POINTS-Seeker: An Open Recipe for Native Multimodal Search Agents with Visual Memory Management

Anonymous ACL submission

Abstract

Large Multimodal Models (LMMs) excel at visual perception but struggle with real-time, knowledge-intensive queries due to their reliance on static parametric knowledge. While multimodal search agents offer a promising solution, developing them from vanilla LMMs presents two major challenges: the lack of open recipes to cultivate search agency from scratch, and the severe context explosion (attention dilution) that occurs during long-horizon multi-turn investigations. In this paper, we address both challenges by developing a **native** multimodal search agent. First, we introduce an open training recipe featuring *Agentic Seeding*, a formative training stage that bootstraps tool-use and planning abilities directly from a non-agentic foundation. Second, to overcome the context bottleneck, we propose **V-Fold**, an adaptive memory management mechanism. V-Fold retains recent interactions as high-fidelity text while "folding" stale historical context into the visual space via rendering, exploiting the model's cross-modal alignment to preserve raw evidence without text token redundancy. Combining these innovations, we present **POINTS-Seeker-8B**, which achieves state-of-the-art performance among models of comparable scale across six multimodal search benchmarks.

1 Introduction

Large Multimodal Models (LMMs) (Bai et al., 2025; Liu et al., 2025c, 2024b; Guo et al., 2025b; Team et al., 2025) have demonstrated impressive proficiency in static perception and reasoning. However, their reliance on static parametric knowledge makes them prone to hallucinations when faced with real-time, knowledge-intensive queries. To address this limitation, recent work has shifted toward agentic search, enabling LMMs to autonomously gather and synthesize web evidence for complex visual question answering (VQA) tasks (Liu et al., 2025d; Yao et al., 2026; Wu et al., 2025; Narayan et al., 2025).

Despite these advancements, developing a robust multimodal search agent remains challenging on two fronts. **First**, prevailing search agents (Wu et al., 2025; Huang et al., 2026b; Narayan et al., 2025; Zhang et al., 2025b) are typically built upon foundations that have already undergone extensive proprietary agentic mid-training. This paradigm formulates multimodal search as a downstream specialization, leaving a critical gap in understanding how to systematically cultivate search agency from a vanilla, non-agentic LMM. **Second**, as agents acquire the ability to conduct multi-turn investigations, they inevitably suffer from context explosion. The accumulation of lengthy web snippets inflates the interaction history, leading to severe "attention dilution" (akin to the "Lost-in-the-Middle" effect (Liu et al., 2024a)). Consequently, the model often fails to locate ground-truth evidence even when it is present in its context (Fig. 3b), rendering complex reasoning prone to failure.

To address these coupled challenges, we introduce an open training recipe and a novel visual memory management mechanism to build a *native* multimodal search agent from scratch. To tackle the first challenge, we design a staged training curriculum powered by an automated data curation pipeline. A key innovation in this recipe is **Agentic Seeding**—a formative stage immediately following pre-training. By exposing the model to dense, long-horizon reasoning and ReAct-style (Yao et al., 2022) trajectories, Agentic Seeding bootstraps the model's intrinsic abilities in tool use, planning, and evidence seeking. Subsequent Supervised Fine-Tuning (SFT) and Reinforcement Learning (RL) stages further refine these behaviors, allowing us to study the complete evolution of a vanilla LMM into a capable search agent.

To tackle the second challenge of context explosion, we propose **V-Fold**, an adaptive, history-aware folding mechanism that uniquely exploits the multimodal nature of the agent. Unlike con-

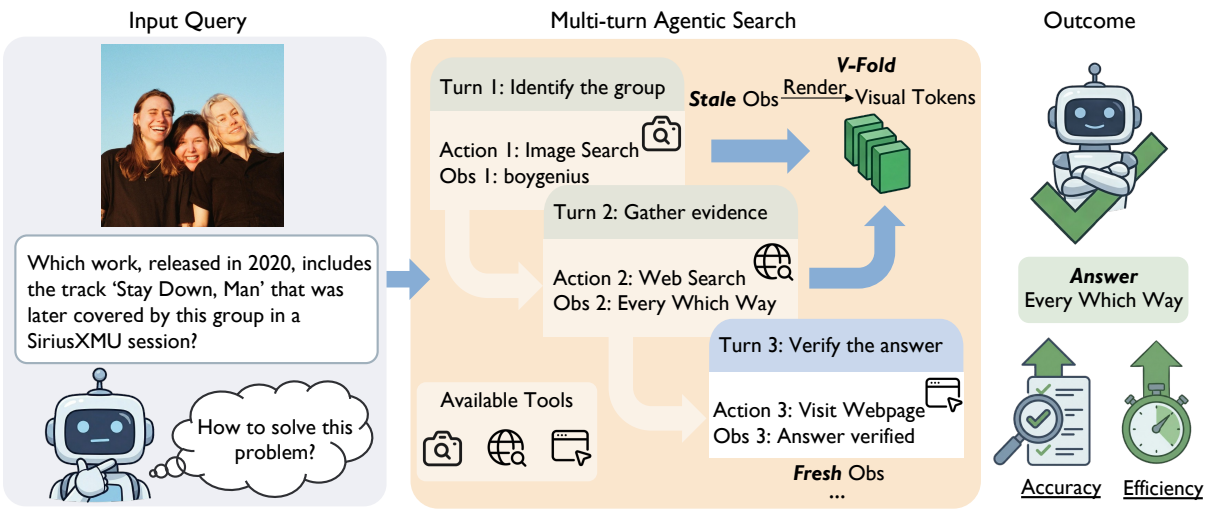


Figure 1: **Overview of POINTS-Seeker.** Our model adaptively interacts with external tools to solve complex, multi-hop VQA tasks. By leveraging V-Fold compression, stale history is rendered into compact visual tokens, effectively bypassing long-context performance degradation while achieving efficiency and reasoning fidelity.

ventional text-only strategies such as naive truncation or summarization (Yu et al., 2025; Liu et al., 2025a) which risk discarding critical cues, V-Fold shifts the information burden from textual processing to visual perception. Once the interaction history reaches a complexity threshold, V-Fold adopts a sliding-window strategy: recent interactions are retained as high-fidelity text for immediate reasoning, while stale historical observations are "folded" into the visual space by rendering them into images. This dual-representation ensures the model can still "see" the raw historical evidence via its native vision-language alignment without being overwhelmed by textual token redundancy. Notably, V-Fold is highly data-efficient; by injecting merely 5k rendered trajectories during SFT, the model internalizes this cross-modal folding capability, supporting robust out-of-the-box deployment.

By integrating this training recipe with V-Fold, we present **POINTS-Seeker-8B**. Extensive experiments across six benchmarks demonstrate that our model substantially outperforms existing agents based on post-hoc fine-tuning, effectively mitigating the bottleneck in long-horizon interactions.

In summary, our main contributions are as follows: (i) we introduce an open, systematic training recipe to cultivate multimodal search agency from a vanilla LMM, highlighting *Agentic Seeding* as a crucial formative stage for bootstrapping tool-use and planning abilities; (ii) we propose *V-Fold*, a novel visual memory management mechanism that adaptively renders stale textual history into compact visual tokens, effectively alleviating attention dilution in long-horizon search without losing raw evidence; (iii) we develop POINTS-Seeker-8B, which achieves state-of-the-art performance among models of comparable scale across six challenging multimodal search benchmarks.

2 Related Work

Retrieval-augmented generation. Retrieval-augmented generation (RAG) grounds LLM responses in retrieved information, mitigating the static nature of model parameters (Karpukhin et al., 2020; Lewis et al., 2020). This paradigm has been widely extended to multimodal settings, especially knowledge-intensive VQA, where external text, images, and structured multimodal data provide evidence for complex reasoning (Yu et al., 2024; Liu et al., 2025b; Chen et al., 2024, 2022; Hu et al., 2023; Yan and Xie, 2024; Mensink et al., 2023; Shi et al., 2026b). However, conventional RAG methods typically rely on static knowledge bases and decouple retrieval from generation, limiting end-to-end optimization. Recent work thus shifts toward autonomous web-search agents that interact with real-time web environments (Chen et al., 2026; Zheng et al., 2025; Li et al., 2026; Chu et al., 2026). In multimodal domain, methods such as MMSearch-R1, DeepMMSearch-R1, and related systems (Wu et al., 2025; Narayan et al., 2025; Wang et al., 2025a; Du et al., 2026; Huang et al., 2026a) further improve search agents via end-to-end RL. In contrast, this paper studies how to systematically cultivate search agency from a vanilla, non-agentic multimodal model through an open recipe and visual memory management.

Agent memory. Continuous interactions often

cause context explosion, motivating various memory management strategies (Chhikara et al., 2025; Wang et al., 2025b; Yan et al., 2025). For example, MemAgent (Yu et al., 2025) proposes maintaining a fixed-length memory that is proactively updated by the model. AgentFold (Ye et al., 2025) introduces a folding mechanism to intelligently fold segments of context during task execution. Recently, cross-modal compression (Shi et al., 2026a; Feng et al., 2026) has emerged as a potent alternative. DeepSeek-OCR (Wei et al., 2025a) proposes that images can serve as an effective medium for compression. Similarly, Glyph (Cheng et al., 2025a) advances a related idea by transforming the long-text modeling problem into a multiple image–text modeling paradigm. Furthermore, AgentOCR (Feng et al., 2026) proposes a self-compression mechanism, in which the model determines the compression ratio. However, memory management in multimodal search remains under-explored. In this paper, we study how history-aware memory folding can preserve critical evidence while mitigating context overload in long-horizon multimodal search.

