically trained for multimodal search and browsing, providing a comparison with learned search policies.

Implementation Details. We perform cold-start SFT of Qwen3.5-9B (Team et al., 2026b) using MS-Swift (Zhao et al., 2025b) on 8 NVIDIA H100 GPUs for two epochs. Agentic RL is implemented in verl (Sheng et al., 2025) with Megatron (Shoeybi et al., 2019) as the training backend. The co-evolution stage uses 40 H100 GPUs, with 32 allocated to agentic RL and 8 to MEI serving and fine-tuning. Policy RL runs for one epoch with a batch size of 64 and 8 rollouts per question. The vision encoder and multimodal merger remain frozen throughout training, and evaluation allows up to 50 interaction turns per question. Detailed experimental settings are provided in Appendix C.

Table 2: **Training-data ablation.** The VQA mixture draws from FVQA, LiveVQA, InfoSeek, and VisualProbe, with the same expert model for trajectory generation and rejection sampling.

SFT data	BC-VL	MMSearch+	FVQA
Base Qwen3.5-9B	43.6	24.8	61.3
VQA mixture (0.7K)	53.4	32.9	72.7
REDSearcher-MM (1.1K)	56.9	38.9	77.3
Our synthetic data (1K)	58.4	43.2	78.3
Full mixture (2.8K)	**60.4**	**44.8**	**79.7**

Table 3: **Co-evolution ablation.** Initial and adapted refer to MEI before and after co-evolution. The RL settings use the same SFT policy initialization.

Policy	MEI	MMSearch+	BC-V^3	VDR
SFT	Initial	42.8	33.7	28.0
SFT	Adapted	44.8	35.3	30.2
RL	Frozen initial	45.8	35.7	28.2
RL	Frozen adapted	46.9	36.3	30.4
Alternating		**47.6**	**37.0**	**31.4**

4.2 Main Results

As shown in Table 1, Neos-9B achieves leading performance among specialized multimodal search agents, reaching an overall score of 60.7 and outperforming DeepVoyager-VL-8B by 5.8 points across ten benchmarks. Its strengths are particularly evident on MMSearch-Plus, MM-BrowseComp, and VisBrowse-Bench, where it achieves 47.6, 39.5, and 53.3, exceeding the strongest prior specialized-agent results by 10.5, 9.4, and 5.0 points, respectively. These gains are consistent with our emphasis on acquiring visual evidence that informs subsequent search, while strong results on LiveVQA and FVQA demonstrate the breadth of its capabilities.

The training progression further highlights the value of learning beyond tool access. Equipping Qwen3.5-9B with our workflow raises its overall score from 21.6 to 43.3, while cold-start SFT and subsequent co-evolution further improve it to 58.7 and 60.7, respectively. Neos-9B outperforms multiple 30B-scale multimodal deep search agents across diverse benchmarks, despite its smaller policy backbone. This cross-scale advantage suggests that effective agent design and training can deliver greater gains than increasing backbone size alone. Neos also substantially narrows the gap to frontier-model workflows, reducing the average gap to Gemini-3.1-Pro from 23.8 to 6.4 points.

4.3 Ablation Studies

Unless otherwise specified, ablations follow the settings of the corresponding stage in our main experiments. Except for the MEI-related ablations, all experiments use our final co-evolved MEI. Additional ablations and analyses are provided in Appendix E.

Training data. Table 2 compares SFT with different training sources. All trajectories, including those based on REDSearcher-MM and the VQA mixture, are generated and rejection-sampled using the same expert model. Our 1K synthetic examples yield 58.4, 43.2, and 78.3 on BrowseComp-VL, MMSearch-Plus, and FVQA, exceeding the REDSearcher-MM variant by 1.5, 4.3, and 1.0 points, respectively. The largest advantage on MMSearch-Plus supports the value of training on questions with intermediate visual dependencies. Combining all sources further improves performance to 60.4, 44.8, and 79.7, suggesting complementary benefits from general visual question answering and structured multimodal search supervision.

Agent–environment co-evolution. Table 3 separates the benefit of a stronger MEI from that of adapting it alongside the policy. Beyond evaluating the SFT policy with different MEI checkpoints, we compare three RL settings from the same SFT policy initialization with matched policy-update budgets: freezing the initial MEI, freezing the co-evolved MEI, and alternating policy RL with MEI SFT. Using the co-evolved MEI as a fixed interface improves over freezing the initial MEI by 1.1, 0.6, and 2.2 points on MMSearch-Plus, BrowseComp-V^3, and VDR-Bench, respectively. Alternating updates further improve performance by 0.7, 0.7, and 1.0 points over this stronger fixed-interface control, reaching 47.6, 37.0, and 31.4. These results show that a better MEI accounts for part of the improvement, while continued adaptation during policy optimization provides additional gains across all three benchmarks.

MEI backbone. We fix the final Neos-9B policy and vary the model serving as MEI in Table 4. Relative to the Qwen3.5-9B MEI baseline, our co-evolved MEI improves MM-BrowseComp, BrowseComp-VL, and VisBrowse-Bench by 4.5, 5.1, and 6.6 points, respectively. It outperforms Qwen-3.7-Flash, GPT-5.6-Luna, and GPT-5-nano on all three benchmarks, while remaining within

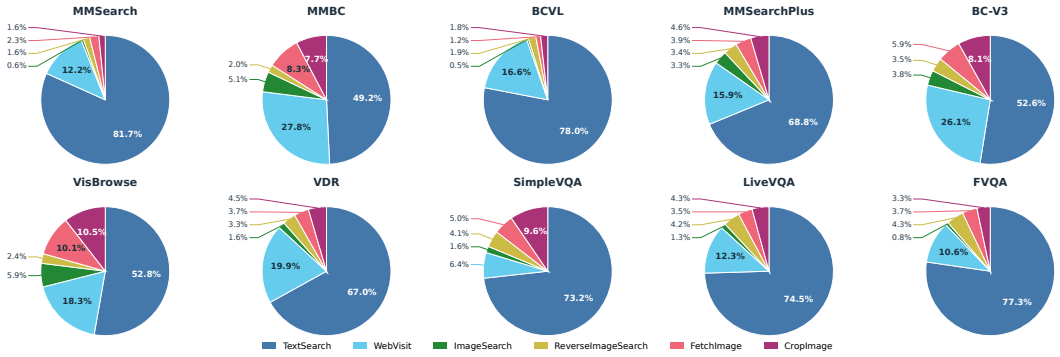


Figure 4: **Tool call distribution of ten benchmarks.**

Table 4: **MEI backbone ablation.**

MEI model	MM-BC	BC-VL	VisBrowse
Gemini-3.7-Flash	41.3	64.2	54.4
Qwen-3.7-Flash	39.3	62.4	51.5
GPT-5.6-Luna	38.7	61.2	52.7
GPT-5-nano	36.5	57.9	47.9
Qwen3.5-9B	35.0	58.1	46.7
Co-evolved MEI (9B)	39.5	63.2	53.3

Table 5: **Credit-assignment ablation.**

Training variant	BC-VL	MM-BC	FVQA
Cold-start SFT	60.4	37.8	79.7
Outcome-only GRPO	59.6	35.3	78.3
Productive credit only	60.4	36.3	77.7
+ First-error penalty	61.1	37.8	80.7
+ Downstream attribution	61.9	38.3	81.3
+ Corrective credit	62.7	39.0	82.0
Full credit assignment	**63.2**	**39.5**	**83.3**

1.8 points of Gemini-3.7-Flash on each. These results show that a task-adapted 9B evidence interface can support search performance competitive with strong off-the-shelf models.

Reference-guided credit assignment. Table 5 progressively introduces the credit components. Outcome-only GRPO falls below the cold-start SFT baseline on all three benchmarks, while productive-only credit provides mixed gains. Adding first-error penalties improves over productive-only credit by 0.7, 1.5, and 3.0 points on BrowseComp-VL, MM-BrowseComp, and FVQA, respectively. Downstream attribution and corrective credit then yield further improvements across all three benchmarks. The full method reaches 63.2, 39.5, and 83.3, exceeding outcome-only GRPO by 3.6, 4.2, and 5.0 points. This progression highlights the value of distinguishing critical errors, downstream consequences, and useful actions rather than rewarding productive steps alone.

Tool-use analysis. Figure 4 illustrates Neos's tool-use patterns across ten benchmarks. `TextSearch` accounts for the largest share of calls throughout, reaching 81.7% on MMSearch and 78.0% on BrowseComp-VL. In contrast, MM-BrowseComp, BrowseComp-V^3, and VisBrowse-Bench exhibit a more visually oriented tool mix: image search, reverse image search, image loading, and cropping collectively account for approximately 23%, 21%, and 29% of calls, respectively. On VisBrowse-Bench, `FetchImage` and `CropImage` alone contribute 20.6%, reflecting frequent image acquisition and fine-grained visual inspection. Meanwhile, `WebVisit` accounts for 27.8% and 26.1% of calls on MM-BrowseComp and BrowseComp-V^3, indicating substantial engagement with source pages beyond search results. Together, these patterns suggest that Neos combines text-based information gathering with selective visual exploration, with the balance varying across benchmarks.

5 Conclusions

We presented **Neos**, a multimodal deep search framework that sustains an *evidence–decision loop* by learning both evidence acquisition and evidence-conditioned decision making. Its evidence-dependent pipeline constructs tasks with intermediate visual dependencies while screening textual shortcuts. A lightweight Multimodal Evidence Interface (MEI) co-adapts with the policy through agent–environment co-evolution, while reference-guided credit assignment localizes critical errors and preserves credit for useful decisions in failed searches. Alternating policy RL with MEI SFT turns rollout outcomes into supervision for both policy and MEI. Across ten benchmarks, Neos-9B achieves leading performance among specialized multimodal deep search agents and substantially narrows the gap to frontier proprietary-model workflows.

AI USE STATEMENT

We used generative AI tools to polish the language and improve the readability of the manuscript. As part of our research pipeline, we used large language models and vision-language models to assist with question generation and rejection sampling during data construction. We also used judge models to assess answer correctness during evaluation and reinforcement learning, and a credit assignment model to provide turn-level credit signals during reinforcement learning. We take responsibility for verifying AI-assisted content and for the final text, claims, experimental results, and artifacts presented in this work.

ETHICS STATEMENT

All datasets used in this study were collected from publicly available sources in accordance with their respective licenses. Sensitive or unverifiable content was excluded during preprocessing to mitigate potential ethical risks. While Neos demonstrates substantial improvements in reasoning, perception, and search capabilities, these capabilities may pose risks, including misuse to generate misleading content, overreliance on automated reasoning, and privacy concerns when handling sensitive information. The proposed methods and released resources are intended strictly for research purposes. We strongly oppose malicious applications, including surveillance and the dissemination of misinformation, as well as their use for high-stakes decision-making without human oversight.

REPRODUCIBILITY STATEMENT

We describe the Neos framework and its data construction and filtering procedures in the main paper and appendix. Detailed experimental configurations, including hyperparameters for supervised fine-tuning and reinforcement learning, are provided in Appendix C. Evaluations use established public benchmarks, with references to the corresponding datasets provided in the paper. Our code and training data are currently being organized and documented for public release to facilitate reproduction and further research.

REFERENCES

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. *arXiv preprint arXiv:2605.05185*, 2026.

Yang Chen, Hexiang Hu, Yi Luan, Haitian Sun, Soravit Changpinyo, Alan Ritter, and Ming-Wei Chang. Can pre-trained vision and language models answer visual information-seeking questions? In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 14948–14968, 2023.

Xianfu Cheng, Wei Zhang, Shiwei Zhang, Jian Yang, Xiangyuan Guan, Xianjie Wu, Xiang Li, Ge Zhang, Jiaheng Liu, Yuying Mai, et al. Simplevqa: Multimodal factuality evaluation for multimodal large language models. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4637–4646. IEEE, 2025.

Yong Xien Chng, Tao Hu, Wenwen Tong, Xueheng Li, Jiandong Chen, Haojia Yu, Jiefan Lu, Hewei Guo, Hanming Deng, Chengjun Xie, et al. Sensenova-mars: Empowering multimodal agentic reasoning and search via reinforcement learning. *arXiv preprint arXiv:2512.24330*, 2025.

Zheng Chu, Xiao Wang, Jack Hong, Huiming Fan, Yuqi Huang, Yue Yang, Guohai Xu, Chenxiao Zhao, Cheng Xiang, Shengchao Hu, et al. Redsearcher: A scalable and cost-efficient framework for long-horizon search agents. *arXiv preprint arXiv:2602.14234*, 2026.

Ming Dai, Zhihong Lu, Jinjie Gu, Jiedong Zhuang, Yefeng Liu, Wankou Yang, Jian Wang, and Chunhua Shen. Simplesearch-vl: A simple recipe for multimodal agentic deep search. *arXiv preprint arXiv:2606.31504*, 2026.

Yifan Du, Zikang Liu, Jinbiao Peng, Jie Wu, Junyi Li, Jinyang Li, Wayne Xin Zhao, and Ji-Rong Wen. Towards long-horizon agentic multimodal search. *arXiv preprint arXiv:2604.12890*, 2026.

Lang Feng, Zhenghai Xue, Tingcong Liu, and Bo An. Group-in-group policy optimization for llm agent training. *Advances in Neural Information Processing Systems*, 38:46375–46408, 2026.

Mingyang Fu, Yuyang Peng, Benlin Liu, Yao Wan, and Dongping Chen. Livevqa: Live visual knowledge seeking. *arXiv e-prints*, pp. arXiv–2504, 2025.