3 Method

This section starts by formulating the problem in Sec. 3.1, followed by the description of our open training recipe and data curation pipeline in Sec. 3.2. Finally, Sec. 3.3 elaborates on V-Fold, our adaptive visual compression mechanism.

3.1 Problem Formulation

We consider the task of tackling complex visual question answering in a real-world web environment. Given a query (q), which typically consists of a textual question and an image, the policy model ($\mathcal{M}_\theta$) determines the action at each interaction turn. Specifically, it may either invoke an external tool to acquire additional information or generate the final answer directly. The interaction process terminates once the final answer is produced. We define the core components as follows:

- **Action** (a_t). The atomic operation executed at step t. An action may either invoke an external tool (as listed in Sec. 4.1) or generate the answer.
- **Observation** (o_t). The result returned by the tool after executing a_t. Observation may include webpage URLs, titles, and snippets, among other information. Please refer to Sec. 4.1 for details.
- **History** ($\mathcal{H}_t$). The sequence of preceding events up to step t: $\mathcal{H}_t = (q, a_1, o_1, \ldots, a_{t-1}, o_{t-1})$. The policy model ($\mathcal{M}_\theta$) parameterizes an action distribution (π_θ), from which the next action is sampled: $a_t \sim \pi_\theta(\cdot \mid \mathcal{H}_t)$.
- **Answer** (y). The final response is generated upon collecting sufficient evidence or reaching the maximum interaction limit.

3.2 Training Pipeline

Starting from a vision-language model with standard visual-language pre-training (see supplementary material for details), we develop agentic search capabilities through a three-stage training pipeline, as illustrated in Fig. 2. First, *Agentic Seeding* injects general reasoning, planning, and tool-use priors into the model through mid-training. We then curate task-specific agentic search data for the subsequent stages. Second, SFT aligns the model with accurate tool invocation and response formatting using curated trajectories. Finally, RL further optimizes the model to improve correctness, reduce invalid tool calls, and encourage exploration.

3.2.1 Stage I: Agentic Seeding

Agentic Seeding is a mid-training stage designed to cultivate general agentic capabilities before task-specific search training. Rather than directly adapting the model to a particular tool environment, this stage strengthens its reasoning, planning, and preliminary tool-use priors. We train the model with the standard next-token prediction objective on a mixture of the following data:

- **Multi-modal reasoning data**: High-quality VQA data featuring dense reasoning traces to align visual perception with logical reasoning.
- **Text reasoning data**: Diverse logic-intensive datasets, primarily centering on mathematical and coding corpora, that instill system-2 thinking. This curriculum provides the cognitive foundation for the model's internal world model, facilitating complex planning across modalities (Tu et al., 2024; Wei et al., 2025b).
- **Text agentic data with ReAct style**: ReAct-style textual agent data that foster fundamental tool-use logic and planning strategies. By exposing the model to these interaction patterns early on, we facilitate the seamless transfer of agentic priors from the textual to the multimodal domain (Tu et al., 2024; Wei et al., 2025b).

This stage yields a checkpoint with stronger reasoning and planning priors, which is then used as the initialization for the subsequent SFT stage.

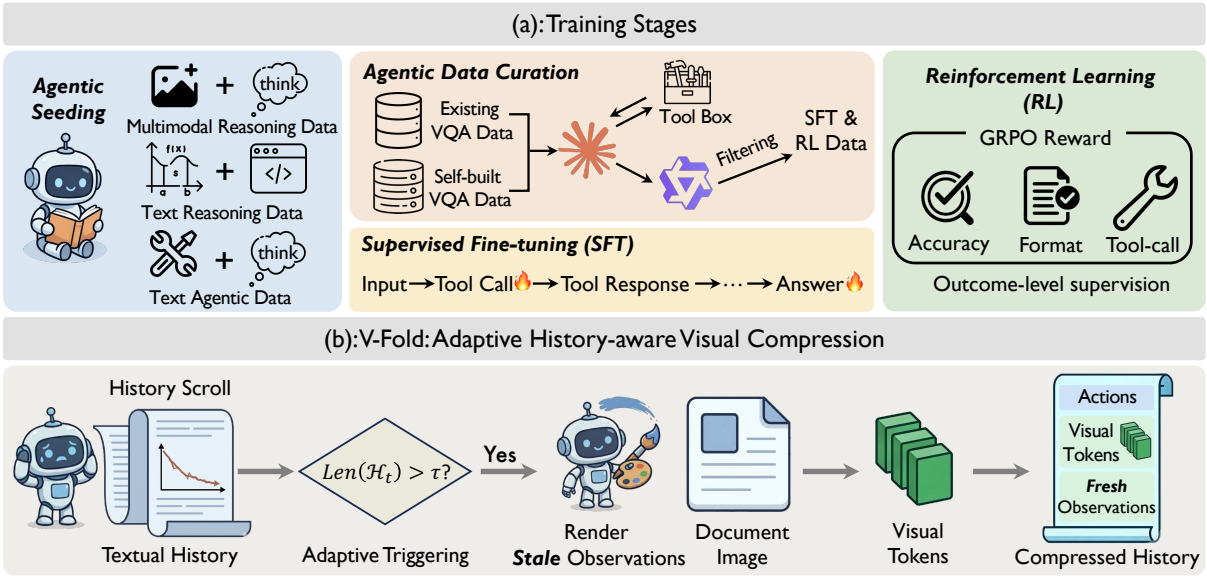


Figure 2: **The framework of our agentic search system. (a)** Training stages, including Agentic Seeding, SFT, and RL. **(b)** V-Fold, which adaptively compresses stale history into visual tokens for efficient long-context reasoning.

3.2.2 Agentic Search Data Curation

After Agentic Seeding, we construct task-specific agentic search data for the following SFT and RL stages. It produces two types of training resources: (i) high-quality tool-use trajectories for SFT; (ii) VQA instances with ground-truth answers for RL.

VQA data construction. We curate high-quality VQA data from search-oriented VQA datasets (*i.e.*, LiveVQA (Fu et al., 2025) and FVQA (Wu et al., 2025)), and by converting complex multi-hop QA datasets (Xia et al., 2025). Our automated pipeline has three key steps: (i) **Entity extraction & selection:** We use Claude to identify visually representable entities from questions. (ii) **Query debiasing:** To ensure retrieval accuracy, we generate context-aware search queries (*e.g.*, appending "New South Wales politician" to "Mike Baird") to resolve potential ambiguities. (iii) **VQA reformulation:** The question is rewritten to implicitly reference the target entity as "the person/object in the image." Finally, we pair the reformulated question with an image retrieved via a search engine to form a coherent, search-intensive VQA instance.

Trajectory generation and filtering. Given the (*image, question, answer*) triplets, we synthesize structured trajectories composed of sequential `<think>`, `<tool_call>`, `<tool_response>`, and `<answer>` segments. Specifically, Claude is employed to generate trajectories with tool-call annotations. Whenever a `<tool_call>` is produced, the corresponding tool is executed, and its `<tool_response>` is appended to the context for the next query, iteratively constructing the full trajectory. We then use `Qwen3-235B-A22B` as an LLM judge to compare the predicted answer against the ground truth, retaining only correct trajectories. This process yields 5,439 high-fidelity trajectories for SFT and 5,339 samples for RL training.

3.2.3 Stage II: Supervised Fine-Tuning

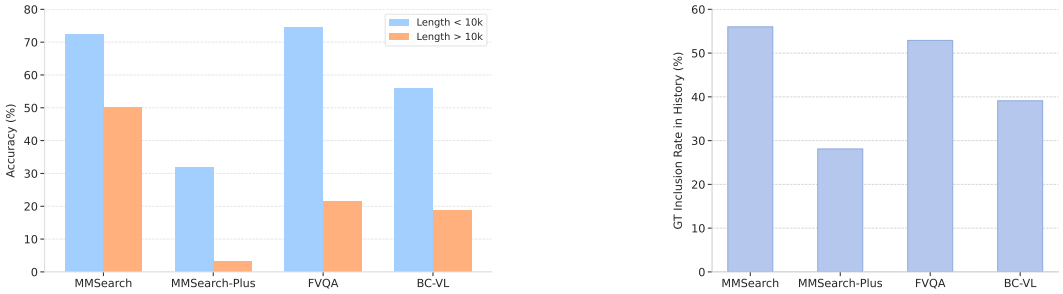
Starting from the Agentic-Seeding checkpoint, we perform supervised fine-tuning on the curated high-quality tool-use trajectories. This stage adapts the model's general reasoning and planning priors to the target agentic search environment. During the cold-start phase, we perform token-level supervision on the curated trajectories. For a training batch of N samples, let $X^i = \{x_1^i, x_2^i, \ldots, x_{L_i}^i\}$ denote the i-th training sample of length L_i, which comprises the query, reasoning thoughts, tool calls, tool responses, and the final answer. The training objective minimizes the negative log-likelihood:

$$\mathcal{L}(\theta) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{L_i} m_j^i \log P_\theta \left(x_j^i \mid x_{<j}^i \right)$$

where $m_j^i \in \{0, 1\}$ is a binary learning mask. We set $m_j^i = 1$ for model-generated tokens, i.e., those in `<think>`, `<tool_call>`, or `<answer>` blocks, and $m_j^i = 0$ for the input query q and tool-returned `<tool_response>` tokens, ensuring the model learns only its own reasoning and actions.

3.2.4 Stage III: Reinforcement Learning

After SFT, the model gains basic tool-use abilities but may still produce redundant tool calls or inconsistent formats. We thus adopt a tool-augmented



(a) Comparison of model accuracy categorized by interaction length.

(b) Ratio of history containing GT in failure cases.

Figure 3: **Motivation for history-aware compression.** (a) Performance degradation in long-context scenarios. (b) The presence of GT in the failure case history indicates that compression must minimize historical information loss.

variant of GRPO (Guo et al., 2025a) to avoid noisy trajectories and encourage exploration.

Rollouts. The policy model iteratively calls tools via pre-defined schemas, appending tool responses to the context. This process halts upon generating an answer or reaching a turn limit (e.g., 10).

Reward. The RL training employs a multi-component reward function to optimize both reasoning quality and interaction integrity:

- **Accuracy** (S_{acc}): A correctness score $\{0, 1\}$ assigned by `Qwen3-235B-A22B` by comparing the model's output with the ground truth.
- **Format** (S_{format}): A binary reward ensuring reasoning and final answers are correctly enclosed within `<think>` and `<answer>` tags, respectively.
- **Tool-call** (S_{tool}): A binary score verifying if tool invocations adhere to the predefined schema and are properly wrapped in `<tool_call>` tags.

The total reward is defined as $R_{\text{total}} = \lambda_{\text{acc}} S_{\text{acc}} + \lambda_{\text{format}} S_{\text{format}} + \lambda_{\text{tool}} S_{\text{tool}}$, where λ_{acc}, λ_{format} and λ_{tool} denote the corresponding reward weights.

3.3 V-Fold: Adaptive Visual Compression

V-Fold addresses long-horizon agentic search by adaptively compressing stale textual observations into compact visual representations. This design is motivated by two empirical findings: as shown in Fig. 3a, model performance drops as interaction length increases; meanwhile, Fig. 3b shows that the ground-truth evidence is often already contained in the histories of failure cases. These observations suggest that failures are largely due to attention dilution over long contexts rather than missing evidence. Thus, instead of truncating or aggressively summarizing historical observations, V-Fold preserves raw evidence by rendering stale observations as images, while retaining the most recent interactions in text to maintain local reasoning continuity.

Specifically, the history at step t is denoted as $\mathcal{H}_t = (q, a_1, o_1, \ldots, a_{t-1}, o_{t-1})$. We define a triggering criterion $\mathbb{1}_{\text{compress}}$ based on a predefined token threshold τ:

$$\mathbb{1}_{\text{compress}} = \begin{cases} 1, & \text{if Len}(\mathcal{H}_t) > \tau \\ 0, & \text{otherwise} \end{cases}$$

When $\mathbb{1}_{\text{compress}} = 1$, the compression mechanism is triggered. To minimize historical information loss while keeping reasoning logic intact, we maintain all previous actions $\{a_i\}$ in textual form but partition the observations into a "stale" set $\mathcal{O}_{stale} = \{o_i\}_{i=1}^{t-k-1}$ and a "fresh" set $\mathcal{O}_{fresh} = \{o_j\}_{j=t-k}^{t-1}$, where k is the number of most recent turns retained as text. For $\mathcal{O}_{stale}$, we render each o_i individually into an image via a rendering function $\mathcal{F}_{\text{render}}$, and map it to visual tokens:

$$\tilde{o}_i = \phi(E_v(\mathcal{F}_{\text{render}}(o_i)))$$

where E_v is the visual encoder and ϕ is a vision projection layer. The compressed history $\tilde{\mathcal{H}}_t$ thus accurately maintains the interleaved structure by substituting the lengthy text of stale observations with their compact visual counterparts:

$$\tilde{\mathcal{H}}_t = \left(q,\ \{a_i, \tilde{o}_i\}_{i=1}^{t-k-1},\ \{a_j, o_j\}_{j=t-k}^{t-1}\right)$$

This strategy effectively transforms the $O(L)$ total token length of historical tool responses into an $O(N_v)$ length (where N_v is the number of visual patches), ensuring $N_v \ll L$. On average, rendering reduces the token budget of each stale observation by about 50%. The rendering parameters are provided in the supplementary materials.

Discussions. V-Fold features two primary advantages: (i) Adaptive Compression: Our context-aware sliding window selectively renders stale observations ($\mathcal{O}_{stale}$). Preserving the most recent k

Model	MMSearch	MMSearch+	FVQA	SimpleVQA	LiveVQA	BC-VL
Direct Answer						
GPT-5	33.3	19.1	57.3	61.7	57.5	47.2
Gemini-2.5 Flash	30.4	8.1	47.7	55.5	51.0	37.1
Qwen3-VL-8B-Instruct	15.2	3.2	28.0	42.9	41.0	25.1
RAG Workflow						
GPT-4o	24.1	-	-	61.6	34.0	13.4
Gemini-2.5 Flash	43.9	-	-	68.6	41.3	13.0
Qwen3-VL-8B-Instruct	47.3	-	53.6	62.3	39.3	29.3
Agent Workflow						
GPT-5	63.7	17.2	69.0	-	73.3	46.1
Gemini-2.5 Flash	64.0	19.9	68.0	-	73.0	44.6
Claude-4-Sonnet	67.2	23.1	69.0	-	69.7	**48.6**
Agentic Search MLLM						
MMSearch-R1-7B (Wu et al., 2025)	53.8	-	58.4	57.4	48.4	-
DeepMMSearch-R1-7B (Narayan et al., 2025)	-	-	-	55.9	-	-
WebWatcher-32B (Geng et al., 2025)	55.3	-	-	59.0	58.7	27.0
DeepEyesV2 (Hong et al., 2025)	63.7	-	60.6	59.4	-	-
Qwen3-VL-8B-Instruct (Bai et al., 2025)	58.5	10.4	64.0	65.8	60.5	30.8
Skywork-R1V4-30B-A3B (Zhang et al., 2025b)	66.1	-	67.2	-	-	38.4
SenseNova-MARS-8B (Chng et al., 2025)	67.8	-	67.1	**70.2**	56.2	-
MM-DeepResearch-8B (Yao et al., 2026)	67.8	-	69.2	65.9	65.0	37.9
Vision-DeepResearch-8B (Huang et al., 2026b)	69.6	20.4	64.7	-	76.7	42.6
POINTS-Seeker-8B	70.8	**25.2**	**71.2**	68.8	77.7	44.4
Qwen3-VL-8B-Instruct + Our method	**71.3**	24.3	71.1	69.2	**80.5**	43.9

Table 1: **Comparison with state-of-the-art models.**

interactions as text maintains the high-precision environment necessary for local reasoning continuity. (ii) Data-Efficient Visual Adaptation: Instead of relying on extensive RL or massive datasets, our approach is highly data-efficient. By incorporating a mere 5k fully rendered trajectories into the SFT mixture, the model learns to interpret visual history and seamlessly generalizes to our adaptive strategy without complex post-hoc alignment.

4 Experiments

4.1 Experimental Setup

Training datasets. For the training dataset, the text reasoning data is sourced from AM-DeepSeek-R1-Distilled-1.4M (Zhao et al., 2025), while the text agentic data is obtained from Toucan-1.5M (Xu et al., 2025). The multi-modal reasoning datasets are derived from Reason-RFT (Tan et al., 2025) and VisualWebInstruct (Jia et al., 2025). The dataset used for SFT is drawn from the training set of LiveVQA and FVQA, supplemented with a portion of general-domain data from Honey-Data (Zhang et al., 2025a), whereas the RL dataset consists of LiveVQA and our self-constructed VQA samples. Detailed information about these training datasets is provided in the supplementary materials.

Evaluation benchmarks. We evaluate on six benchmarks: MMSearch (Jiang et al., 2025), MMSearch-Plus (Tao et al., 2025), FVQA (Wu et al., 2025), SimpleVQA (Cheng et al., 2025b), LiveVQA (Fu et al., 2025), and BrowseComp-VL (BC-VL) (Geng et al., 2025). Following prior protocols (Wu et al., 2025; Huang et al., 2026b), we filter MMSearch for image-containing QA pairs, MMSearch-Plus for single-image samples, and SimpleVQA for English instances. The remaining benchmarks use full datasets.

Tools. Our agent utilizes three tools: web search, image search, and visit. (i) For web search, we use the Google Custom Search API or Serper API to retrieve the titles, snippets, and URLs of the top-10 relevant webpages. (ii) For image search, we use the Serper API to obtain the top-5 webpage URLs, parse them with Crawl4AI (UncleCode, 2024), and summarize goal-relevant content using Qwen3-235B-A22B (Yang et al., 2025). (iii) The visit tool follows a similar pipeline, extracting webpage content with Crawl4AI and generating a goal-focused summary with Qwen3-235B-A22B.

Implementation details. POINTS-Seeker initial-

Seeding	SFT	RL	V-Fold	MMSearch	MMSearch+	FVQA	SimpleVQA	LiveVQA	BC-VL
POINTS-Seeker									
✘	✔	✘	✘	63.2	18.0	67.9	65.1	70.8	32.3
✔	✔	✘	✘	66.1	23.0	68.9	66.4	76.8	39.8
✔	✔	✔	✘	69.6	23.4	71.2	68.0	77.3	42.6
✔	✔	✔	✔	**70.8**	**25.2**	**71.2**	**68.8**	**77.7**	**44.4**
Qwen3-VL-8B-Instruct									
-	✘	✘	✘	58.5	10.4	64.0	65.8	60.5	30.8
-	✔	✘	✘	69.0	23.0	68.9	67.6	76.9	42.4
-	✔	✔	✘	70.2	23.0	70.7	68.5	80.2	42.6
-	✔	✔	✔	**71.3**	**24.3**	**71.1**	**69.2**	**80.5**	**43.9**

Table 2: **Ablation study on our core design.**

Method	Dataset	Acc	Resp. Len.
w/o V-Fold	BC-VL	42.6	9530
V-Fold	BC-VL	44.4$_{(1.8\uparrow)}$	7353$_{(22.8\%\downarrow)}$
w/o V-Fold	MMSearch+	23.4	9178
V-Fold	MMSearch+	25.2$_{(1.8\uparrow)}$	6871$_{(25.1\%\downarrow)}$

Table 3: **Impact of V-Fold on length and accuracy.**

Method	FVQA-H	LiveVQA-H	BC-VL-H
No Compression	48.0	48.8	21.3
Discard History	52.0	50.2	21.3
Summarization	50.0	48.8	22.7
Image Rendering	48.0	44.4	17.3
Our V-Fold	54.0	50.7	26.7

Table 4: **Comparison under long-horizon settings.**

izes its ViT and LLM with Qwen2-VL-ViT (Wang et al., 2024) and Qwen3-8B-Base (Yang et al., 2025). For both Agentic Seeding and SFT, the global batch size is set to 64, and the learning rate is 5×10^{-5}. Batch samples are flattened and concatenated into sequences up to 32k tokens. For RL training, we use a global batch size of 64 and perform 8 rollouts per sample. The learning rate is set at 1×10^{-6}. Additionally, the λ_{acc}, λ_{format}, and λ_{tool} are set to 0.7, 0.2, and 0.1, respectively. We empirically set τ to 8k and k to 2. Further details are provided in the supplementary materials.