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. *arXiv preprint arXiv:2607.07820*, 2026a.

Xinyu Geng, Peng Xia, Zhen Zhang, Xinyu Wang, Qiuchen Wang, Ruixue Ding, Chenxi Wang, Jialong Wu, Kuan Li, Yida Zhao, et al. Webwatcher: Breaking new frontiers of vision-language deep research agent. In *International Conference on Learning Representations*, volume 2026, pp. 61240–61271, 2026b.

Google. Gemini 3.1 Pro Preview. Gemini API documentation, 2026. URL `https://ai.google.dev/gemini-api/docs/models/gemini-3.1-pro-preview`. Accessed: 2026-09-21.

Nikita Gupta, Riju Chatterjee, Lukas Haas, Connie Tao, Andrew Wang, Chang Liu, Hidekazu Oiwa, Elena Gribovskaya, Jan Ackermann, John Blitzer, et al. Deepsearchqa: Bridging the comprehensiveness gap for deep research agents. *arXiv preprint arXiv:2601.20975*, 2026.

Xingqi He, Yujie Zhang, Shuyong Gao, Wenjie Li, Lingyi Hong, Mingxi Chen, Kaixun Jiang, Jiyuan Fu, and Wenqiang Zhang. Rsagent: Learning to reason and act for text-guided segmentation via multi-turn tool invocations. *arXiv preprint arXiv:2512.24023*, 2025.

Jack Hong, Chenxiao Zhao, ChengLin Zhu, Weiheng Lu, and Guohai Xu. Deepeyesv2: Toward agentic multimodal model. In *International Conference on Learning Representations*, volume 2026, pp. 114851–114872, 2026.

Ziniu Hu, Ahmet Iscen, Chen Sun, Kai-Wei Chang, Yizhou Sun, David Ross, Cordelia Schmid, and Alireza Fathi. Avis: Autonomous visual information seeking with large language model agent. *Advances in Neural Information Processing Systems*, 36:867–878, 2023.

Shijue Huang, Hangyu Guo, Guanting Dong, Chenxin Li, Junting Lu, Xinyu Geng, Zhaochen Su, Zhenyu Li, Shuang Chen, Hongru Wang, et al. Towards on-policy data evolution for visual-native multimodal deep search agents. *arXiv preprint arXiv:2605.10832*, 2026a.

Wenxuan Huang, Yu Zeng, Qiuchen Wang, Zhen Fang, Shaosheng Cao, Zheng Chu, Qingyu Yin, Shuang Chen, Zhenfei Yin, Lin Chen, et al. Vision-deepresearch: Incentivizing deepresearch capability in multimodal large language models. *arXiv preprint arXiv:2601.22060*, 2026b.

Xinyu Huang, Yuhao Dong, Weiwei Tian, Bo Li, Rui Feng, and Ziwei Liu. Mgpo: Thinking with images via multi-turn grounding-based reinforcement learning. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 383–399, 2026c.

Dongzhi Jiang, Renrui Zhang, Ziyu Guo, Yanmin Wu, Jiayi Lei, Pengshuo Qiu, Pan Lu, Zehui Chen, Chaoyou Fu, Guanglu Song, et al. Mmsearch: Benchmarking the potential of large models as multi-modal search engines. *arXiv preprint arXiv:2409.12959*, 2024.

Zhengbo Jiao, Yiming Cheng, Yilei Jiang, Kaituo Feng, Rui Huang, Tianyi Jiang, Juanxi Tian, Qunzhong Wang, Tailai Chen, Qianshan Wei, et al. Searcheyes: Towards frontier multimodal deep search intelligence via search world simulation. *arXiv preprint arXiv:2607.05943*, 2026.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.

Xin Lai, Junyi Li, Wei Li, Tao Liu, Tianjian Li, and Hengshuang Zhao. Mini-o3: Scaling up reasoning patterns and interaction turns for visual search. In *International Conference on Learning Representations*, volume 2026, pp. 76722–76746, 2026.

Kuan Li, Zhongwang Zhang, Huifeng Yin, Liwen Zhang, Litu Ou, Jialong Wu, Wenbiao Yin, Baixuan Li, Zhengwei Tao, Xinyu Wang, et al. Websailor: Navigating super-human reasoning for web agent. *arXiv preprint arXiv:2507.02592*, 2025a.

Shilong Li, Xingyuan Bu, Wenjie Wang, Jiaheng Liu, Jun Dong, Haoyang He, Hao Lu, Haozhe Zhang, Chenchen Jing, Zhen Li, et al. Mm-browsecomp: A comprehensive benchmark for multimodal browsing agents. *arXiv preprint arXiv:2508.13186*, 2025b.

Xiaoxi Li, Jiajie Jin, Guanting Dong, Hongjin Qian, Yongkang Wu, Ji-Rong Wen, Yutao Zhu, and Zhicheng Dou. Webthinker: Empowering large reasoning models with deep research capability. *Advances in Neural Information Processing Systems*, 38:120091–120131, 2026.

Boyang Liu, Senjie Jin, Peixin Wang, Zhangyue Yin, Yibo Wang, Yuhao Zhou, Xinbing Liang, Shizheng Zhu, Yuhui Wang, Jingqi Tong, et al. Cafe: Self-improving search agents need co-evolving feedback. *arXiv preprint arXiv:2608.24794*, 2026a.

Yikun Liu, Yuan Liu, Le Tian, Xiao Zhou, Jiangchao Yao, Yanfeng Wang, and Weidi Xie. Pointsseeker: Towards training a multimodal agentic search model from scratch. *arXiv preprint arXiv:2604.14029*, 2026b.

Yijun Lu, Rui Ye, Jiajun Wang, Yuwen Du, Tian Jin, Songhua Liu, and Siheng Chen. Abseeker: Training long-horizon search agents via answer-backtracked credit assignment. *arXiv preprint arXiv:2608.05102*, 2026.

OpenAI. GPT-5.6: Frontier intelligence that scales with your ambition, July 2026. URL `https://openai.com/index/gpt-5-6/`. Accessed: 2026-09-21.

Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, et al. Humanity's last exam. *arXiv preprint arXiv:2501.14249*, 2025.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, pp. 1279–1297, 2025.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*, 2019.

Xijia Tao, Teng Yihua, 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 *International Conference on Learning Representations*, volume 2026, pp. 70385–70422, 2026.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, Jianfeng Cai, Xinyuan Cai, Peizhou Cao, Yuxuan Cao, Ziwei Chai, Y Charles, et al. Kimi k3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026a.

MiroMind Team, Song Bai, Lidong Bing, Carson Chen, Guanzheng Chen, Yuntao Chen, Zhe Chen, Ziyi Chen, Jifeng Dai, Xuan Dong, et al. Mirothinker: Pushing the performance boundaries of open-source research agents via model, context, and interactive scaling. *arXiv preprint arXiv:2511.11793*, 2025a.

Qwen Team. Qwen3.8-max: A new bar for coding and cowork, 2026.

Qwen Team et al. Qwen3.5: Towards native multimodal agents, february 2026. *URL https://qwen.ai/blog*, 2026b.

Tongyi DeepResearch Team, Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, Guangyu Li, Guoxin Chen, Huifeng Yin, Jialong Wu, Jingren Zhou, et al. Tongyi deepresearch technical report. *arXiv preprint arXiv:2510.24701*, 2025b.

Jason Wei, Zhiqing Sun, Spencer Papay, Scott McKinney, Jeffrey Han, Isa Fulford, Hyung Won Chung, Alex Tachard Passos, William Fedus, and Amelia Glaese. Browsecomp: A simple yet challenging benchmark for browsing agents. *arXiv preprint arXiv:2504.12516*, 2025.

Xiyu Wei, Qingwei Zong, Zhuocheng Yu, and Sujian Li. Unibrowse: A data-to-agent framework for multimodal browsecomp. *arXiv preprint arXiv:2607.10557*, 2026.

Jialong Wu, Baixuan Li, Runnan Fang, Wenbiao Yin, Liwen Zhang, Zhenglin Wang, Zhengwei Tao, Ding-Chu Zhang, Zekun Xi, Robert Tang, et al. Webdancer: Towards autonomous information seeking agency. *Advances in Neural Information Processing Systems*, 38:120957–120985, 2026.

Jinming Wu, Zihao Deng, Wei Li, Yiding Liu, Bo You, Bo Li, Zejun Ma, and Ziwei Liu. Mmsearch-r1: Incentivizing lmms to search. *arXiv preprint arXiv:2506.20670*, 2(4), 2025.

Tianle Xia, Ming Xu, Lingxiang Hu, Yiding Sun, Wenwei Li, Linfang Shang, Liqun Liu, Peng Shu, Huan Yu, and Jie Jiang. Search-p1: Path-centric reward shaping for stable and efficient agentic rag training. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (ACL 2026)*, pp. 517–534, 2026.

Shilin Yan, Jintao Tong, Hongwei Xue, Xiaojun Tang, Yangyang Wang, Kunyu Shi, Guannan Zhang, Ruixuan Li, and Yixiong Zou. Act wisely: Cultivating meta-cognitive tool use in agentic multi-modal models. *arXiv preprint arXiv:2604.08545*, 2026a.

Wentao Yan, Shengqin Wang, Huichi Zhou, Yihang Chen, Kun Shao, Yuan Xie, and Zhizhong Zhang. Prommsearchagent: A generalizable multimodal search agent trained with process-oriented rewards. *arXiv preprint arXiv:2604.20486*, 2026b.

Wenhao Yang, Yu Xia, Jinlong Huang, Shiyin Lu, Qing-Guo Chen, Zhao Xu, Weihua Luo, Kaifu Zhang, Yuchen Zhou, Xiaobo Xia, et al. Walk the talk: Bridging the reasoning-action gap for thinking with images via multimodal agentic policy optimization. *arXiv preprint arXiv:2604.06777*, 2026.

Huanjin Yao, Qixiang Yin, Min Yang, Ziwang Zhao, Yibo Wang, Haotian Luo, Jingyi Zhang, and Jiaxing Huang. Mm-deepresearch: A simple and effective multimodal agentic search baseline. *arXiv preprint arXiv:2603.01050*, 2026.

Yu Zeng, Wenxuan Huang, Zhen Fang, Shuang Chen, Yufan Shen, Yishuo Cai, Xiaoman Wang, Zhenfei Yin, Lin Chen, Zehui Chen, et al. Vision-deepresearch benchmark: Rethinking visual and textual search for multimodal large language models. *arXiv preprint arXiv:2602.02185*, 2026.

Huanyao Zhang, Jiepeng Zhou, Bo Li, Bowen Zhou, Yanzhe Shan, Haishan Lu, Zhiyong Cao, Jiaoyang Chen, Yuqian Han, Zinan Sheng, et al. Browsecomp-v^3: A visual, vertical, and verifiable benchmark for multimodal browsing agents. *arXiv preprint arXiv:2602.12876*, 2026a.

Huanyao Zhang, Jiepeng Zhou, Runhao Zhao, Yanzhe Shan, Jiaoyang Chen, Bowen Zhou, Bo Li, Fang Wang, Jialong Wu, Zhengwei Tao, et al. Deepvoyager-vl: Incentivizing vision-in-the-loop search for long-horizon multimodal agents. *arXiv preprint arXiv:2608.01827*, 2026b.

Ruiyang Zhang, Qianguo Sun, Chao Song, Yiyan Qi, and Zhedong Zheng. Vsearcher: Long-horizon multimodal search agent via reinforcement learning. *arXiv preprint arXiv:2603.02795*, 2026c.

Xintong Zhang, Zhi Gao, Bofei Zhang, Pengxiang Li, Xiaowen Zhang, Yang Liu, Tao Yuan, Yuwei Wu, Yunde Jia, Song-Chun Zhu, et al. Chain-of-focus: Adaptive visual search and zooming for multimodal reasoning via rl. *arXiv e-prints*, pp. arXiv–2505, 2025a.

Yi-Fan Zhang, Xingyu Lu, Shukang Yin, Chaoyou Fu, Wei Chen, Xiao Hu, Bin Wen, Kaiyu Jiang, Changyi Liu, Tianke Zhang, et al. Thyme: Think beyond images. *arXiv preprint arXiv:2508.11630*, 2025b.

Zhengbo Zhang, Changtao Miao, Jinbo Su, Zhaowen Zhou, Chunxia Zhang, Xukai Wang, Ruiqi Liu, Kaiyuan Zheng, Jiansheng Cai, Bo Zhang, et al. Visual-seeker: Towards visual-native multimodal agentic search via active visual reasoning. *arXiv preprint arXiv:2606.15231*, 2026d.

Zhengbo Zhang, Jinbo Su, Zhaowen Zhou, Changtao Miao, Yuhan Hong, Qimeng Wu, Yumeng Liu, Feier Wu, Yihe Tian, Yuhao Liang, et al. Visbrowse-bench: Benchmarking visual-native search for multimodal browsing agents. *arXiv preprint arXiv:2603.16289*, 2026e.

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.

Shitian Zhao, Haoquan Zhang, Shaoheng Lin, Ming Li, Qilong Wu, Kaipeng Zhang, and Chen Wei. Pyvision: Agentic vision with dynamic tooling. *arXiv preprint arXiv:2507.07998*, 2025a.

Yuze Zhao, Jintao Huang, Jinghan Hu, Xingjun Wang, Yunlin Mao, Daoze Zhang, Zeyinzi Jiang, Zhikai Wu, Baole Ai, Ang Wang, et al. Swift: a scalable lightweight infrastructure for fine-tuning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 29733–29735, 2025b.