4.2 Comparison with State-of-the-art

We evaluate POINTS-Seeker against mainstream models under four settings: (i) **Direct answer**, where the model answers without external search; (ii) **RAG workflow**, where relevant information is first retrieved and then used for answer generation; (iii) **Agent workflow**, which follows a ReAct-style agent paradigm (Huang et al., 2026b); and (iv) **Agentic search MLLM**, where the system prompt specifies the task and available tools, and the model autonomously decides when and which tools to invoke before producing the final answer. As shown in Table 1, we can draw the following conclusions. (i) The agent workflow paradigm significantly outperforms the direct answer setting, indicating the tool invocation is essential for achieving strong performance on these benchmarks. (ii) Our model demonstrates superior performance across these benchmarks, substantially outperforming several existing open-source agentic search MLLMs. Moverover, it either surpasses or achieves performance comparable to proprietary agent workflow models. (iii) We further evaluate the generality of our approach on Qwen3-VL-8B-Instruct by applying V-Fold and the training pipeline without the agentic seeding stage. The consistent improvements highlight the robustness of our method across different backbone models.

4.3 Ablations and Analysis

Ablation on our core design. To validate the effectiveness of our training strategies and V-Fold, we conduct ablation experiments in Table 2, yielding several key observations. (i) Agentic seeding provides consistent performance gains across all benchmarks, demonstrating that injecting agentic-related data during the model's formative stages is essential for long-horizon interaction. (ii) SFT on high-quality trajectory data substantially improves basic agentic abilities, enabling more accurate tool invocation and strong performance on relatively simple benchmarks such as SimpleVQA and FVQA. (iii) RL further reduces repetitive tool calls and improves output-format reliability over the SFT-only baseline, with case studies in the supplementary material. (iv) V-Fold provides additional improvements, especially on challenging benchmarks such as MMSearch-Plus and BrowseComp-VL, suggesting its effectiveness in handling complex long-horizon problems.

Method	MMBench	MMStar	MMMU	MathVista	Hallu.	AI2D	OCRBench	MMVet	Avg.
POINTS-Baseline	74.9	55.5	52.4	75.7	53.2	81.6	85.1	67.8	68.3
POINTS-Seeker	79.1	59.7	52.6	73.7	54.2	84.8	86.7	73.9	70.6

Table 5: **Performance on general VQA benchmarks.**

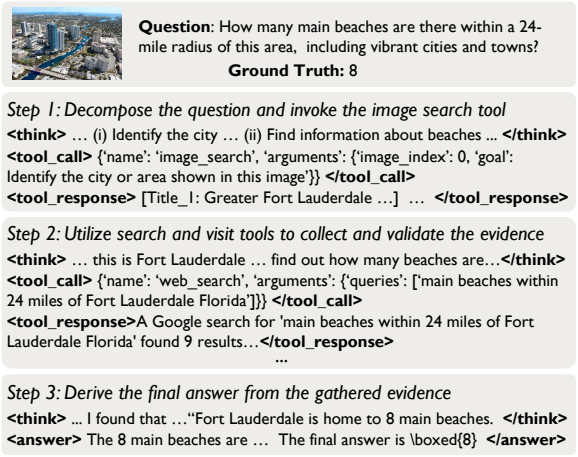


Figure 4: **Visualization of POINTS-Seeker.** Additional qualitative results are in the supplementary materials.

In-depth analysis of V-Fold. We further analyze how V-Fold alleviates bottlenecks in long-horizon agentic search. As shown in Table 3, V-Fold achieves a dual optimization of efficiency and reasoning fidelity: (i) On BC-VL and MMSearch-Plus, it reduces the average response length by 22.8% and 25.1%, respectively, mitigating token overflow. (ii) V-Fold's history-aware rendering preserves raw evidence from "stale" observations in a visual format. By folding extensive textual history into compact visual tokens, it enables the model to more effectively utilize historical cues while bypassing the performance degradation.

V-Fold under long-horizon settings. To evaluate V-Fold in harder long-horizon scenarios, we use Claude-Sonnet-4-5 to construct hard subsets from each benchmark, selecting VQA instances requiring over ten interaction rounds. We compare V-Fold with three baselines: *Discard History*, which keeps only the initial question and recent interactions after exceeding a threshold; *Summarization*, which compresses history with Qwen3-235B-A22B; *Image Render*, which converts all past observations into images. As shown in Table 4, V-Fold outperforms these alternatives. Notably, it outperforms Image Render, highlighting the advantages of our adaptive mechanism and sliding-window strategy. This shows that selectively maintaining the recent textual context yields a better balance between compression and information retention.

τ	4k	8k	12k	8k	8k	8k
k	2	2	2	0	2	4
Acc.	25.2	25.2	23.4	23.4	25.2	25.2

Table 6: **Ablation study on hyperparameters.**

Impact on general visual capabilities. A common concern in injecting specialized agentic skills into LMMs is the potential degradation of their basic perceptual abilities. To examine this, we evaluate POINTS-Seeker on general VQA benchmarks and compare it with POINTS-Baseline, which removes the Agentic Seeding stage and all agentic search trajectories during SFT, using only general multimodal data. As shown in Table 5, POINTS-Seeker outperforms the baseline by an average of 2.3 points. This suggests that internalizing autonomous inquiry capabilities in early training does not harm general visual understanding; instead, agentic search and reasoning skills can improve performance on standard non-agentic visual tasks.

Ablation on hyperparameters of V-Fold. As shown in Table 6, we ablate the hyperparameters of V-Fold on MMSearch-Plus. The results show that an overly large τ leads to a performance drop, as it triggers compression less frequently. We also find that setting k too small degrades performance, highlighting the importance of preserving recent textual context. We finally set τ to 8k and k to 2.

5 Conclusion

In this paper, we present **POINTS-Seeker-8B**, a multimodal search agent built from a vanilla, non-agentic vision-language model. With a dedicated dataset and training recipe, the model acquires essential agentic search capabilities. To handle extended interaction histories, we further propose **V-Fold**, a history-aware memory folding mechanism that preserves stale but critical evidence in the visual space and alleviates attention dilution. Experiments on six benchmarks show that POINTS-Seeker-8B consistently achieves state-of-the-art performance among models of the same size, demonstrating the effectiveness of our framework. We hope our work provides a practical foundation for developing native agentic LMMs capable of active exploration and reliable reasoning.

Limitations

Despite its effectiveness, our approach has several limitations. First, the current context window is constrained to 32k tokens with a maximum of 10 interaction turns, which may limit performance in extremely long-horizon reasoning tasks. Second, our toolset is currently restricted to three core search tools; incorporating more specialized APIs, such as Google Scholar or Google Maps, could further broaden the model's utility. Finally, due to search API quota constraints, the scale of our RL training was limited. Future work will focus on scaling the reinforcement learning process and expanding the tool-use diversity to further enhance the model's autonomous search capabilities.

References

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, and 1 others. 2025. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*.

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

Wenhu Chen, Hexiang Hu, Xi Chen, Pat Verga, and William Cohen. 2022. Murag: Multimodal retrieval-augmented generator for open question answering over images and text. In *EMNLP*.

Zhanpeng Chen, Chengjin Xu, Yiyan Qi, and Jian Guo. 2024. Mllm is a strong reranker: Advancing multimodal retrieval-augmented generation via knowledge-enhanced reranking and noise-injected training. *arXiv preprint arXiv:2407.21439*.

Jiale Cheng, Yusen Liu, Xinyu Zhang, Yulin Fei, Wenyi Hong, Ruiliang Lyu, Weihan Wang, Zhe Su, Xiaotao Gu, Xiao Liu, and 1 others. 2025a. Glyph: Scaling context windows via visual-text compression. *arXiv preprint arXiv:2510.17800*.

Xianfu Cheng, Wei Zhang, Shiwei Zhang, Jian Yang, Xiangyuan Guan, Xianjie Wu, Xiang Li, Ge Zhang, Jiaheng Liu, Yuying Mai, and 1 others. 2025b. Simplevqa: Multimodal factuality evaluation for multimodal large language models. In *ICCV*.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. 2025. Mem0: Building production-ready ai agents with scalable long-term memory. *arXiv preprint arXiv:2504.19413*.

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

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

Cheng Cui, Ting Sun, Manhui Lin, Tingquan Gao, Yubo Zhang, Jiaxuan Liu, Xueqing Wang, Zelun Zhang, Changda Zhou, Hongen Liu, and 1 others. 2025. Paddleocr 3.0 technical report. *arXiv preprint arXiv:2507.05595*.