Yuxiang Zheng, Dayuan Fu, Xiangkun Hu, Xiaojie Cai, Lyumanshan Ye, Pengrui Lu, and Pengfei Liu. Deepresearcher: Scaling deep research via reinforcement learning in real-world environments. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 414–431, 2025.

Ziwei Zheng, Minghao Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, and Chao Shen. Deepeyes: Incentivizing" thinking with images" via reinforcement learning. In *International Conference on Learning Representations*, volume 2026, pp. 126775–126798, 2026.

Peilin Zhou, Bruce Leon, Xiang Ying, Can Zhang, Yifan Shao, Qichen Ye, Dading Chong, Zhiling Jin, Chenxuan Xie, Meng Cao, et al. Browsecomp-zh: Benchmarking web browsing ability of large language models in chinese. *arXiv preprint arXiv:2504.19314*, 2025.

Bin Zhu, Qianghuai Jia, Tian Lan, Junyang Ren, Feng Gu, Feihu Jiang, Longyue Wang, Zhao Xu, and Weihua Luo. Marco deepresearch: Unlocking efficient deep research agents via verification-centric design. *arXiv preprint arXiv:2603.28376*, 2026.

APPENDIX CONTENTS

A Details of Our Data Pipeline and Training Data **16**

A.1 Details of Our Data Pipeline 16

A.2 Training Data 17

B Engineering Techniques in Agentic RL **18**

C Experimental Details **19**

C.1 Benchmarks and Evaluation Splits 19

C.2 Baselines and Evaluation Protocol 20

C.3 Training and Implementation 20

D Supplementary Training Details **22**

E Additional Ablation Studies. **23**

F Prompts **27**

G Visualization of Interleaved Multimodal Search **35**

A Details of Our Data Pipeline and Training Data

A.1 Details of Our Data Pipeline

We construct questions designed to require newly acquired visual evidence during multimodal deep search. Their intended solutions interleave relation search with visual grounding and, when needed, visual answer extraction. Intermediate images resolve entities needed for subsequent search, while images of the resolved target provide visual attributes requested by the question. These roles can coexist within a question and recur along its solution path. Our pipeline first selects seed entities and constructs relational reasoning graphs, then derives and verifies programs with explicit visual dependencies, grounds them in online image evidence, and realizes natural-language questions before filtering candidates and curating successful search trajectories.

Selecting seeds for compositional visual search. Seed selection considers both the relational structure around an entity and the opportunities it offers for visual grounding. We draw from Wikipedia-linked Wikidata entities with substantive encyclopedic coverage, excluding disambiguation pages and non-entity entries. Within semantic domains, we prioritize diverse incoming and outgoing relations and the presence of image-associated neighbors, while downweighting highly connected hubs and neighborhoods dominated by generic attributes or taxonomic links. This favors seeds whose neighborhoods offer several potential visual checkpoints along a reasoning structure, including entities other than the seed itself. These signals guide construction; successful visual grounding and resistance to textual shortcuts are assessed downstream.

Expanding seeds into relational DAGs. We treat each seed as a candidate terminal entity and expand its relational prerequisites backward. For each expanded entity, we select a small *distinguishing set* of neighboring entities and relations whose conjunction uniquely identifies it in the KG snapshot. The neighbors become prerequisites for resolving that entity, and recursively expanding them produces a directed acyclic graph (DAG) of identification dependencies. Edges are oriented from prerequisites to the entity they identify, with the direction of each underlying Wikidata fact preserved. Expansion explores complementary relations across branches and recursively extends their dependencies, subject to depth and branching constraints. We favor concrete semantic relations, such as creation, membership, and location, and exclude encyclopedic maintenance and bibliographic-source links. The resulting DAG provides a structured search space for extracting reasoning structures, with entity identities and relation constraints established before question realization.

Constructing visual dependencies. From each DAG, we enumerate paths leading to the seed and subgraphs with two branches that converge on a shared entity, which either is the terminal entity or leads to it through further relations. Convergent branches have disjoint entity sets before the merge and constrain the shared entity through distinct relations. We select an upstream entity as the visual entry anchor and identify feasible visual checkpoints within the program. At an intermediate checkpoint, the visually grounded entity supplies the input to a subsequent relation; at an answer checkpoint, the retrieved image supplies the requested property of the resolved target. Programs contain one or multiple intermediate checkpoints, optionally followed by visual answer extraction. We apply structural filters before acquiring images and compile each program into a conjunctive query over the complete local KG snapshot $\mathcal{K}$:

$$\mathcal{U}_{\mathcal{K}}(C,B)=\left\{ u \;\middle|\; \exists \mathbf{z},\ z_{\text{out}}=u,\ \bigwedge_{(v,r,w)\in C}(z_v,r,z_w)\in\mathcal{K},\ \bigwedge_{v\in B} z_v=b_v \right\}. \tag{5}$$

Here, C specifies relation constraints expressed in their original KG direction, B specifies entry and planned visual-entity bindings, and z_{out} denotes the terminal entity. For a seed entity $u^\star$, we retain programs with $\mathcal{U}_{\mathcal{K}}(C,B)=\{u^\star\}$; multiple endpoints or unresolved query evaluations are rejected. This verifies endpoint identity conditional on the planned bindings. The target is either $u^\star$ or a visual property subsequently verified from images of that target.

Grounding dependencies in online evidence. For each verified program, a planner discovers images and source pages for the required checkpoints without introducing new graph relations. A

blind observer describes full images using sanitized scene cues, with target names, search titles, graph relations, and expected answers withheld. A separate observer examines actual crops without receiving the first observer's conclusions. For entity grounding, a binding step matches the visual evidence to candidate entities including distractors, without marking the intended target. We accept identities supported by distinctive pixels or by compatible visual observations corroborated through reverse search or source pages; incorrect and unresolved matches are rejected. Each intermediate binding is explicitly connected to its subsequent relation. For visual answer extraction, both full-image and crop observations must support the requested attribute. When a program spans converging branches, we additionally check distinct images, different source domains, and consistency with the verified graph. We verbalize the relations, compose a base question, and rewrite it to remove revealing names and mechanical structural cues while preserving its facts and answer. A semantic-coverage check verifies that each required relation remains expressed. The solver receives only the question and entry image; images needed at later checkpoints, including final attribute reading, must be acquired during search.

Filtering and curating search trajectories. We sequentially test candidates using text-only answering, text-only search with `TextSearch`, direct visual question answering with the entry image, and short multimodal search. Both search settings allow at most four tool-use turns followed by a mandatory final-answer turn. Any correct answer rejects the candidate; unresolved infrastructure or judging failures are excluded. These filters screen for shortcuts and insufficient search depth relative to the chosen models and budgets. For surviving questions, a stronger teacher performs long-horizon search, and an independent answer judge selects successful trajectories. We retain traces with complete observations, valid action order, and compliant sequence lengths for policy SFT and training-side success references. Local tool tasks additionally provide MEI training inputs, with targets separately regenerated and checked against the supplied material.

A.2 Training Data

We construct the policy training sets from our synthetic search tasks, REDSearcher-MM data, and a mixture of visual question answering datasets. Qwen3.8-Max serves as the expert model for trajectory generation and rejection sampling within the Neos workflow. Table 6 summarizes the resulting SFT and RL datasets.

Data sources and expert trajectories. Our data pipeline produces approximately 2K synthetic search instances with verified successful trajectories. To incorporate external search questions, we rank the REDSearcher-MM SFT data (Chu et al., 2026) by decreasing trajectory length, measured in interaction turns, and retain the top 60%. We additionally sample approximately 3.7K candidate questions from FVQA (Wu et al., 2025), LiveVQA (Fu et al., 2025), InfoSeek (Chen et al., 2023), and VisualProbe (Lai et al., 2026). For FVQA and LiveVQA, all sampled instances are selected from their corresponding official training splits, and after deduplication against the benchmarks used in our evaluation, the overlap rate with the evaluation benchmarks is 0%. For these external questions, we generate new trajectories using Qwen3.8-Max in Neos and retain successful trajectories through rejection sampling, rather than directly adopting demonstrations produced under another workflow.

SFT composition. We allocate 989 examples from our synthetic data to policy SFT and reserve the remaining 973 instances for RL. From the expert-generated external trajectories, we select 1,177 based on REDSearcher-MM questions and 732 based on the VQA mixture. The resulting SFT dataset contains 2,898 trajectories, combining demonstrations of interleaved multimodal search with broader visual question answering.

RL question selection. After cold-start SFT, we generate three rollouts for each question in the available data pool using the initialized policy, excluding the 973 synthetic instances reserved for RL. We retain questions answered correctly in zero or one of the three rollouts. The selected pool contains 1,599 questions: 599 from our synthetic data, 500 from REDSearcher-MM, and 500 from the VQA mixture. Combining these questions with the 973 reserved synthetic instances yields an RL dataset of 2,572 questions. Table 6 summarizes the composition of both training sets.

Table 6: **Composition of the policy training datasets.** SFT counts refer to expert-generated trajectories, whereas RL counts refer to questions. Difficulty filtering retains questions answered correctly in zero or one of three cold-start policy rollouts.

Data source	Selection	Count
SFT total		**2,898**
Our synthetic data	Expert trajectories	989
REDSearcher-MM	Expert trajectories	1,177
VQA mixture	Expert trajectories	732
RL total		**2,572**
Reserved synthetic data	Reserved before difficulty filtering	973
Our synthetic data	Difficulty-filtered	599
REDSearcher-MM	Difficulty-filtered	500
VQA mixture	Difficulty-filtered	500

Table 7: **Tools exposed to the Neos policy.** MEI is embedded in evidence-processing tools. Search and page-reading return text and image references; pixels enter the policy context through `FetchImage` or `CropImage`.

Tool	Role	Input	Output
`TextSearch`	Discover candidate webpages	Text query or query batch	Page titles, URLs, and snippets
`WebVisit`	Extract page evidence with MEI	Page URL(s) and local goal	Goal-relevant evidence and page-image URLs
`ImageSearch`	Rank image candidates with MEI	Text query(s) and visual goal	Ranked image/source URLs and visual analysis
`ReverseImageSearch`	Match visual evidence with MEI	Image IDs/URLs and matching goal	Match analysis and candidate/source URLs
`FetchImage`	Load selected visual evidence	Image URLs; optional captions	Image pixels and reusable image IDs
`CropImage`	Inspect fine-grained regions	Image IDs and normalized bounding boxes	Crop pixels and reusable image IDs

B Engineering Techniques in Agentic RL

Turn-level asynchronous execution. We decouple policy generation from tool execution and answer judging. Each trajectory maintains its own history; while its tools execute, other trajectories can submit ready turns to available SGLang replicas. Requests are scheduled at turn boundaries without permanently binding a trajectory to one replica. Thus, asynchronous execution overlaps model and environment work, while each rollout round still uses fixed policy and MEI versions.

Adaptive mixed-group sampling. For an update with B question groups and G trajectories per group, we admit up to $2BG$ trajectories over an active window of $2B$ questions. Completed trajectories are answer-judged immediately, and released slots replenish groups with the fewest completed samples. Groups with G homogeneous outcomes receive priority for additional samples, up to $1.5G$; groups still homogeneous at that limit are deferred. Once B mixed-outcome groups are available, we select eligible groups in queue order. If a selected group has more than G completed trajectories, we retain the lowest-session-index success and failure, then fill the remaining positions by session index, without enforcing a balanced success/failure ratio. This yields exactly BG trajectories with a nondegenerate group-relative learning signal.

Overlapped credit assessment and policy boundaries. Credit assessment of failures starts once a group becomes eligible and overlaps the remaining rollout work. Only assessments attached to

the ultimately selected trajectories affect optimization; successful trajectories retain ordinary GRPO advantages. Structured-output validation and bounded repair retries isolate malformed or timed-out assessments, with ordinary GRPO as the fallback. Before optimization, surplus old-policy rollouts are cancelled or discarded; unused questions are carried forward for fresh sampling after the weight update. Asynchrony therefore operates within a rollout round, not as unbounded cross-version training on stale trajectories.

Training and runtime overview. Figure 5 summarizes the training and runtime diagnostics. Rollout time remains stable around 80–90 minutes per update, confirming that asynchronous scheduling maintains consistent throughput. Policy optimization time drops sharply after the first update and stabilizes near 6 minutes, indicating negligible overhead from overlapped credit assessment. The selected-batch reward, surrogate objective, gradient norm, and PPO likelihood-ratio proxy all exhibit typical noisy-but-bounded dynamics without divergence or instability.

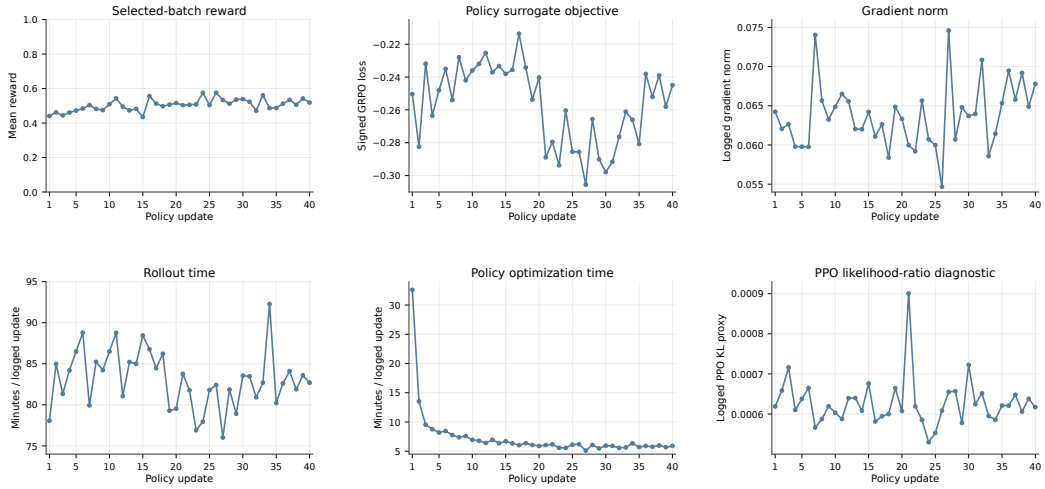


Figure 5: **Training and runtime diagnostics.** Top row: selected-batch reward, signed GRPO surrogate objective, and logged gradient norm. Bottom row: wall-clock rollout time (including environment waits), policy optimization time, and PPO likelihood-ratio diagnostic.

C Experimental Details

C.1 Benchmarks and Evaluation Splits

We evaluate Neos on ten benchmarks covering factual visual question answering, multi-step multimodal search, and reasoning over newly acquired visual evidence. Table 7 introduces the tools in Neos framework, and Table 8 specifies the evaluation set used for each benchmark.

Factual visual question answering. SimpleVQA (Cheng et al., 2025), LiveVQA (Fu et al., 2025), and FVQA (Wu et al., 2025) assess factual question answering grounded in visual inputs. LiveVQA emphasizes up-to-date visual knowledge, while FVQA includes both search-required and search-free questions. For these three benchmarks, we use the same 300-example subsets released with Vision-DeepResearch (Huang et al., 2026b).

Multimodal search and browsing. MMSearch (Jiang et al., 2024) evaluates information seeking using visual and textual web evidence, while MMSearch-Plus (Tao et al., 2026) emphasizes fine-grained visual clues and provenance-aware search. We evaluate all 300 MMSearch queries and all 311 MMSearch-Plus questions. BrowseComp-VL (Geng et al., 2026b) tests multi-step vision–language search; we include both difficulty levels, totaling 399 questions. For MM-BrowseComp (Li et al., 2025b), which requires integrating evidence encountered during multimodal browsing, we use the expanded 400-question release.

Table 8: **Evaluation sets used in our experiments.**

Benchmark	Examples	Evaluation set
MMSearch	300	Complete benchmark
MMSearch-Plus	311	Complete benchmark
BrowseComp-VL	399	Both difficulty levels
MM-BrowseComp	400	Expanded 400-question release
VisBrowse-Bench	169	Complete benchmark
BrowseComp-V^3	300	Complete benchmark
VDR-Bench	500	Official testmini subset
SimpleVQA	300	Vision-DeepResearch subset
LiveVQA	300	Vision-DeepResearch subset
FVQA	300	Vision-DeepResearch subset

Visual-native and verifiable search. BrowseComp-V^3 (Zhang et al., 2026a) evaluates cross-modal, multi-hop browsing with verifiable reasoning requirements, and VisBrowse-Bench (Zhang et al., 2026e) emphasizes inspecting newly discovered web images to advance the search. We use their complete evaluation sets. For VDR-Bench (Zeng et al., 2026), which targets complementary visual and textual search, we use the official 500-example testmini subset.

C.2 Baselines and Evaluation Protocol

Baseline categories. We organize the baselines in Table 1 into three groups, covering general-purpose MLLMs with and without tool access, as well as specialized multimodal search agents.

Direct Answer. We consider GPT-5.6-Sol (OpenAI, 2026), Gemini-3.1-Pro (Google, 2026), Kimi-K3 (Team et al., 2026a), Qwen3.8-Max (Team, 2026), Qwen3.5-35B-A3B, and Qwen3.5-9B. Each model receives the benchmark question and available input images and produces an answer without external tools. This setting assesses visual understanding and factual reasoning using only the supplied inputs and parametric knowledge.

Agentic Workflow. The same general-purpose models operate within our framework. The policy models are used without additional search-specific training and are allowed up to 50 interaction turns per question, which is the same as our Neos-9B 's evaluation protocol. Comparing this setting with Direct Answer measures the benefit of external evidence acquisition, while comparisons with Neos-9B assess the additional gains from specialized training.

Multimodal deep search Agents. We compare against MMSearch-R1-7B (Wu et al., 2025), WebWatcher-32B (Geng et al., 2026b), DeepEyesV2-7B (Hong et al., 2026), SenseNova-MARS-32B (Chng et al., 2025), Vision-DeepResearch-30B-A3B (Huang et al., 2026b), REDSearcher-MM-30B-A3B (Chu et al., 2026), MM-DeepResearch-32B (Yao et al., 2026), POINTS-Seeker-8B (Liu et al., 2026b), LMM-Searcher-30B-A3B (Du et al., 2026), OpenSearch-VL-30B-A3B (Chen et al., 2026), Visual-Seeker-8B (Zhang et al., 2026d), SearchEyes-9B (Jiao et al., 2026), SimpleSearch-VL-8B (Dai et al., 2026), UniBrowse-35B-A3B (Wei et al., 2026), and DeepVoyager-VL-8B (Zhang et al., 2026b). These baselines span different model scales and approaches to multimodal search training. For results taken from prior publications, we retain their reported benchmark coverage and evaluation settings. Unreported benchmark scores are marked with "–".

Evaluation metrics. We assess answer correctness using Qwen3.7-Flash with the official evaluation prompt released by Vision-DeepResearch (Huang et al., 2026b). The prompt supplies the question, reference answer, and model-produced answer or report. The judge determines whether the response contains an answer consistent with the reference, without requiring a particular answer format or location within the response. We apply this prompt and judge model across all ten benchmarks for predictions generated in our experiments.

C.3 Training and Implementation

Search infrastructure. Neos integrates Google for text-based web search, Google Images for image retrieval, and Google Lens for reverse image search, with Jina Reader supporting webpage

Table 9: **Hyperparameters for policy supervised fine-tuning.**

Hyperparameter	Value
Training epochs	2
Maximum sequence length	131,072 tokens
Global / micro batch size	8 / 1
Optimizer	AdamW
Adam (β_1, β_2)	$(0.9, 0.95)$
Adam ϵ	10^{-8}
Peak learning rate	10^{-5}
Minimum learning rate	10^{-6}
Learning-rate schedule	Cosine
Warmup fraction	5%
Weight decay	0.1
Gradient clipping	1.0
Precision	BF16
Packing / padding-free	Disabled / enabled
Random seed	42

Table 10: **Hyperparameters for agentic RL and online MEI supervised fine-tuning.**

Policy optimization		**Online MEI SFT**	
Hyperparameter	**Value**	**Hyperparameter**	**Value**
Optimizer	AdamW	Epochs per update	1
Learning rate	10^{-6}	Peak learning rate	10^{-6}
Adam (β_1, β_2)	$(0.9, 0.999)$	Minimum learning rate	10^{-7}
Learning-rate schedule	Constant	Learning-rate schedule	Cosine
Warmup	None	Warmup fraction	5%
Weight decay	0.01	Weight decay	0.1
Gradient clipping	1.0	Gradient clipping	1.0
PPO epochs per batch	1	Global / micro batch size	16 / 1
PPO clip, ε	0.2	Maximum sequence length	32,768 tokens
Dual clip, c_{dc}	3.0	Adam (β_1, β_2)	$(0.9, 0.95)$
KL / entropy coefficients	0 / 0	MEI update interval, K	5 through step 10; 10 thereafter
Rollout sampling		**Credit assignment**	
Hyperparameter	**Value**	**Hyperparameter**	**Value**
Groups per update, B	64	Critical-error floor, κ_{e}	0.5
Trajectories per group, G	8	Downstream floor, κ_{d}	0.25
Maximum samples per group	12	Downstream strength, λ_{d}	0.5
Active question window	128	Downstream decay, γ	0.65
Trajectory admission limit	1,024	Corrective weight, λ_{c}	0.3
Maximum context / turns	131,072 tokens / 50	Productive weight, λ_{p}	0.1
Temperature / top-p / top-k	1.0 / 1.0 / disabled	Negative amplification cap, M	4
Correct / incorrect reward	1.0 / 0.0	Negative margin, η	0.02

access. `FetchImage` and `CropImage` materialize selected images and image regions as actual pixels, retain them in an episode-local visual register, and encode them as multimodal inputs to the policy. Alibaba Cloud OSS primarily provides externally accessible URLs for query images required by reverse image search. To reduce API costs and improve efficiency, we employ a hierarchical cache spanning in-memory storage, node-local disk, and shared persistent storage. Reusing search

results, webpage content, and image assets across repeated requests avoids redundant external calls and data transfers, accelerating both rollout collection and evaluation.

Models and supervised initialization. The policy and MEI are separate instances of Qwen3.5-9B. Their language backbones are trainable, while the vision encoders and multimodal mergers remain frozen throughout training. Cold-start policy SFT uses MS-Swift (Zhao et al., 2025b) on 8 NVIDIA H100 GPUs, with 2.9K examples, two epochs, and a learning rate of 10^{-5}. The detailed hyperparameters of our SFT training is provided in Table 9 Policy supervision covers assistant reasoning, serialized tool calls, and final answers; tool observations serve as conditioning context and are excluded from the loss. MEI is supervised on local goal-conditioned evidence-processing responses. Training-data composition is described in Section A.

Agentic RL and MEI adaptation. Policy RL uses verl (Sheng et al., 2025) with the Megatron training backend (Shoeybi et al., 2019). The co-evolution stage runs on 40 H100 GPUs: 32 support policy RL, and 8 support MEI serving and supervised fine-tuning. Policy training uses 2.6K questions for one epoch, with a batch size of 64 questions and 8 rollouts per question, with Qwen3.7-Plus as the credit judge model. The detailed hyperparameters are shown in Table 10. Online MEI updates use outcome- and process-filtered local interactions, retaining valid calls from correct trajectories and productive or corrective turns from selected failures. The MEI version remains fixed within each trajectory, and updated checkpoints are activated before subsequent rollout batches. MEI adaptation therefore uses SFT with a learning rate of 10^{-6}.

D Supplementary Training Details

Supervised objective. For either the policy or MEI, let f_ψ denote the model and $\mathcal{S}$ a minibatch of serialized examples. We use

$$\mathcal{L}_{\mathrm{SFT}}(\psi;\mathcal{S}) = -\frac{\sum_{\mathbf{x}\in\mathcal{S}}\sum_k m_k(\mathbf{x})\log f_\psi(x_k \mid \mathbf{x}_{<k})}{\sum_{\mathbf{x}\in\mathcal{S}}\sum_k m_k(\mathbf{x})}, \tag{6}$$

where m_k selects target tokens and excludes conditioning content and padding. Policy targets include reasoning, tool calls, and final answers; tool observations provide context only. MEI targets are structured evidence responses conditioned on local goals and retrieved material. The same objective is used for MEI updates during co-evolution.

Credit parameters and rescaling. In Equation 3, severity and confidence satisfy $s_{i,t}, c_{i,t} \in [0,1]$. We use $\gamma \in (0,1)$, $\kappa_{\mathrm{e}}, \kappa_{\mathrm{d}} \in (0,1]$, $\lambda_{\mathrm{d}} \in [0,1]$, and $0 < \lambda_{\mathrm{p}} \le \lambda_{\mathrm{c}}$. Let $L_i = |\mathcal{M}_i|$ and define positive and negative credit masses

$$P_i = \sum_{k\in\mathcal{M}_i} [\widetilde{A}_{i,k}]_+, \qquad N_i = \sum_{k\in\mathcal{M}_i} [-\widetilde{A}_{i,k}]_+, \tag{7}$$

where $[u]_+ = \max(u,0)$. For an assessed failure with $N_i > 0$, a target margin $\mu_i = \eta b_i L_i$ ($\eta > 0$), and an amplification cap $M \ge 1$, we set

$$\begin{aligned} \alpha_i &= \min\left\{M, \max\left\{1, \frac{P_i+\mu_i}{N_i}\right\}\right\}, \quad \beta_i = \min\left\{1, \frac{[\alpha_i N_i - \mu_i]_+}{P_i}\right\}, \\ \widehat{A}_{i,k} &= \beta_i[\widetilde{A}_{i,k}]_+ - \alpha_i[-\widetilde{A}_{i,k}]_+, \end{aligned} \tag{8}$$

with $\beta_i = 1$ when $P_i = 0$. This keeps the sum of token advantages negative while bounding each advantage in $[-Mb_i, \lambda_{\mathrm{c}} b_i]$; the target margin is enforced only when feasible. Successful trajectories and failures without valid assessments identifying a critical error retain $\widehat{A}_{i,k} = A_i$; the same fallback applies when $N_i = 0$.

Dual-clipped surrogate. For a token likelihood ratio ρ and advantage A, define

$$\ell_\varepsilon(\rho, A) = \min\{\rho A, \mathrm{clip}(\rho, 1-\varepsilon, 1+\varepsilon)A\}. \tag{9}$$

With $c_{\mathrm{dc}} > 1$, the surrogate in Equation 4 is

$$\ell_{\mathrm{dc}}(\rho, A) = \begin{cases} \max\{\ell_\varepsilon(\rho, A), c_{\mathrm{dc}} A\}, & A < 0, \\ \ell_\varepsilon(\rho, A), & A \ge 0. \end{cases} \tag{10}$$

All judge assessments and adjusted advantages are detached from policy optimization.

E ADDITIONAL ABLATION STUDIES.