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

Lang Feng, Fuchao Yang, Feng Chen, Xin Cheng, Haiyang Xu, Zhenglin Wan, Ming Yan, and Bo An. 2026. Agentocr: Reimagining agent history via optical self-compression. *arXiv preprint arXiv:2601.04786*.

Mingyang Fu, Yuyang Peng, Dongping Chen, Zetong Zhou, Benlin Liu, Yao Wan, Zhou Zhao, Philip S Yu, and Ranjay Krishna. 2025. Seeking and updating with live visual knowledge. *arXiv preprint arXiv:2504.05288*.

Xinyu Geng, Peng Xia, Zhen Zhang, Xinyu Wang, Qiuchen Wang, Ruixue Ding, Chenxi Wang, Jialong Wu, Yida Zhao, Kuan Li, and 1 others. 2025. Webwatcher: Breaking new frontier of vision-language deep research agent. *arXiv preprint arXiv:2508.05748*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, and 1 others. 2025a. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang, Jianyu Jiang, Jiawei Wang, and 1 others. 2025b. Seed1. 5-vl technical report. *arXiv preprint arXiv:2505.07062*.

Jack Hong, Chenxiao Zhao, ChengLin Zhu, Weiheng Lu, Guohai Xu, and Xing Yu. 2025. Deepeyesv2: Toward agentic multimodal model. *arXiv preprint arXiv:2511.05271*.

Ziniu Hu, Ahmet Iscen, Chen Sun, Zirui Wang, Kai-Wei Chang, Yizhou Sun, Cordelia Schmid, David A

Ross, and Alireza Fathi. 2023. Reveal: Retrieval-augmented visual-language pre-training with multi-source multimodal knowledge memory. In *CVPR*.

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

Wenxuan Huang, Yu Zeng, Qiuchen Wang, Zhen Fang, Shaosheng Cao, Zheng Chu, Qingyu Yin, Shuang Chen, Zhenfei Yin, Lin Chen, and 1 others. 2026b. Vision-deepresearch: Incentivizing deepresearch capability in multimodal large language models. *arXiv preprint arXiv:2601.22060*.

Yiming Jia, Jiachen Li, Xiang Yue, Bo Li, Ping Nie, Kai Zou, and Wenhu Chen. 2025. Visualwebinstruct: Scaling up multimodal instruction data through web search. In *EMNLP*.

Dongzhi Jiang, Renrui Zhang, Ziyu Guo, Yanmin Wu, Pengshuo Qiu, Pan Lu, Zehui Chen, Guanglu Song, Peng Gao, Yu Liu, and 1 others. 2025. Mmsearch: Unveiling the potential of large models as multimodal search engines. In *ICLR*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *EMNLP*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *NIPS*.

Guankai Li, Jiabin Chen, Yi Xu, Xichen Zhang, and Yuan Lu. 2026. Hypereyes: Dual-grained efficiency-aware reinforcement learning for parallel multimodal search agents. *Preprint*, arXiv:2605.07177.

Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, and 1 others. 2025a. Deepseek-v3. 2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556*.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024a. Lost in the middle: How language models use long contexts. In *ACL*.

Yikun Liu, Yajie Zhang, Jiayin Cai, Xiaolong Jiang, Yao Hu, Jiangchao Yao, Yanfeng Wang, and Weidi Xie. 2025b. Lamra: Large multimodal model as your advanced retrieval assistant. In *CVPR*.

Yuan Liu, Le Tian, Xiao Zhou, Xinyu Gao, Kavio Yu, Yang Yu, and Jie Zhou. 2024b. Points1. 5: Building a vision-language model towards real world applications. *arXiv preprint arXiv:2412.08443*.

Yuan Liu, Zhongyin Zhao, Le Tian, Haicheng Wang, Xubing Ye, Yangxiu You, Zilin Yu, Chuhan Wu, Zhou Xiao, Yang Yu, and 1 others. 2025c. Points-reader: Distillation-free adaptation of vision-language models for document conversion. In *EMNLP*.

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

Thomas Mensink, Jasper Uijlings, Lluis Castrejon, Arushi Goel, Felipe Cadar, Howard Zhou, Fei Sha, André Araujo, and Vittorio Ferrari. 2023. Encyclopedic vqa: Visual questions about detailed properties of fine-grained categories. In *ICCV*.

Kepan Nan, Rui Xie, Penghao Zhou, Tiehan Fan, Zhenheng Yang, Zhijie Chen, Xiang Li, Jian Yang, and Ying Tai. 2024. Openvid-1m: A large-scale high-quality dataset for text-to-video generation. *arXiv preprint arXiv:2407.02371*.

Kartik Narayan, Yang Xu, Tian Cao, Kavya Nerella, Vishal M Patel, Navid Shiee, Peter Grasch, Chao Jia, Yinfei Yang, and Zhe Gan. 2025. Deepmmsearch-r1: Empowering multimodal llms in multimodal web search. *arXiv preprint arXiv:2510.12801*.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, and 1 others. 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. In *NIPS*.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*.

Yaorui Shi, Shugui Liu, Yu Yang, Wenyu Mao, Yuxin Chen, Qi Gu, Hui Su, Xunliang Cai, Xiang Wang, and An Zhang. 2026a. Memocr: Layout-aware visual memory for efficient long-horizon reasoning. *arXiv preprint arXiv:2601.21468*.

Yudi Shi, Shangzhe Di, Qirui Chen, Qinian Wang, Jiayin Cai, Xiaolong Jiang, Yao Hu, and Weidi Xie. 2026b. Weaver: End-to-end agentic system training for video interleaved reasoning. *arXiv preprint arXiv:2602.05829*.

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

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*.

Huajie Tan, Yuheng Ji, Xiaoshuai Hao, Minglan Lin, Pengwei Wang, Zhongyuan Wang, and Shanghang Zhang. 2025. Reason-rft: Reinforcement fine-tuning for visual reasoning. *arXiv e-prints*, pages arXiv–2503.

Xijia Tao, Yihua Teng, Xinxing Su, Xinyu Fu, Jihao Wu, Chaofan Tao, Ziru Liu, Haoli Bai, Rui Liu, and Lingpeng Kong. 2025. Mmsearch-plus: Benchmarking provenance-aware search for multimodal browsing agents. *arXiv preprint arXiv:2508.21475*.

Kimi Team, Angang Du, Bohong Yin, Bowei Xing, Bowen Qu, Bowen Wang, Cheng Chen, Chenlin Zhang, Chenzhuang Du, Chu Wei, and 1 others. 2025. Kimi-vl technical report. *arXiv preprint arXiv:2504.07491*.

Jianhong Tu, Zhuohao Ni, Nicholas Crispino, Zihao Yu, Michael Bendersky, Beliz Gunel, Ruoxi Jia, Xin Liu, Lingjuan Lyu, Dawn Song, and 1 others. 2024. Mlan: Language-based instruction tuning preserves and transfers knowledge in multimodal language models. *arXiv preprint arXiv:2411.10557*.

UncleCode. 2024. Crawl4ai: Open-source llm friendly web crawler & scraper. https://github.com/unclecode/crawl4ai.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, and 1 others. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Qiuchen Wang, Ruixue Ding, Yu Zeng, Zehui Chen, Lin Chen, Shihang Wang, Pengjun Xie, Fei Huang, and Feng Zhao. 2025a. Vrag-rl: Empower vision-perception-based rag for visually rich information understanding via iterative reasoning with reinforcement learning. *arXiv preprint arXiv:2505.22019*.

Yu Wang, Ryuichi Takanobu, Zhiqi Liang, Yuzhen Mao, Yuanzhe Hu, Julian McAuley, and Xiaojian Wu. 2025b. Mem-{\alpha}: Learning memory construction via reinforcement learning. *arXiv preprint arXiv:2509.25911*.

Haoran Wei, Yaofeng Sun, and Yukun Li. 2025a. Deepseek-ocr: Contexts optical compression. *arXiv preprint arXiv:2510.18234*.

Yana Wei, Liang Zhao, Jianjian Sun, Kangheng Lin, Jisheng Yin, Jingcheng Hu, Yinmin Zhang, En Yu, Haoran Lv, Zejia Weng, and 1 others. 2025b. Open vision reasoner: Transferring linguistic cognitive behavior for visual reasoning. *arXiv preprint arXiv:2507.05255*.

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

Ziyi Xia, Kun Luo, Hongjin Qian, and Zheng Liu. 2025. Open data synthesis for deep research. *arXiv preprint arXiv:2509.00375*.

Zhangchen Xu, Adriana Meza Soria, Shawn Tan, Anurag Roy, Ashish Sunil Agrawal, Radha Poovendran, and Rameswar Panda. 2025. Toucan: Synthesizing 1.5 m tool-agentic data from real-world mcp environments. *arXiv preprint arXiv:2510.01179*.

Sikuan Yan, Xiufeng Yang, Zuchao Huang, Ercong Nie, Zifeng Ding, Zonggen Li, Xiaowen Ma, Jinhe Bi, Kristian Kersting, Jeff Z Pan, and 1 others. 2025. Memory-r1: Enhancing large language model agents to manage and utilize memories via reinforcement learning. *arXiv preprint arXiv:2508.19828*.

Yibin Yan and Weidi Xie. 2024. Echosight: Advancing visual-language models with wiki knowledge. In *Findings of EMNLP*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

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

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. In *ICLR*.

Rui Ye, Zhongwang Zhang, Kuan Li, Huifeng Yin, Zhengwei Tao, Yida Zhao, Liangcai Su, Liwen Zhang, Zile Qiao, Xinyu Wang, and 1 others. 2025. Agentfold: Long-horizon web agents with proactive context management. *arXiv preprint arXiv:2510.24699*.