Learning the evidence–decision loop. Table 11 suggests that the benefit of two-stage training lies in the complementary roles of supervised initialization and reinforcement learning. Neither SFT-only nor RL-only training on the deduplicated union of both training sets matches the final Neos-9B model. We interpret this pattern as supporting a curriculum for learning the evidence–decision loop: cold-start SFT provides a behavioral prior for acquiring evidence, interpreting visual observations, and selecting the next action, while RL refines these transitions through interaction and outcome feedback. In particular, RL can improve how the agent decides whether the available evidence is sufficient, requires further inspection, or calls for a different search direction. Expanding SFT alone does not provide this on-policy refinement, whereas RL directly from the base model must acquire useful interaction patterns while also learning which decisions lead to successful outcomes. Cold-start SFT therefore offers a structured starting point that allows subsequent RL to focus on improving an already functional loop, rather than discovering the entire behavior through trial and error. This provides a plausible mechanism for reducing the additional rollout exploration, and consequently the training time and compute, that RL-only learning may require.

Table 11: **Training-stage ablation.** $\mathcal{S}$ is the question pool underlying the 2,898 SFT trajectories, $\mathcal{R}$ contains 2,572 RL questions, and $\mathcal{U} = \mathcal{S} \cup \mathcal{R}$ denotes their question-level deduplicated union. RL-only variants start directly from the base Qwen3.5-9B policy. SFT-only variants use the final co-evolved MEI; RL-based variants use the MEI obtained in their respective RL runs. The best and second-best results are **bolded** and underlined.

Training strategy	SFT pool	RL pool	BC-VL	MMSearch+	FVQA
Qwen3.5-9B	–	–	43.6	24.8	61.3
SFT only	$\mathcal{S}$	–	60.4	44.8	79.7
SFT only	$\mathcal{U}$	–	61.3	45.8	<u>81.3</u>
RL only	–	$\mathcal{S}$	59.6	44.1	80.3
RL only	–	$\mathcal{U}$	<u>62.4</u>	<u>46.3</u>	80.3
Neos-9B (SFT → RL)	$\mathcal{S}$	$\mathcal{R}$	**63.2**	**47.6**	**83.3**

Search horizons across benchmarks. Figure 6 presents the distribution of Neos-9B's interaction lengths, separated by answer correctness. Among trajectories terminating before the 50-turn limit, MMSearch and the VQA benchmarks concentrate in the first ten turns, while others exhibit broader distributions across intermediate search lengths. This contrast is consistent with relatively straightforward visual questions often being resolved through short searches, while demanding multimodal investigations involve more sustained exploration. Nevertheless, all benchmarks show a pronounced accumulation at the turn limit, including the VQA tasks. These budget-reaching trajectories are predominantly correct on LiveVQA and FVQA but predominantly incorrect on MM-BrowseComp, BrowseComp-V^3, and VDR-Bench. The coexistence of short and budget-reaching trajectories within each benchmark reveals heterogeneous search behavior, while their contrasting outcomes show that extended interaction is neither a guarantee of success nor necessarily a sign of failure.

Tool-use dynamics during RL. Figure 7 tracks the composition and frequency of tool calls across RL update windows. From the first to the final window, the share of `TextSearch` decreases from 59.7% to 44.5%, while `WebVisit` increases from 21.2% to 32.1%. The four visual tools—`ImageSearch`, `ReverseImageSearch`, `FetchImage`, and `CropImage`—collectively increase their share from approximately 19.1% to 23.4%. Meanwhile, the average number of calls per trajectory falls from 29.7 to 20.2, a reduction of 32.0%. Thus, the changing tool mix accompanies fewer overall calls rather than an expansion of interaction volume. These trends are consistent with a shift from text-search-heavy exploration toward more selective information gathering, with greater relative emphasis on examining source pages and visual evidence. The resulting trajectories are more compact in tool-call count while retaining a diverse set of evidence-processing operations.

Tool ablation. Table 12 evaluates individual tool contributions using Qwen3.5-35B-A3B without additional search-specific policy training. Removing any of the five tools reduces performance

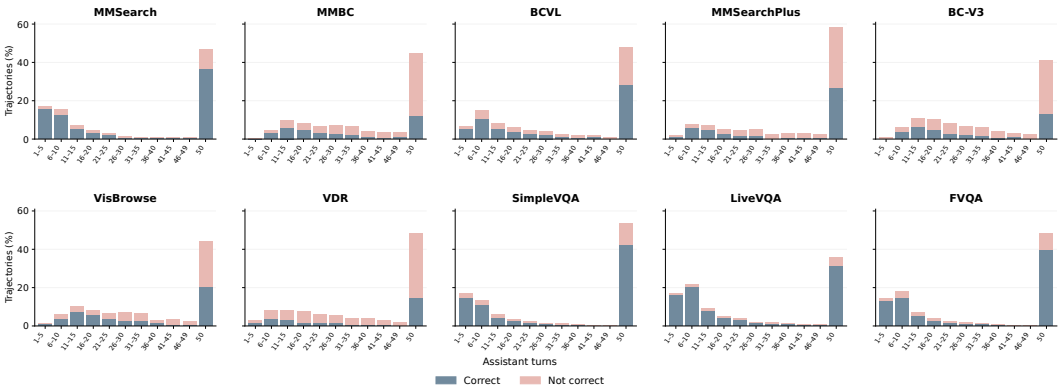


Figure 6: **Neos-9B's turn distribution across ten benchmarks.**

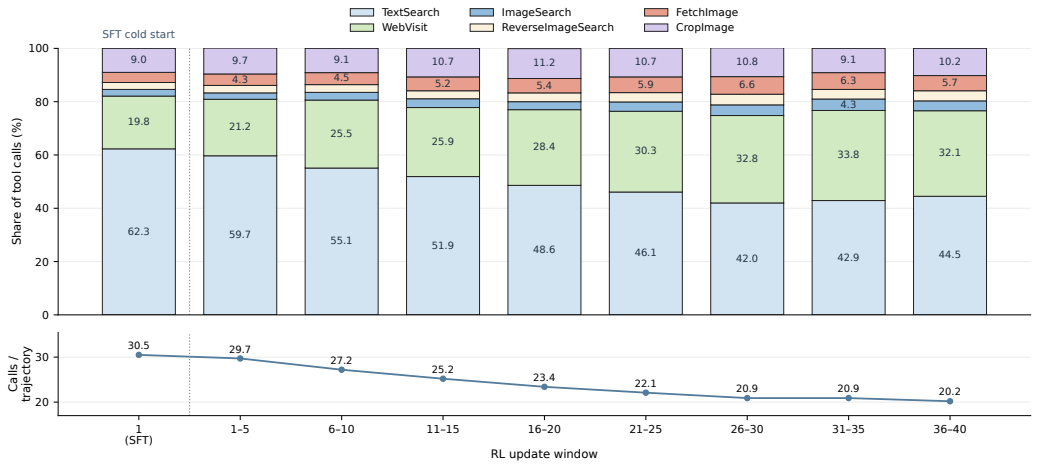


Figure 7: **Tool usage distribution during Neos-9B's co-evolution training.**

on all three benchmarks, but their effects differ substantially across tasks. On SimpleVQA, removing `ReverseImageSearch` produces the largest decline, from 71.7 to 58.0, while removing `ImageSearch` lowers accuracy by 4.7 points, highlighting the importance of image-based search for this benchmark. In contrast, removing `FetchImage` causes the largest degradation on BrowseComp-V^3 and MM-BrowseComp, with a 3.7-point drop on each, underscoring the value of inspecting newly acquired images. Disabling `CropImage` or `WebVisit` also consistently reduces performance. These benchmark-dependent effects support combining evidence discovery, page reading, and visual inspection within the Neos workflow.

Workflow comparison. Table 13 compares Neos with our reproduction of the REDSearcher-MM workflow (Chu et al., 2026), holding the policy model fixed within each comparison. Neos improves performance across all nine model–benchmark pairs, with average gains of 12.9, 9.0, and 7.8 points across the three benchmarks for GPT-5.6, Qwen3.5-35B-A3B, and Qwen3.5-9B. The largest improvements occur on MM-BrowseComp, reaching 24.0, 14.5, and 13.0 points. Notably, Qwen3.5-9B with Neos surpasses Qwen3.5-35B-A3B with REDSearcher-MM on both MM-BrowseComp and MMSearch-Plus. These results indicate that the benefits of Neos extend across policy backbones and that workflow design can partially compensate for differences in model scale.

Generalization to text-only tasks. We further evaluate Neos-9B on BrowseComp (Wei et al., 2025), BrowseComp-ZH (Zhou et al., 2025), HLE-Text (Phan et al., 2025), and DeepSearchQA (Gupta et al., 2026) to examine its applicability beyond multimodal tasks. As shown in Table 14, Neos-9B improves over the base Qwen3.5-9B under the same workflow by 12.2, 9.7, 11.9, and 10.6 points on these benchmarks, respectively, yielding an average gain of 11.1 points.

Table 12: **Tool ablation using Qwen3.5-35B-A3B without additional search-specific policy training.** Each ablation removes the indicated tool from Neos.

Tool configuration	SimpleVQA	BC-V^3	MM-BC
Full Neos workflow	**71.7**	**30.7**	**29.0**
w/o `FetchImage`	68.7	27.0	25.3
w/o `CropImage`	69.7	29.0	28.3
w/o `WebVisit`	70.3	29.0	27.0
w/o `ImageSearch`	67.0	28.7	27.7
w/o `ReverseImageSearch`	58.0	27.3	28.3

Table 13: **Workflow comparison with our reproduction of REDSearcher-MM.**

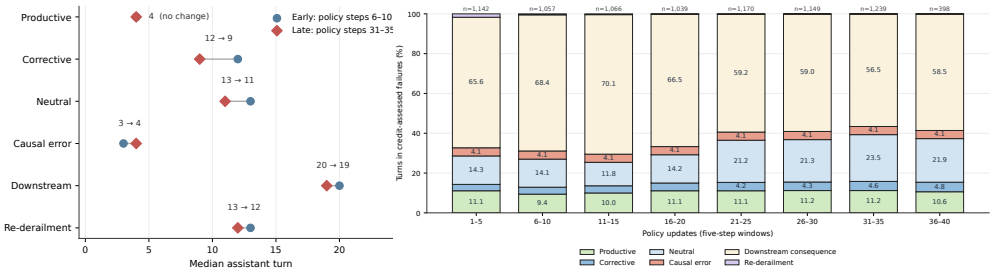
Policy model	Workflow	BC-VL	MM-BC	MMSearch+
GPT-5.6	REDSearcher-MM	64.2	35.5	53.5
	Neos	**66.4**	**59.5**	**65.9**
Qwen3.5-35B-A3B	REDSearcher-MM	45.4	14.5	21.6
	Neos	**51.4**	**29.0**	**28.1**
Qwen3.5-9B	REDSearcher-MM	38.6	9.8	19.3
	Neos	**43.6**	**22.8**	**24.8**

Compared with specialized search agents, it exceeds Marco-DR-8B (Zhu et al., 2026) by 5.0 points on BrowseComp and 25.1 points on DeepSearchQA. On HLE-Text, Neos-9B reaches 33.5, outperforming DeepSearch-World-9B (Geng et al., 2026a) and REDSearcher-MM-30B (Chu et al., 2026) by 7.8 and 8.2 points, respectively. It also remains competitive on BrowseComp-ZH, scoring 45.3 compared with Marco-DR-8B's 47.1 and exceeding the other baselines. Together, these results show that Neos's improvements extend beyond multimodal inputs to text-based search and reasoning, with gains over the base workflow highlighting the contribution of training beyond visual ability alone.

Table 14: **Generalization to text-only benchmarks.** The best and second-best scores in each column are **bolded** and underlined, respectively.

Model	BrowseComp	BrowseComp-ZH	HLE-Text	DeepSearchQA
Qwen3.5-9B-base(our workflow)	24.2	35.6	21.6	<u>43.7</u>
Marco-DR-8B	<u>31.4</u>	**47.1**	–	29.2
REDSearcher-MM-30B	31.2	44.5	25.3	–
DeepSearch-World-9B	31.2	36.4	<u>25.7</u>	–
Neos-9B	**36.4**	<u>45.3</u>	**33.5**	**54.3**

Credit Dynamics During RL Training. Figure 8 shows that RL training changes both where credit types occur and how frequently they appear. As shown in Figure 8a, *corrective* turns move earlier in the trajectory, from a median position of turn 12 to turn 9, while *neutral* turns shift from turn 13 to turn 11 and *re-derailment* from turn 13 to turn 12. *Productive* turns remain anchored at turn 4, *causal error* shifts slightly later from turn 3 to turn 4, and *downstream consequence* remains concentrated near the end of the trajectory, with its median position changing only from turn 20 to turn 19. Figure 8b further shows that *downstream consequence* dominates throughout training, accounting for 66–70% of credited failure turns in the first half before declining to 56–59% later, while the share of *neutral* turns rises from around 14% to 22–24%. In contrast, *productive* and *causal error* shares remain relatively stable at approximately 10–11% and 4%, respectively. The share of *corrective* turns increases from 3.2% to 4.8%, indicating more frequent recovery behavior within failed searches, whereas *re-derailment* appears only in the earliest window (1.7%) and is rarely observed thereafter. Taken together, these results suggest that, as training progresses, corrective actions occur earlier and fewer assessed turns are attributed to downstream consequences, while the overall positional structure of productive decisions and late-stage error propagation remains stable.



(a) Shift in credit position.

(b) Credit type composition across training.

Figure 8: **Evolution of reference-guided credit assignment over 40 policy updates.** (a) Median assistant turn at which each credit type occurs, comparing early (steps 6–10) and late (steps 31–35) training. (b) Composition of six credit types aggregated in five-step windows.

Reliability and utility of reference-guided credit. We audit 60 complete failed trajectories from distinct questions, with 20 sampled from each of the early, middle, and late training stages. Two annotators independently label the trajectories without access to the judge's predictions, reaching 86.9% turn-level agreement and a Cohen's κ of 0.77 before adjudication (Table 15). Against the adjudicated labels, providing successful references to the same credit judge improves critical-error F1 from 61.0 to 83.0 and five-class Macro-F1 from 60.3 to 81.1 (Table 16). Productive and corrective F1 also increase from 54.3 and 48.7 to 79.2 and 72.5, respectively, indicating better identification of useful decisions within failed searches. On the 126 eligible MEI interactions, retention precision and recall both reach 85.0, compared with 63.4 and 65.0 without references. To examine the effect on policy optimization, we hold the SFT initialization, MEI checkpoint, credit-assignment rules, and policy-update budget fixed, varying only the judge's access to successful references. In this controlled comparison, reference-guided credit yields gains of 3.3, 1.2, and 4.6 points on BrowseComp-VL, MM-BrowseComp, and FVQA, respectively, raising the three-benchmark average from 59.0 to 62.0 (Table 17). Together, these results support the utility of successful references for both process assessment and policy optimization, with downstream gains observed even when MEI is held fixed.

Table 15: **Human audit of 60 failed trajectories.** Credit counts use adjudicated labels; human agreement and κ use independent before adjudication.

Statistic	Early	Middle	Late	All
Trajectories	20	20	20	60
Annotated policy turns	320	335	345	1,000
Critical-error turns	17	19	19	55
Downstream turns	196	207	212	615
Productive turns	33	35	35	103
Corrective turns	15	16	17	48
Neutral turns	59	58	62	179
Failures with critical errors	13	14	15	42
Eligible MEI interactions	39	42	45	126
Human-retained MEI	25	27	28	80
Human agreement (%)	89.1	86.9	84.9	86.9
Cohen's κ	0.81	0.77	0.74	0.77

Table 16: **Credit-judge agreement with human labels (%).** First-error EM uses 42 failures with a critical error. Reported 95% CIs use paired trajectory bootstrapping.

Metric	w/o ref.	w/ ref.	Δ [95% CI]
Critical-error precision	64.0	**86.3**	+22.3 [11.4, 32.8]
Critical-error recall	58.2	**80.0**	+21.8 [10.9, 32.2]
Critical-error F1	61.0	**83.0**	+22.0 [11.2, 32.5]
First critical-error EM	45.2	**71.4**	+26.2 [10.4, 41.2]
Productive F1	54.3	**79.2**	+24.9 [13.1, 36.4]
Corrective F1	48.7	**72.5**	+23.8 [11.7, 35.8]
Downstream F1	71.2	**86.8**	+15.6 [8.7, 22.6]
Neutral F1	66.4	**84.1**	+17.7 [9.5, 25.8]
Five-class Macro-F1	60.3	**81.1**	+20.8 [12.4, 29.0]
MEI retention precision	63.4	**85.0**	+21.6 [11.0, 31.9]
MEI retention recall	65.0	**85.0**	+20.0 [10.6, 29.5]
Valid-label coverage	86.3	**98.7**	+12.4 [6.5, 18.2]

Table 17: **Effect of successful references on policy optimization with MEI held fixed.** Both RL runs use the same SFT policy initialization, credit-assignment rules, and policy-update budget. Avg. is the unweighted mean over the three benchmarks; averages and gains are computed from the displayed scores.

Policy setting	BC-VL	MM-BC	FVQA	Avg.
Cold-start SFT	60.4	37.8	79.7	59.2
RL, without reference	59.9	38.3	78.7	59.0
RL, with reference	**63.2**	**39.5**	**83.3**	**62.0**
Δ (with − without)	+3.3	+1.2	+4.6	+3.0

F PROMPTS

We provide the system prompt, native tool schemas and definitions, user-message template, and final-turn instruction used by the search policy, followed by the task-specific prompts of MEI and the reference-guided credit judge. The policy operates within 50 assistant turns and addresses loaded images through reusable `img_N` identifiers.

System prompt

```
You are a long-horizon multimodal web-search agent. Solve the user's question by
    interleaving careful reasoning with live-web retrieval and selective visual
    observation. Work for as many steps as needed, but return the best-supported
    final answer within 50 assistant turns.

EVIDENCE DISCIPLINE
1. Track the question's constraints and unresolved evidence gaps. Use each action to
    reduce a specific uncertainty, revise hypotheses when evidence conflicts, and
    verify decisive facts before answering.
2. Distinguish visual evidence from metadata. Titles, snippets, alt text, captions,
    URLs, and helper summaries are discovery signals, not proof of what an image's
    pixels contain.
3. Search results deliberately contain only lightweight references. A candidate image
    becomes visible evidence only after FetchImage loads it into the visual register.
    CropImage creates another reusable visual observation from pixels already in that
    register.
4. Inspect fine-grained visual details when relevant: text, logos, faces, object
    parts, colors, counts, spatial relations, and background clues. Do not claim a
    pixel-level fact about an image that has not been loaded.
5. Prefer focused, complementary searches and source pages. Cross-check ambiguous
    identities and final claims; do not repeatedly issue an identical cached call
    expecting new information.

TOOL WORKFLOW
- TextSearch discovers webpages as title-URL-snippet records.
- WebVisit extracts goal-relevant evidence and up to fifteen lightweight page-image
    references.
- ImageSearch discovers and goal-ranks image candidates without exposing their pixels
    to you.
- ReverseImageSearch searches from up to three loaded image tokens or image URLs and
    returns textual match evidence plus lightweight candidate references; it does not
    load candidate pixels.
- FetchImage loads up to five selected URLs in parallel and appends successful images
    to the visual register in request order.
- CropImage crops one or more loaded img_N observations with normalized 0..1000 [left,
    top, right, bottom] boxes. It processes requested crops in parallel and appends
    them in deterministic request order.

NATIVE ACTION CONTRACT
- Tool definitions are supplied through the API tools field. Use native function calls
    only. Never print XML tags, JSON tool-call wrappers, or a textual imitation of a
    function call.
- On each assistant turn, take exactly one action: either issue exactly one native
    function call or return the final answer as ordinary text. Batch independent
    queries, URLs, images, or boxes inside that one function call when its schema
    permits.
- Think carefully before choosing the action. Do not duplicate a narration of the
    intended function call in answer text.
- Do not fabricate tool results. The environment returns a role=tool message whose
    tool_call_id matches the assistant tool call and reports any newly loaded img_N
    references.
- Tool-returned pixels are registered as img_N only when the environment explicitly
    reports them. Only use existing visual-register tokens.
- When the complete answer is supported, stop calling tools and return only the
    concise exact answer requested by the user as ordinary text. Do not wrap it in
    XML, Markdown, an answer label, or an explanation unless the question explicitly
    asks for one.

Before answering, synthesize the visual and web evidence and check that the visual
    identification and all answer constraints are supported.
```

Figure 9: **System prompt.** The policy interleaves retrieval and selective visual observation, issuing one native function call or a final answer per turn.

Tool schema

TextSearch

```
{
  "type": "function",
  "function": {
    "name": "TextSearch",
    "description": "Search the live web. Each query returns up to 10 lightweight
     title-URL-snippet results; it does not load page or image pixels.",
    "parameters": {
      "type": "object",
      "properties": {
        "query": {
          "type": "array",
          "minItems": 1,
          "items": {"type": "string", "minLength": 1},
          "description": "A non-empty array of focused web queries."
        }
      },
      "required": ["query"],
      "additionalProperties": false
    }
  }
}
```

WebVisit

```
{
  "type": "function",
  "function": {
    "name": "WebVisit",
    "description": "Visit up to five webpages and return goal-conditioned textual evidence
     plus lightweight page-image references. Referenced images are not visually observed
     until FetchImage loads them.",
    "parameters": {
      "type": "object",
      "properties": {
        "url": {
          "type": "array",
          "minItems": 1,
          "items": {"type": "string", "minLength": 1},
          "maxItems": 5,
          "description": "A non-empty array of at most 5 complete http(s) URLs."
        },
        "goal": {
          "type": "string",
          "description": "The non-empty evidence goal for these pages."
        }
      },
      "required": ["url", "goal"],
      "additionalProperties": false
    }
  }
}
```

Figure 10: **Tool schema and definitions for TextSearch and WebVisit.** Each block is a native function object, including its description, parameter types, and required fields.

Tool schema

ImageSearch

```
{
  "type": "function",
  "function": {
    "name": "ImageSearch",
    "description": "Search for online images with up to five text queries. Return at most
     five goal-conditioned candidate references per query; candidate pixels are not placed
     in the policy context. Use FetchImage on a selected direct image URL before making a
     visual claim.",
    "parameters": {
      "type": "object",
      "properties": {
        "query": {
          "type": "array",
          "minItems": 1,
          "items": {"type": "string", "minLength": 1},
          "maxItems": 5,
          "description": "A non-empty array of at most 5 image queries."
        },
        "goal": {
          "type": "string",
          "description": "The visual evidence goal used to rank candidates."
        }
      },
      "required": ["query", "goal"],
      "additionalProperties": false
    }
  }
}
```

ReverseImageSearch

```
{
  "type": "function",
  "function": {
    "name": "ReverseImageSearch",
    "description": "Run reverse-image search on up to three loaded image tokens and/or
     public image URLs. Return goal-conditioned textual matches and lightweight candidate
     references, never implicit candidate pixels.",
    "parameters": {
      "type": "object",
      "properties": {
        "image_token": {
          "type": "array",
          "minItems": 1,
          "items": {"type": "string", "minLength": 1},
          "maxItems": 3,
          "description": "A non-empty array of at most 3 loaded img_N tokens."
        },
        "image_url": {
          "type": "array",
          "minItems": 1,
          "items": {"type": "string", "minLength": 1},
          "maxItems": 3,
          "description": "A non-empty array of at most 3 public http(s) image URLs."
        },
        "goal": {
          "type": "string",
          "description": "The identity or visual-match goal."
        }
      },
      "required": ["goal"],
      "anyOf": [{"required": ["image_token"]}, {"required": ["image_url"]}],
      "additionalProperties": false
    }
  }
}
```

Figure 11: **Tool schema and definitions for ImageSearch and ReverseImageSearch.** Candidate discovery returns lightweight references; it does not implicitly expose candidate pixels to the policy.

Tool schema

FetchImage

```
{
  "type": "function",
  "function": {
    "name": "FetchImage",
    "description": "Load up to five selected image URLs in parallel. Successfully decoded
     pixels are appended to the visual register as new img_N tokens in request order.
     Every successful result record directly includes its new output_ref.",
    "parameters": {
      "type": "object",
      "properties": {
        "urls": {
          "type": "array",
          "minItems": 1,
          "items": {"type": "string", "minLength": 1},
          "maxItems": 5,
          "description": "A non-empty ordered array of at most 5 direct image URL strings."
        }
      },
      "required": ["urls"],
      "additionalProperties": false
    }
  }
}
```

CropImage

```
{
  "type": "function",
  "function": {
    "name": "CropImage",
    "description": "Crop loaded visual observations in parallel. crop_config maps each
     img_N token to one normalized [left, top, right, bottom] box in the 0..1000
     coordinate space or a list of boxes. At most five boxes per image and ten total are
     materialized as reusable new img_N observations in deterministic request order. Every
     successful crop record directly includes its new output_ref.",
    "parameters": {
      "type": "object",
      "properties": {
        "crop_config": {
          "type": "object",
          "minProperties": 1,
          "additionalProperties": {
            "anyOf": [
              {
                "type": "array",
                "items": {"type": "number", "minimum": 0, "maximum": 1000},
                "minItems": 4,
                "maxItems": 4
              },
              {
                "type": "array",
                "minItems": 1,
                "maxItems": 5,
                "items": {
                  "type": "array",
                  "items": {"type": "number", "minimum": 0, "maximum": 1000},
                  "minItems": 4,
                  "maxItems": 4
                }
              }
            ]
          },
          "description": "Mapping from loaded img_N tokens to normalized 0..1000 boxes."
        }
      },
      "required": ["crop_config"],
      "additionalProperties": false
    }
  }
}
```

Figure 12: **Tool schema and definitions for FetchImage and CropImage.** Selected images and image regions become addressable visual observations.

User-message template

```
Question: {question}

Use the available live-web and visual-observation tools to investigate the question.
    Return only the concise exact answer as ordinary text once the evidence chain is
    supported.

The question image is already loaded as img_1. FetchImage and CropImage outputs are
    appended as img_2, img_3, ... in the order reported by the environment.
```

Runtime instruction at turn 50

```
FINAL TURN (50/50): tools are disabled. Using only the evidence already collected,
    return the best-supported concise exact answer now as ordinary text, with no
    answer label or surrounding markup.
```

Figure 13: **User-message template**, shown for one initial image, and runtime instruction at turn 50. Tools are disabled on the final turn and the policy answers from the accumulated evidence.