Hongli Yu, Tinghong Chen, Jiangtao Feng, Jiangjie Chen, Weinan Dai, Qiying Yu, Ya-Qin Zhang, Wei-Ying Ma, Jingjing Liu, Mingxuan Wang, and 1 others. 2025. Memagent: Reshaping long-context llm with multi-conv rl-based memory agent. *arXiv preprint arXiv:2507.02259*.

Shi Yu, Chaoyue Tang, Bokai Xu, Junbo Cui, Junhao Ran, Yukun Yan, Zhenghao Liu, Shuo Wang, Xu Han, Zhiyuan Liu, and 1 others. 2024. Visrag: Vision-based retrieval-augmented generation on multi-modality documents. *arXiv preprint arXiv:2410.10594*.

Yi Zhang, Bolin Ni, Xin-Sheng Chen, Heng-Rui Zhang, Yongming Rao, Houwen Peng, Qinglin Lu, Han Hu, Meng-Hao Guo, and Shi-Min Hu. 2025a. Bee: A high-quality corpus and full-stack suite to unlock advanced fully open mllms. *arXiv preprint arXiv:2510.13795*.

Yifan Zhang, Liang Hu, Haofeng Sun, Peiyu Wang, Yichen Wei, Shukang Yin, Jiangbo Pei, Wei Shen, Peng Xia, Yi Peng, and 1 others. 2025b. Skywork-r1v4: Toward agentic multimodal intelligence through interleaved thinking with images and deepresearch. *arXiv preprint arXiv:2512.02395*.

Han Zhao, Haotian Wang, Yiping Peng, Sitong Zhao, Xiaoyu Tian, Shuaiting Chen, Yunjie Ji, and Xiangang Li. 2025. 1.4 million open-source distilled reasoning dataset to empower large language model training. *arXiv preprint arXiv:2503.19633*.

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody H Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E Gonzalez, and 1 others. 2024. Sglang: Efficient execution of structured language model programs. In *NIPS*.

Yuxiang Zheng, Dayuan Fu, Xiangkun Hu, Xiaojie Cai, Lyumanshan Ye, Pengrui Lu, and Pengfei Liu. 2025. Deepresearcher: Scaling deep research via reinforcement learning in real-world environments. In *EMNLP*.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, and 1 others. 2025. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*.

A Details about the Training Stages

A.1 Alignment & Pre-training

Alignment stage. During this initial phase, we keep the parameters of both the vision encoder and the LLM frozen, updating only the projection layer (*i.e.*, the MLP adapter). The training corpus for this stage consists of approximately 3M samples curated from Captioning and OCR datasets, primarily sourced from Laion-5B (Schuhmann et al., 2022) and OpenVid (Nan et al., 2024). To enhance the quality, the raw metadata from these datasets are re-annotated by state-of-the-art vision-language models (*e.g.,* InternVL3 (Zhu et al., 2025)).

Pre-training stage. Following alignment, we perform large-scale pre-training to enhance the model's multimodal understanding and document-level reasoning. We download all PDFs from the CC-MAIN-2021-31-PDF-UNTRUNCATED dataset[1] and filter for documents in Chinese and English. To construct an image-text dataset for training, we develop an automated pipeline comprising layout detection and text extraction. For layout detection, we employ PP-DocLayout (Cui et al., 2025) to extract all images from a page. Subsequently, we utilize a document OCR model (*i.e.*, POINTS-Reader (Liu et al., 2025c)) to extract all text from the page. Finally, the images and text associated with each document are concatenated, with images positioned at the beginning. The resulting pre-training dataset contains approximately 400 billion tokens. In contrast to the alignment stage, both the vision projector layer and the LLM are jointly optimized during this phase.

[1] https://digitalcorpora.org/corpora/file-corpora/cc-main-2021-31-pdf-untruncated/

A.2 Agentic Seeding

As detailed in Section 3.2.1 of the main paper, this stage focuses on instilling the model with complex reasoning and planning capabilities. We curate a diverse training suite comprising AM-DeepSeek-R1-Distilled-1.4M, Toucan-1.5M, Reason-RFT, and VisualWebInstruct. For multimodal datasets, we utilize only the images and questions, and then distill the reasoning chain using `Qwen3-VL-235B-A22B-Thinking`, filtering based on the original answer.

- **AM-DeepSeek-R1-Distilled-1.4M**: This dataset provides approximately 1.4 million high-quality, challenging reasoning problems designed to provoke deep chain-of-thought processing.
- **Toucan-1.5M**: A tool-agent dataset featuring 1.5 million trajectories. These are synthesized from 495 real-world Model Context Protocols (MCP), spanning over 2,000 unique tools to bolster the model's agentic proficiency.
- **Reasoning-RFT**: A multimodal reasoning dataset containing over 10,000 samples. Its construction follows a rigorous three-stage pipeline: automated generation, filtering, and human verification to ensure logical groundedness.
- **VisualWebInstruct**: Consisting of approximately 900,000 question-answer (QA) pairs, this dataset specifically addresses the scarcity of high-quality, diverse training data for complex multimodal reasoning tasks.

A.3 Supervised Fine-tuning

As introduced in Section 3.2.3, the SFT stage aims to align the model's intrinsic agentic capabilities with the precise execution of search-related tools. Following the methodology described in Section 3.2.2, we construct a multimodal VQA dataset comprising 5,339 high-quality trajectories. To facilitate the V-Fold mechanism, these trajectories are transformed into a dual-format representation:

Method	MMSearch	MMSearch+	FVQA	SimpleVQA	LiveVQA	BC-VL
V-Fold RL	70.8	24.3	70.6	68.0	78.5	42.1
Our RL	70.8	25.2	71.2	68.8	77.7	44.4

Table 7: **Ablation study on RL training strategy.**

for each VQA sample, we provide both a text-only trajectory and a rendered visual trajectory where all intermediate tool observations are rendered into images to enable context-efficient visual processing. This data is further augmented with some QA trajectories from the InfoSeek dataset (Xia et al., 2025) and 15k OCR-specific samples to ensure high-fidelity text recognition. Finally, to maintain broad-spectrum instruction-following capabilities and prevent catastrophic forgetting, we incorporate Honey-Data-1M (Wei et al., 2025b) as a general multimodal foundation. For samples in Honey-Data containing existing reasoning traces (*i.e.*, <think> tags), we employ Qwen3-VL-235B-A22B-Thinking to re-annotate the chain-of-thought content, followed by a rigorous filtering process to exclude any samples where the re-annotated answer deviates from the original response provided in the dataset. The training is conducted using our in-house framework, which is analogous to Megatron (Shoeybi et al., 2019).

A.4 Reinforcement Learning

Building upon the SFT stage, we perform reinforcement learning to further refine the model's decision-making and tool-use precision. Our RL framework is implemented using the verl (Sheng et al., 2024), integrated with the SGLang (Zheng et al., 2024) inference engine to accelerate the rollout process. The training is conducted for 60 steps, with a maximum trajectory depth of 10 turns per episode. The training corpus for this phase is composed of our self-constructed reasoning tasks and an adapted version of the LiveVQA dataset. To encourage more robust and grounded reasoning, we reformulate a significant portion of the originally multiple-choice LiveVQA samples into open-ended questions. Crucially, we bypass V-Fold during RL training. Empirical results, as detailed in Table 7, demonstrate that employing V-Fold during the RL phase yields performance comparable to training without compression. Consequently, we opt for the more generalized uncompressed approach for RL training to maintain simplicity and robustness.

B More Implementation Details

B.1 Model Architecture

Our model architecture is initialized with Qwen3-8B-Base (Yang et al., 2025) and Qwen2-VL-ViT (Wang et al., 2024). For the positional encoding, the model applies 1D RoPE (Su et al., 2024) for visual tokens within the LLM backbone and 2D RoPE within the ViT image encoder. Furthermore, the intermediate projector incorporates a pixel-shuffle operation, effectively reducing the visual sequence length by a factor of 4.

B.2 Rendering Parameters

For image rendering, we adopt the parameter configurations from Glyph (Cheng et al., 2025a). The detailed specifications of these rendering parameters are summarized in Table 8.

Parameter	Value
page-size	595×842
dpi	72
margin-x	10
margin-y	10
font-path	Verdana.ttf
font-size	9
line-height	10
font-color	#000000
alignment	LEFT
horizontal-scale	1.0
first-line-indent	0
left-indent	0
right-indent	0
space-before	0
space-after	0
border-width	0
border-padding	0
page-bg-color	#FFFFFF
para-bg-color	#FFFFFF
auto-crop-width	true
auto-crop-last-page	true

Table 8: Parameter configuration for image rendering.

B.3 Computation Resources

We utilize 128×H20 GPUs for both the agentic seeding stage and the SFT stage. The agentic seeding stage runs for approximately 18 hours, while the SFT stage takes around 7 hours. For the subsequent RL stage, we employ 64× H20 GPUs, with a total execution time of around 24 hours. For benchmarks with fewer than 300 samples, we report the average results on 3 trials.