MEI prompt for WebVisit

System prompt

```
You are a precise multimodal-aware web content analyst. Return a valid JSON object
    only, with no Markdown or commentary. Never include navigation menus, sidebars,
    cookie banners, footers, or advertisements.
```

User-message template

```
Given the webpage content, the user goal, and the list of images found on the page:
    (1) locate the sections, tables, names, dates, numbers, and claims directly
    related to the goal, and explain briefly why the page is relevant; (2) extract
    the strongest supporting content from the article body while preserving
    qualifications and source claims; (3) summarize how that evidence answers or
    contributes to the goal without introducing unsupported facts; and (4) select
    only page images that may help verify or answer the goal, retaining their alt,
    url, and a short caption. If the page contains no direct evidence, return empty
    evidence and an empty image list. Output exactly the keys rational, evidence,
    summary, and relevant_images, with no additional keys.

User goal:
{goal}

Webpage content:
{page_text}

Page image references:
{image_refs}
```

Figure 14: **MEI prompts for WebVisit.** Extracted webpage text, a local goal, and page-image references are transformed into structured evidence, a summary, and relevant image references.

MEI prompt for ImageSearch

System prompt

```
You are a precise visual-search candidate analyst. Inspect pixels, not only titles or
    captions. Return one valid JSON object and no Markdown.
```

User-message template

```
Rank the candidate images for the stated visual goal. For each candidate, report a
    relevance level (high, medium, low, or none) and a concise pixel-grounded
    caption. Metadata can guide inspection but is not visual proof.

Return exactly:
{"rational": string, "candidates": [{"index": integer, "relevance":
    "high|medium|low|none", "caption": string}]}

Search query: {query}
Visual goal: {goal}
Candidate metadata:
{candidates}
```

Figure 15: **MEI prompts for ImageSearch.** Candidate metadata and attached image pixels support goal-conditioned relevance judgments and visual descriptions.

MEI prompt for ReverseImageSearch

System prompt

```
You are a precise reverse-image comparison analyst. Compare visible pixels and return
    one valid JSON object with no Markdown.
```

User-message template

```
The first image is the query observation. Remaining images are reverse-search
    candidates in metadata order. Compare their pixels for the stated goal. Use
    match_state visual_match, partial_match, not_match, or insufficient_evidence.

Return exactly:
{"rational": string, "candidates": [{"index": integer, "match_state":
    "visual_match|partial_match|not_match|insufficient_evidence", "caption": string}]}

Goal: {goal}
Candidate metadata:
{candidates}
```

Figure 16: **MEI prompts for ReverseImageSearch.** The query observation and candidate images are compared in a fixed order to produce structured visual-match judgments.

Credit judge

System prompt

```
You are a causal credit-assignment judge for a multimodal web-research policy.

The candidate trajectory has already been judged to have a wrong final answer. Compare
     its complete trajectory with one independently verified successful reference
     trajectory for the same question. Identify policy-authored assistant turns that
     causally pushed the candidate onto a wrong branch. This is causal process
     supervision, not imitation learning and not another final-answer grading task.

Rules:
1. The reference is one valid solution, not the only valid solution. Never penalize a
     different search query, evidence order, tool choice, or reasoning path merely for
     differing from the reference when it remains valid and useful.
2. Grade assistant decisions only. A failed URL, empty search result, provider
     timeout, or misleading webpage is an environment observation, not itself a policy
     error. The policy may be at fault only when it ignores contrary evidence,
     overclaims what pixels/text establish, repeatedly pursues a disproven hypothesis,
     or makes an unjustified decision from that observation.
3. Distinguish a causal error from its downstream consequences. Mark the decision turn
     that makes the wrong branch materially more likely as causal_error; later turns
     that merely continue the same unrecovered branch are downstream_consequence, not
     additional root errors.
4. Detect recovery. If the candidate later rejects or repairs the earlier mistake and
     again approaches a valid solution, mark that turn recovery. If it subsequently
     makes a new independent error, mark that later turn re_derailment. There may be
     multiple error events only when genuine recovery or independent re-branching
     occurred between them.
5. Use visual evidence only when the corresponding image is embedded in that exact
     trajectory turn. Titles, URLs, captions, snippets, and helper summaries are not
     pixel evidence. img_N numbering is local to each trajectory.
6. A wrong final-answer turn is causal only if it introduces/selects an unsupported
     answer after viable evidence remained. If it merely states the inevitable result
     of an earlier unrecovered mistake, mark it downstream_consequence.
7. Return only significant causal/recovery events and genuinely productive turns.
     Omitted candidate turns are neutral. Use only candidate span IDs shown in the
     trajectory; do not invent or duplicate IDs.

Event meanings:
- causal_error: first currently unrecovered decision that diverts the solution;
- recovery: explicitly repairs an earlier error and restores a viable route;
- re_derailment: a new causal error after recovery, or an independent later error.

List a turn under productive_turns only when it establishes valid evidence or
     materially advances a viable solution. An explicitly productive turn receives
     positive credit even if an earlier error has not yet been repaired; it must not
     be relabeled as a downstream consequence. Productive is not automatically
     recovery: only an explicit recovery event clears the earlier error, and other
     intervening turns retain their chronological downstream decay. Outside an active
     error branch, omitted turns are neutral.

Some terminal tool events are marked observation_delivered_to_policy=false: the tool
     ran, but its result did not fit the remaining policy context and was not
     presented to the policy. Their audit-only response/pixels are intentionally
     absent here. Do not infer those unavailable results or blame the policy for not
     using them. Images may be uniformly enlarged and/or padded for API compatibility;
     the border and img_N label are not evidence and no new scene detail is created.

Return strict JSON only, matching the requested schema. Severity and confidence are
     numbers in [0,1]. Keep rationales concise and evidence-specific.
```

Figure 17: **Credit judge system prompt.**

Credit judge

User-message template

```
===== VERIFIED_REFERENCE TRAJECTORY START =====
{verified_reference_trajectory_with_inline_images}
===== VERIFIED_REFERENCE TRAJECTORY END =====

===== FAILED_CANDIDATE TRAJECTORY START =====
{failed_candidate_trajectory_with_inline_images}
===== FAILED_CANDIDATE TRAJECTORY END =====

Candidate span IDs, in chronological order: {candidate_span_ids}
Return strict JSON only using this compact schema once:
{output_schema}
```

Figure 18: **User-message template.**

Credit judge

Output format

```
{
  "schema_version": "mm-detective-failed-turn-credit-v2",
  "candidate_overall_summary": "brief causal account",
  "events": [
    {
      "span_id": "candidate Txx.Ay ID",
      "type": "causal_error | recovery | re_derailment",
      "severity": "0..1; zero for recovery",
      "confidence": "0..1",
      "error_category": "short snake_case label or empty string",
      "rationale": "concise evidence-specific reason",
      "reference_support": ["reference assistant-turn numbers, if useful"]
    }
  ],
  "productive_turns": [
    {
      "span_id": "candidate Txx.Ay ID",
      "confidence": "0..1",
      "rationale": "valid evidence or useful progress"
    }
  ],
  "confidence": "0..1"
}
```

Figure 19: **Credit judge output format.**

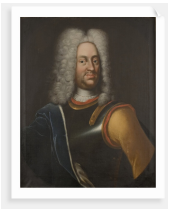
G VISUALIZATION OF INTERLEAVED MULTIMODAL SEARCH

We visualize constructed questions and representative interactions that illustrate the Neos workflow. The examples show how visual observations guide subsequent retrieval and how newly acquired images are inspected to resolve fine-grained questions. Evidence descriptions are abridged, while turn indices and the action sequence are preserved. In the action strips, TS, WV, IS, RS, FI, CI, and A denote TextSearch, WebVisit, ImageSearch, ReverseImageSearch, FetchImage, CropImage, and Answer, respectively.

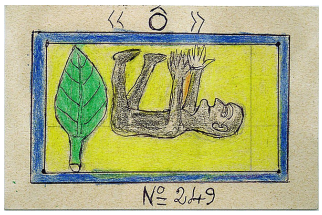
A new visual artifact resolves the answer

Data construction 6-turn reference trajectory

Question. The supplied portrait shows a Baroque-era Hessian landgrave in a gray curled wig, dark breastplate, buff-colored sleeve, and blue mantle; he founded Kassel's Baroque orangery. The lettering on a photographed entrance canopy identifies the eleventh edition of an international contemporary-art exhibition held at that orangery. An Ivorian artist who participated in that edition devised the writing system represented on a reference card whose blue frame encloses a green leaf and a reclining figure. What visible object—specifically, which handwritten numeral—is centered beneath that blue-framed illustration?



Input portrait `img_1`

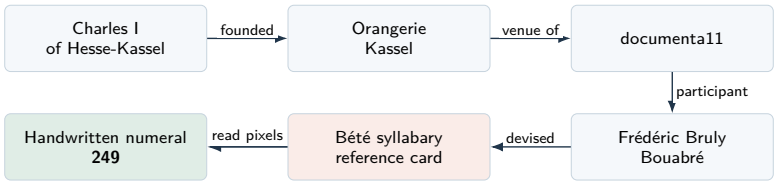


Newly retrieved card T4 `img_2`



Number crop T5 `img_4`

Construction reasoning structure



Reference search trajectory

Turn	Action	Evidence and subsequent decision
1	TextSearch	Connect the landgrave, Kassel exhibition, participating artist, and the Bété syllabary. These relations identify what to search for, not the card's handwritten number.
2	WebVisit	Look for the relevant reference card on pages about the writing system and the artist. The requested card is not yet obtained.
3	ImageSearch	Search for the blue-framed leaf-and-reclining-figure card. The visual-search observation proposes a matching image and transcribes its numeral.
4	FetchImage	Load the candidate card from the art-documentation site. Its subject and inscription are new evidence, distinct from the initial portrait.
5	CropImage	Crop beneath the blue border and verify the handwritten numeral "249" in the returned pixels.

T6 answer: 249.

Complete action sequence

1 TS | 2 WV | 3 IS | 4 FI | 5 CI | 6 A

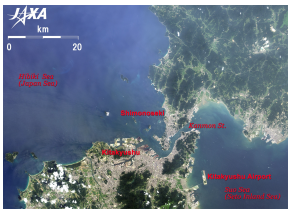
Figure 20: **A new visual artifact resolves the answer.**

Two visual branches converge on one answer

Data construction 7-turn reference trajectory

Question. The labeled satellite view shows one continuous metropolitan area spanning the densely urbanized shores of the narrow strait between Honshu and Kyushu, while a planning map of that metropolitan extent highlights the labeled northern Kyushu municipality in pale yellow and divides it into seven wards. In Japan's national semiprofessional football league, the circular red-and-orange crest featuring a stylized white winged bird and "SINCE 1971" belongs to a football club. Which person was both born in the mapped municipality and a member of the club represented by that crest?

Branch A: geographic evidence



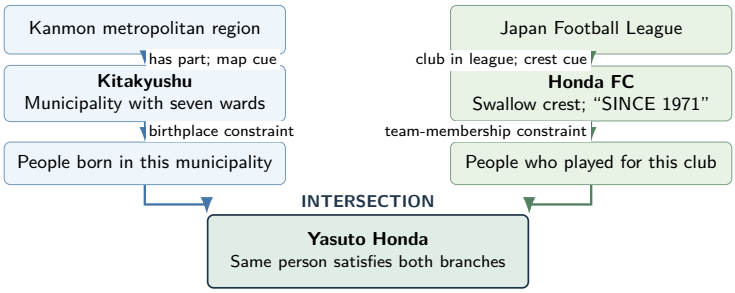
Input satellite view `img_1`

Branch B: club evidence



Retrieved crest T3 `img_2`

Convergent reasoning structure



Reference trajectory: ground, intersect, verify

Turn	Action	Evidence and subsequent decision
1	TextSearch	Read the input's geographic labels to identify **Kitakyushu**. Search the seven-ward clue and the independent red-and-orange, bird-and-1971 crest clue.
2–3	WebVisit; FetchImage	Open the emblem article, retrieve its image URL, and load the crest as `img_2`. The policy confirms **Honda FC** from the returned pixels.
4	TextSearch	Combine both branches in the query: `"Honda FC" soccer player "born in Kitakyushu"`. The retrieved biography identifies **Yasuto Honda**.
5	WebVisit	Verify both relations on the biography: **born in Kitakyushu** and **played for Honda from 1988 to 1992**. Check the club's league on its own page.
6	TextSearch	Test alternative club members. Retrieved biographies place Masaki Tsukano's birth in Yonago and Nobuhiro Takeda's in Osaka, not Kitakyushu.
7	Answer	Return **Yasuto Honda**, supported by the birthplace and team-membership constraints together.

Complete action sequence

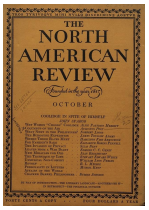
1 TS | 2 WV | 3 FI | 4 TS | 5 WV | 6 TS | 7 A

Figure 21: **Two visual branches converge on one answer.** The geographic branch identifies Kitakyushu and constrains the birthplace, while the emblem branch identifies Honda FC and constrains team membership. Their intersection yields Yasuto Honda. The reference trajectory grounds both branches, combines them in a joint search, and verifies both biographical relations before answering.

A visual checkpoint redirects the search

Data construction 20-turn reference trajectory

Question. The supplied tan title page, framed by a leaf-like ornamental border, displays the three-word masthead of an American literary journal and states that it was founded in 1815. The journal was published in the Midwestern college town named in oversized gold lettering on a municipal mural mounted on a red-brick wall, with sepia images of a church, a historic commercial street, and a house representing three local historic districts. That college town is in the same U.S. state as Cedar Rapids, where a red-brick acute-care hospital is identified by large silver lettering beside a cross logo on its entrance wall. Which person was born in the mural-named college town and died at that hospital?