B.4 Prompts

Our system prompt and three tools schema are as follows.

```
You are a specialized multimodal agent.
Your purpose is to solve visual question
answering tasks by thinking step-by-step
and using tools.

# Tools
You may call one or more functions to
assist with the user query.
You are provided with function signatures
within <tools></tools> XML tags:
<tools>
[Detailed Tool Schemas (web_search, visit,
image_search) will be dynamically injected
here. Refer to the specific functional
definitions for parameter constraints.]
</tools>

For each function call, return a json
object with function name and arguments
within <tool_call></tool_call> XML tags:
<tool_call>
{{"name": <function-name>, "arguments": <
args-json-object>}}
</tool_call>

# Instruction
1. In each turn, you should start with <
think> tag. In this tag, you need to
conduct a step-by-step reasoning process
about the image and question and evaluate
whether tool use would be helpful and give
the reason. If received tool results, you
also need to analyze them.
2. If you think some tools are useful, call
 them in <tool_call> tag. **Remember: You
can only make one single tool call per turn
.**
3. If you think no more tools are needed,
you can answer in <answer> tag. You need to
 provide a concise summary of your
reasoning process that leads to the final
answer. Besides, you also need to put a
simple and direct answer in \boxed{{}} for
verification.

The structure of your response should be
like this:
<think> thinking process satisfying the
Instruction 1 </think>
(<tool_call> tool calls satisfying the
Instruction 2 </tool_call> | <answer>
answer satisfying the Instruction 3 </
answer> )

**Search Trigger**: If you are identifying
a specific entity (building, person,
artwork), you **must** use the image_search
 or web_search tool.
**Visit for Detail**: Use the visit tool to
 scrape specific webpage content if the
search snippet is insufficient.
**Repeated Verification**: Before
presenting a Final Answer, you will **cross-
check** and **validate the information**
you've gathered to confirm its accuracy and
 reliability.
**Question Decomposition**: Break the user
query into atomic sub-questions. (e.g., 1.
Identify the building. 2. Find its height.
3. Compare with the user's target).
```

```
{
  "type": "function",
  "function": {
    "name": "web_search",
    "description": "Call this tool to
interact with the web_search API. You will
receive the top 10 text excerpts from
Google's text search engine using text as
the search query.",
    "parameters": {
      "type": "object",
      "properties": {
        "queries": {
          "type": "array",
          "items": {
            "type": "string",
            "description": "The search
query."
          },
          "description": "The list of
search queries."
        }
      },
      "required": ["queries"]
    }
  }
}
```

```
{
  "type": "function",
  "function": {
    "name": "visit",
    "description": "Visit webpage(s) and
return the summary of the content.",
    "parameters": {
      "type": "object",
      "properties": {
        "url": {
          "type": "array",
          "items": {
            "type": "string",
            "description": "The url of the
webpage to visit."
          },
          "description": "The list of
search urls."
        },
```

```
        "goal": {
          "type": "string",
          "description": "The specific
information goal for visiting webpage(s)."
        }
      },
      "required": ["url", "goal"]
    }
  }
}
```

```
{
  "type": "function",
  "function": {
    "name": "image_search",
    "description": "Utilize the search
engine to retrieve relevant information
based on the input image and return results
 related to the goal.",
    "parameters": {
      "type": "object",
      "properties": {
        "image_index": {
          "type": "integer",
          "description": "The index of the
image you want to search for more
information."
        },
        "goal": {
          "type": "string",
          "description": "The specific
information goal."
        }
      },
      "required": ["image_index", "goal"]
    }
  }
}
```

Our QA2VQA prompt is shown below.

```
You are a Multimodal Data Engineer. Your
goal is to convert a QA pair into a VQA
query where the image acts as a necessary
visual context, NOT as a direct hint to the
 answer.

### STRICT RULES:
1. TARGET ENTITY != ANSWER: The entity you
select to mask (the Target Entity) MUST NOT
 be the same as the Answer.
2. VISUAL ANCHORING: Select an entity from
the QUESTION that serves as a visual anchor
 (e.g., a location, a related object, or
the subject's environment).
3. MAINTAIN COMPLEXITY: Keep all the
complex reasoning steps from the original
question.
4. MASKING: Replace the selected Target
Entity with a visual reference token (e.g.,
 "this location", "this botanical garden",
"the habitat shown here").

### OUTPUT FORMAT (JSON ONLY):
{
  "status": "success" | "discard",
  "selected_target_entity": "The entity
chosen from the QUESTION (must not be the
answer)",
  "transformed_vqa_query": "The rewritten
complex question",
  "original_answer": "{{Answer}}",
  "search_query": "Search string for the
selected_target_entity"
}

### INPUT:
Question: [Question]
Answer: [Answer]
```

Our answer filtering prompt is as follows.

```
### Role
You are a rigorous Semantic Evaluation
Expert. Your task is to determine if the "
Model Output" is factually and semantically
 consistent with the "Ground Truth."

### Evaluation Criteria
1. **Semantic Equivalence**: Respond "yes"
if the core meaning, facts, and conclusions
 are the same, even if wording, syntax, or
language differs.
2. **Detail Level**: If the Model Output is
 more detailed than the Ground Truth but
maintains the same core fact, it is
consistent ("yes").
3. **Fact Conflict**: Respond "no" if there
 is a contradiction, a wrong numerical
value, or a different entity mentioned.
4. **Format Neutrality**: Ignore casing,
punctuation, and minor spelling errors.

### Example 1
- **Question**: "What official residence
was directed to be built by a colonial
Governor of Dominica and whose gardens were
 damaged by hurricanes in 1916, 1930, and
1979? Note: The residence was built near
the location shown in the image."
- **Ground Truth**: "Government House,
Dominica"
- **Model Output**: "Government House"
- **Respond**: "yes"

### Example 2
- **Question**: "What NFL season ended with
 the Giants defeating the Chicago Bears
```

```
47:7 in the championship game shown here,
which was coached by a person whose
offensive coordinator was Vince Lombardi,
and the game was played on an icy field due
 to cold weather conditions, and this Vince
 Lombardi later coached a team that won
five total NFL Championships in seven years
?"
- **Ground Truth**: "1956 New York Giants
season"
- **Model Output**: "1956"
- **Respond**: "yes"

### Context
- **Question**: {question}
- **Ground Truth**: {gt_answer}
- **Model Output**: {output_answer}

### Task
Is the Model Output consistent with the
Ground Truth? Respond only with "yes" or "
no".
```

C More Qualitative Results

C.1 Case Study

Figure 5 and Figure 6 compare the model's behavior before and after RL training. We observe that while the SFT model occasionally suffers from weak instruction-following and redundant tool-calls, RL training effectively rectifies these issues. The RL-refined model exhibits significantly better logical rigor and systematic planning, ensuring efficient, non-repetitive tool usage in complex search tasks.

C.2 Analysis of tool usage.

As shown in Figure 7, we analyze the tool interaction patterns of POINTS-Seeker. We can draw the following observations: (i) POINTS-Seeker can flexibly adjust the number of interaction turns according to the difficulty of different benchmarks. For relatively simpler benchmarks such as SimpleVQA, the average number of interaction turns is 2.8. In contrast, for more challenging benchmarks such as BrowseComp-VL, the average number increases substantially to 6.4. (ii) Across all benchmarks, the model consistently exhibits a preference for using the web search tool, whereas the image search and visit tools are employed less frequently. This observation is intuitive: image search can be invoked at most once per sample, and the visit tool can only be triggered after a valid URL is provided by another tool. (iii) Tool usage varies by benchmark characteristics. For instance, in FVQA, the model relies more heavily on image search to identify rare entities, whereas in BrowseComp-VL, a higher proportion of queries are resolved through textual search or direct image interpretation.

C.3 Qualitative Results

Case Trajectory in MMSearch

Question: Who is the man wearing the number 6 jersey?
Image:

Step 1

```
<think>
I need to identify the
basketball player wearing the
number 6 jersey in this image.
The image shows two players in
USA basketball uniforms, with
one wearing number 6.  This
appears to be from a USA team
game or tournament.
To identify this specific player,
I should use the image_search
tool to find out who wears
number 6 for the USA basketball
team.
</think>
<tool_call>
{"name": "image_search",
"arguments": {"image_index": 0,
"goal": "Identify the
basketball player wearing number
6 jersey for the USA team"}}
</tool_call>
<tool_response>
[Title_1: Chansky's Notebook:
Heartfelt Win -
Chapelboro.com](https://chapelboro
.com/sports/chanskys-notebook-heart
felt-win)
Summary_1: The webpage
discusses the performances of
```

Question: What was the exact number of seats held by the organization represented by the blue backdrop with Australian flags at the time when just over 70% of the vote had been counted? A. 24 B. 85 C. 36 D. 70 **Ground Truth:** C

…
You have reached the maximum number of tool calls. Please answer the question directly …
<think>
… Let me search for information about the 2022 election …
</think>
<tool_call>
{"name": "web_search", "arguments": {"queries": ["2022 Australian election 70% counted seats Coalition Labor"]}}
</tool_call>
SFT

…
You have reached the maximum number of tool calls. Please answer the question directly …
<think>
… the image shows the Liberal-National Coalition (Coalition) at what appears to be an election event in Australia
</think>
<answer>
… option C (36 seats) represents a plausible figure ... \boxed{C}
</answer>
RL

Figure 5: **Case Study 1: SFT vs. RL.** Details are simplified for visualization.