Input journal `img_1`



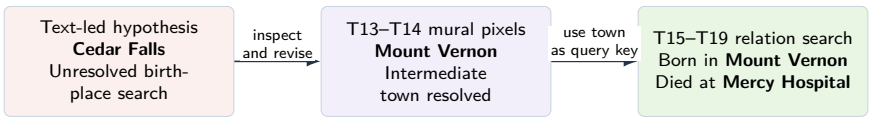
New mural: fetched at T13, cropped at T14 `img_6`



Town-name crop
T14 `img_4`

Visual binding selects the next relational query

The journal has multiple publication places. The mural selects the town used in the birthplace constraint.



Turn	Action	Evidence and subsequent decision
1–6	TextSearch; WebVisit	Identify *The North American Review* and pursue people born in its modern publication town, Cedar Falls. These searches do not resolve the person.
7–12	ImageSearch; WebVisit; TextSearch	A mural candidate suggests Mount Vernon, while hospital images suggest Mercy Medical Center. The policy remains uncertain about the town and continues the Cedar Falls line.
13–14	FetchImage; CropImage	Load a new mural photograph and crop its municipal emblem and lettering. The visible place name resolves the town as **Mount Vernon**, rather than merely confirming the earlier text-led choice.
15–16	TextSearch; WebVisit	Switch to Mount Vernon: verify the journal's Cornell College publication period (1964–1968), then retrieve people associated with that town.
17–19	TextSearch; WebVisit	Check candidate biographies. The usable biography confirms George E. Petrick's **birth in Mount Vernon** and **death at Mercy Hospital in Cedar Rapids**.

T20 final answer: George E. Petrick.

Visual-to-search dependency. The image determines an intermediate *place*; the answer is a *person* obtained through subsequent birth- and death-place retrieval.

Complete action sequence

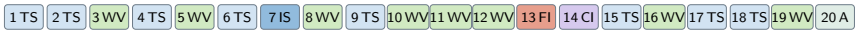


Figure 22: **Intermediate visual disambiguation in a constructed question.** The policy initially follows Cedar Falls, a publication place of the journal. A newly retrieved mural and its town-name crop resolve Mount Vernon instead, redirecting the subsequent birthplace and hospital-death searches that identify George E. Petrick.

From a concert photograph to a recorder's controls

Neos / MMBC 10-turn search trajectory

Question. The band in the picture released a studio album in 1967. During the recording process, how many knobs were there on the front panel of the tape recorder they used?



Input concert photograph / `img_1`



First recorder image / T5 / `img_2`



Actual controls crop / T6 / `img_4`



Additional verification view / T9 / `img_5`

Turn	Action	Evidence and subsequent decision
1	ReverseImageSearch	The returned candidates identify the concert photograph with the Beatles, supplying an entity for subsequent text search.
2–4	TextSearch; WebVisit	Investigate the 1967 recordings. The *Sgt. Pepper* page returns a Studer J37 image reference; its text does not state the requested knob count.
5–6	FetchImage; CropImage	Load the recorder and crop its controls. One crop exposes the central cluster of five rotary knobs, but it does not settle the full-panel count by itself.
7	WebVisit	Abbey Road and equipment pages confirm the recorder model without giving an explicit count. The policy returns to image acquisition.
8–9	ImageSearch; FetchImage	Retrieve additional front-panel photographs. The policy counts five central knobs plus one separate right-side knob.

Answer: 6. **Reference:** 6

Interleaved action sequence

1 RS | 2 TS | 3 WV | 4 TS | 5 FI | 6 CI | 7 WV | 8 IS | 9 FI | 10 A

Case insight. A search result identifies an object, a crop narrows the visual question, and missing textual detail triggers additional image retrieval. The answer is grounded in newly loaded images rather than the concert photograph alone.

Figure 23: **MMBC example**: a concert photograph leads to a recorder model. A partial crop is followed by additional image acquisition to verify the complete front-panel knob count.

Cross-checking an interior match with a local visual cue

Neos / MMSearchPlus 8-turn search trajectory

Question. What is the name of this hotel in Los Angeles?



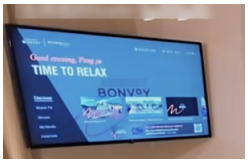
Input kitchenette / `img_1`



Input room / `img_2`



Retrieved property photograph / T5 / `img_6`



Actual TV crop / T6 / `img_7`

Turn	Action	Evidence and subsequent decision
1	ReverseImageSearch	Search from both input images. The returned candidates suggest Residence Inn properties, including Marina del Rey.
2	WebVisit	A hotel listing returns the name "Residence Inn by Marriott Marina del Rey" and its location; this is a textual candidate, not yet a pixel-level match.
3–4	ImageSearch; FetchImage	Retrieve property-specific kitchen photographs and compare floating shelves, cabinetry, the dishwasher, and the wall pattern with `img_1`.
5–6	FetchImage; CropImage	Load another property image, then revisit the original room image. The crop of `img_2` exposes "BONVOY," supporting the Marriott affiliation.
7	TextSearch	Cross-check the property name and Los Angeles-area location before returning the final answer.

Answer: Residence Inn by Marriott Marina del Rey.
Reference: Residence Inn

Interleaved action sequence

1 RS | 2 WV | 3 IS | 4 FI | 5 FI | 6 CI | 7 TS | 8 A

Case insight. The policy alternates candidate discovery, page verification, image comparison, and a late crop of an *initial* image. The visual register makes earlier evidence addressable after several retrieval turns.

Figure 24: **MMSearchPlus example**: candidate hotel interiors are checked against retrieved photographs and source pages. A late crop of an initial image exposes the Bonvoy cue, allowing earlier visual evidence to be revisited during search.

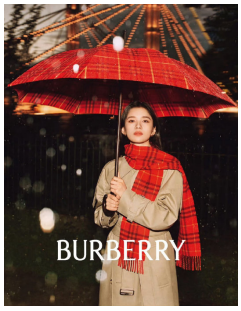
Comparing two regions in a newly retrieved campaign image

Neos / VisBrowse 9-turn search trajectory

Question. The girl in the picture shot a poster for a brand's outerwear series in 2024. In a poster with a rainy background, what item has the same pattern as the umbrella she is holding?



Input photograph / `img_1`



Retrieved campaign image / T5 / `img_3`



Actual comparison crop / T6 / `img_6`

Turn	Action	Evidence and subsequent decision
1–2	ReverseImageSearch; TextSearch	The returned identity candidates lead to a search for the 2024 outerwear campaign and its rainy-background imagery.
3	WebVisit	The official Burberry campaign page supplies campaign evidence and image references. The requested shared pattern still requires inspection.
4–5	ImageSearch; FetchImage	Locate and load campaign photographs, including the rainy scene registered as `img_3`.
6	CropImage	Crop umbrella and scarf regions from `img_3`. Both exhibit the same red check pattern in the recorded visual observation.
7–8	TextSearch	An additional search cross-checks the campaign; the last search returns a gateway timeout. The final answer uses evidence already collected.

Answer: The scarf.
Reference: scarf

Interleaved action sequence

1 RS | 2 TS | 3 WV | 4 IS | 5 FI | 6 CI | 7 TS | 8 TS | 9 A

Case insight. The target relation is between two regions of a *retrieved* image. Entity-level retrieval identifies the relevant campaign, while selective visual inspection grounds the comparison needed for the answer.

Figure 25: **VisBrowse example**: campaign retrieval is followed by fetching and cropping a new image to compare the umbrella and scarf, connecting entity-level search with region-level visual reasoning.

Connecting two visual clues to one historical painting

Neos BC-V3 26-turn trajectory

Question. The city where the tram in Figure 1 operates is the birthplace of a Prussian Field Marshal. This marshal turned the tide of a war. Figure 2 shows a factory of a certain vehicle. The artist who painted this picture also created a work depicting the meeting between this marshal and another marshal after the war. In this meeting painting, which direction are the horses of the two men standing in the center of the picture facing?



Initial tram photograph `img_1`



Initial shipyard painting `img_2`

Visual clue 1: the tram

Reverse search identifies Rostock. Text search links the city to the birthplace of Gebhard Leberecht von Blücher.

Visual clue 2: the shipyard

Reverse search identifies Daniel Maclise's *Peter the Great at Deptford Dockyard*, supplying the artist.

T1–3: the constraints converge

The marshal and artist point to *The Meeting of Wellington and Blücher after the Battle of Waterloo*. ImageSearch now targets this painting, rather than the two initial images.



Retrieved panorama T4 `img_3`

T4: acquire pixels before deciding

Candidate descriptions disagree about the riders and their horses. FetchImage retrieves the panorama and several reproductions. Identifying the painting does not yet answer the spatial question.

Figure 26: **Long-horizon BC-V3 example**, part one. Two initial images provide different entity constraints: a tram leads to a city and a marshal, while a shipyard painting identifies an artist. Their intersection determines a new painting to retrieve and inspect.

Verifying the spatial relation across reproductions

BC-V3 turns 5–26



Retrieved central detail T6 `img_9`



Recorded crop T7 `img_10`

Turn	Action	Evidence and subsequent decision
5–8	CropImage; FetchImage	Inspect the central group, retrieve a clearer reproduction, and crop the mounted figures. The search moves from scene identity to the local relation between the horses.
9–13	WebVisit; TextSearch; CropImage	Check descriptions of the mural and revisit its central region. Pages help identify Wellington and Blücher, but do not settle the requested direction.
14–18	FetchImage; CropImage; WebVisit	One full-image download fails. An alternative Wikimedia reproduction loads at T15, is cropped at T16, and is compared with another print acquired at T18.
19–25	ImageSearch; TextSearch; WebVisit; FetchImage	Additional descriptions remain inconsistent or omit the spatial detail; some further downloads fail. The policy returns to the already acquired visual evidence.



Alternative reproduction, cropped T16 `img_18`



Central riders T16 `img_19`

T26 answer: Left — both central horses face left.

Complete action sequence

1 RS | 2 TS | 3 IS | 4 FI | 5 CI | 6 FI | 7 CI | 8 CI | 9 WV | 10 CI | 11 TS | 12 CI | 13 WV | 14 FI | 15 FI | 16 CI | 17 WV | 18 FI | 19 IS | 20 TS | 21 WV | 22 TS | 23 TS | 24 WV | 25 FI | 26 A

Figure 27: **Long-horizon BC-V3 example**, part two. The 26-turn trajectory alternates between image acquisition, local crops, and source-page checks. Conflicting descriptions and unsuccessful downloads lead the policy back to previously acquired pixels to answer the spatial question.

Disambiguating visually similar species

Neos VDR 28-turn trajectory

Question. What is the official conservation ranking of the yellow flower in the foreground, and in which counties is it primarily found?



Initial photograph `img_1`

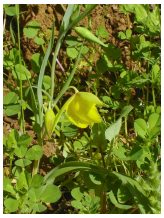


Foreground flower T1 crop `img_2`

Turn	Action	Evidence and subsequent decision
1–3	CropImage; ReverseImageSearch; ImageSearch	Isolate the yellow foreground flower. Reverse search does not identify the exact source, so the policy gathers several species-level alternatives.
4–6	WebVisit; TextSearch	Early taxonomy and status lookups are inconclusive. These tentative names are not yet used as a final identification.
7–12	ReverseImageSearch; WebVisit; ImageSearch; CropImage; TextSearch	Reuse the foreground crop as a new reverse-search query, inspect further candidates, and revisit flower morphology. A later, overly tight crop does not resolve the ambiguity.
13–16	FetchImage; ImageSearch; WebVisit	Acquire comparison photographs and taxon pages. The candidates differ in flower shape, distribution, and conservation status, making the species choice consequential.



Earlier comparison T13 `img_5`



C. pulchellus T17 `img_7`



C. amabilis T18 `img_9`

T17–18: images change the next query

FetchImage brings in the lead photographs of *C. pulchellus* and *C. amabilis*. After comparing the flower forms with the foreground crop, the policy focuses subsequent conservation searches on *C. pulchellus*.

Figure 28: **Long-horizon VDR example**, part one. A foreground crop is reused for reverse search, and comparison photographs help distinguish visually similar species before the policy commits to species-specific factual retrieval.

From visual identification to authoritative evidence

VDR turns 19–28



Observed foreground `img_2`



Selected species comparison `img_7`



Alternative comparison T17 `img_8`

T19: an uninformative page does not end the search

The NatureServe search page returns no species-specific rank or county information. The policy switches to a targeted TextSearch for the California Rare Plant Rank rather than treating the empty page as evidence.

T20–21: locate and read the conservation source

TextSearch locates the California Native Plant Society page *Rare Plant of the Month: Mt. Diablo fairy-lantern*. WebVisit returns two answer-bearing facts for *Calochortus pulchellus*: California Rare Plant Rank **1B.2**, and endemism in **Alameda and Contra Costa counties**.

T22–23: unsuccessful additional image acquisition

The full CNPS photograph and its thumbnail both fail to download. Neither is treated as a new visual observation; the previously fetched comparison photographs remain available.

T24–27: corroborate the assembled answer

Further text queries revisit the county range, species description, and NatureServe status. The identification and requested facts now come from different, linked stages of the search.

T28 answer: California Rare Plant Rank **1B.2**; primarily **Alameda and Contra Costa counties**. The response also reports NatureServe G2 (Imperiled).

Complete action sequence

1 CI	2 RS	3 IS	4 WV	5 TS	6 TS	7 RS	8 WV	9 IS	10 WV	11 CI	12 TS	13 FI	14 IS
15 WV	16 WV	17 FI	18 FI	19 WV	20 TS	21 WV	22 FI	23 FI	24 TS	25 TS	26 TS	27 TS	28 A

Figure 29: **Long-horizon VDR example**, part two. The 28-turn trajectory connects visual identification with conservation evidence. An uninformative database page triggers another search, after which a conservation organization supplies the rank and county range.