Question: Which location of this university in the image experienced the earliest permanent closure following a reduction in operating hours? **Ground Truth:** Saxbys coffee shop

<tool_call>
{"name": "web_search", "arguments": {"queries": ["UPenn New Kensington campus permanently closed 2024"]}}
</tool_call>
…
<tool_call>
{"name": "web_search", "arguments": {"queries": ["UPenn New Kensington campus permanently closed 2024"]}}
</tool_call>
Repeated calls exceeded the context length.
SFT

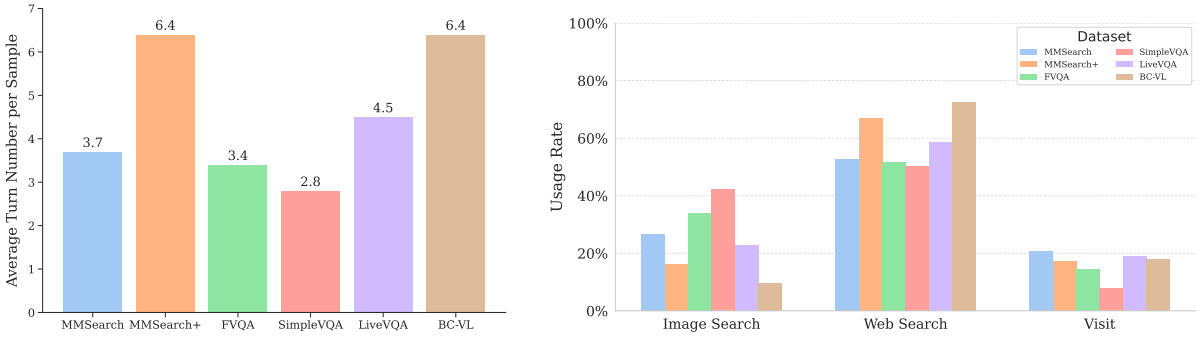
<think>
Based on my search, I can confirm that **Saxbys at 40th and Locust**: Reduced hours (eliminated weekend operations) in April 2025, then closed permanently on May 2, 2025 …
</think>
<answer>
… the Saxbys location at 40th and Locust Streets experienced the earliest permanent closure following a reduction ... \boxed{Saxbys at 40th and Locust Streets}
</answer>
RL

Figure 6: **Case Study 2: SFT vs. RL.** Details are simplified for visualization.

key players on Team USA, including LeBron James, Kevin Durant, Stephen Curry, and Joel Embiid, during a crucial game against Serbia. However, there is no specific mention of the player wearing the number 6 jersey for the USA team, making it impossible to identify based on the provided content.
[Title_2: LeBron James, still leading the way for the US at age 39
...](https://www.wjhl.com/sports/us-world-sports/ap-lebron-james-still-leading-the-way-for-the-us-at-age-39-has-another-chance-at-an-olympic-gold/)
Summary_2: Failed to parse the web page.
[Title_3: U.S. Men's basketball team advances after beating Serbia
...](https://www.youtube.com/watch?v=eisXkbYVt8c)
Summary_3: The webpage content includes multiple references to the U.S. Men's basketball team's performance against Serbia focusing on highlights, commentary ...
...
</tool_response>

Step 2

<think>
Based on the search results, I found that the player wearing number 6 for the USA team is LeBron James. Multiple sources confirm this: 1. The first result mentions "LeBron James, Kevin Durant, and Stephen Curry" as key players for Team USA 2. The fourth result specifically states "LeBron James to be Team USA's male flag bearer for Olympic" 3. The fifth result explicitly states "the



(a) Average number of tool interaction turns.

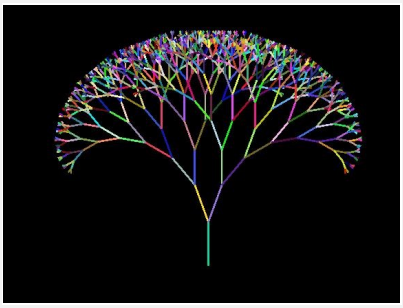
(b) Percentage distribution of tool types across various benchmarks.

Figure 7: **Tool usage analysis for POINTS-Seeker across six diverse evaluation benchmarks.**

basketball player wearing the number 6 jersey for the USA team is LeBron James"
The image shows two players in USA uniforms - one wearing number 6 (LeBron James) and another wearing number 7 (Kevin Durant, as visible on the back of his jersey). This appears to be from a USA team game, likely during the Olympics based on the context from the search results. I have enough information to provide a definitive answer.
</think>
<answer>
Based on the image and the search results, the man wearing the number 6 jersey is **LeBron James**. He is a legendary NBA player who has been a key member of the USA men's basketball team, serving as the team's male flag bearer for the Paris Olympics. The image shows him in a USA team uniform alongside Kevin Durant (wearing number 7), likely during an Olympic basketball game.
\boxed{LeBron James} ✓
</answer>

Case Trajectory in BrowseComp-VL

Question: What is the dimension of this figure, which is constructed by recursively removing the middle third from a line segment and listed by Hausdorff dimension?
Image:



Step 1

<think>
Let me analyze this image. I can see a fractal-like structure that appears to be a tree or branching pattern with many colored lines. The description mentions it's constructed by "recursively removing the middle third from a line segment" and is "listed by Hausdorff dimension." This sounds like it could be related to the Cantor set or a similar fractal construction. When you recursively remove the middle third from a line segment, you're creating a Cantor set. The Hausdorff dimension of the Cantor set is a well-known mathematical concept. However, I should verify what specific fractal this is and what its Hausdorff dimension is. The

image shows a tree-like structure with branching, which might be a different fractal construction. Let me search for more information about this. I need to identify what this specific figure represents and find its Hausdorff dimension.
</think>
<tool_call>
{"name": "image_search", "arguments": {"image_index": 0, "goal": "Identify this fractal figure and find its Hausdorff dimension"}}
</tool_call>
<tool_response>

[Title_1: Colored Fractal Tree - MATLAB Mini Hack](https://nl.mathworks.com/matlabcentral/communitycontests/contests/4/entries/3386)
Summary_1:
The fractal figure displayed is a binary fractal tree, characterized by recursive branching where each branch splits into two sub-branches at fixed angles (±20 degrees). The self-similarity of the structure is evident from the recursive function in the MATLAB code. Although the exact Hausdorff dimension is not provided on the webpage, the fractal's binary branching and recursive nature suggest that its Hausdorff dimension can be calculated using self-similarity principles and the formula involving scaling ratios and branching factors.

[Title_2: Math is actually beautiful & satisfying #shorts #fractals #maths - YouTube](https://www.youtube.com/shorts/qa1N3aE8to)
Summary_2:
The fractal figure being referenced in the content appears to be the Mandelbrot set, which is described as resembling a rose. The mention of "#fractals" and the visual nature of the Mandelbrot set supports this identification. However, the webpage content does not explicitly provide the Hausdorff dimension of the fractal. Further research or analysis would be needed to compute or confirm the Hausdorff dimension of this specific Mandelbrot representation.

[Title_3: Rainbow Tree design using python programming.(With Source Code).. #python - YouTube](https://www.youtube.com/watch?v=cBR3dlxb94k)
Summary_3:
The provided webpage content consists solely of interface elements and status messages from a video streaming platform and contains no information related to fractal geometry or the Hausdorff dimension. Therefore, it is insufficient for identifying a fractal figure or calculating its Hausdorff dimension.

[Title_4: Fun with GDIPlus (GDI+) - AutoHotkey Community](https://www.autohotkey.com/boards/viewtopic.php?t=6071)
Summary_4:
The webpage content provides information about a security verification process implemented by 'www.autohotkey.com' to protect against malicious bots. It confirms that the verification has been completed successfully, and it is waiting for the website to respond. There is no information related to fractal figures or their Hausdorff dimension within the provided content. Therefore, the user's goal cannot be fulfilled using this webpage.

[Title_5: Fractal Trees. | Medium](https://bhavuklabs.medium.com/fractal-trees-in-java-recursion-explained-7bc1b6e6bd57)
Summary_5:
The webpage content provided is related to a bot security verification process and does not mention or describe any fractal figure or mathematical concept such as the Hausdorff dimension. Therefore, it does not contribute to achieving the user's goal of identifying a fractal figure or determining its Hausdorff dimension.

</tool_response>

Step 2

<think>
The image search results indicate this is a "binary fractal tree" with recursive branching. However, the question mentions "removing the middle third from a line segment" which is characteristic of the Cantor set construction, not a binary fractal tree. Let me reconsider. The question specifically mentions: 1. "Recursively removing the middle third from a line segment" - this is the Cantor set construction 2. "Listed by Hausdorff dimension" - this is a mathematical property But the image shows a tree-like structure, not the Cantor set. There might be a mismatch between the question description and the actual figure, OR this could be a different fractal construction. Let me search more specifically for fractals constructed by removing middle thirds and their Hausdorff dimensions. </think>
<tool_call>
{"name": "web_search", "arguments": {"queries": ["fractal removing middle third Hausdorff dimension", "Cantor set Hausdorff dimension", "middle third removal fractal dimension"]}}
</tool_call>
<tool_response>

A Google search for 'fractal removing middle third Hausdorff dimension' found 10 results:

Web Results
1. [List of fractals by Hausdorff dimension - Wikipedia](https://en.wikipedia.org/wiki/List_of_fractals_by_Hausdorff_dimension)

Presented here is a list of fractals, ordered by increasing Hausdorff dimension, to illustrate what it means for a fractal to have a low or a high dimension.

2. [[PDF] the hausdorff dimension: construction and methods of calculation](https://math.uchicago.edu/~may/REU2024/REUPapers/Hao,Wanqing(Anita).pdf)

Step 2: Divide the equilateral triangle into four smaller congruent equilateral triangles. Step 3: Remove the middle triangle (the one formed in the center) ...

3. [Hausdorff dimension of Cantor middle third set - Math Stack Exchange](https://math.stackexchange.com/questions/4104758/hausdorff-dimension-of-cantor-middle-third-set)
Date published: Apr 16, 2021

The Cantor trinary set is defined by starting with the interval [0, 1]. Remove the middle third, leaving [0, 1/3]U [2/3, 1]. Remove the middle ...

4. [[PDF] Exploring Hausdorff Dimension - DigitalCommons@SHU](https://digitalcommons.sacredheart.edu/cgi/viewcontent.cgi?article=1072&context=acadfest)
Date published: Apr 21, 2017

Remove the middle third once again, and continue removing the middle third of each line from Ek to Ek+1.

5. [ELI5 (kinda) Why there aren't fractals with a Hausdorff dimension > 3?](https://www.reddit.com/r/math/comments/4wofz1/eli5_kinda_why_there_arent_fractals_with_a/)
Date published: Aug 8, 2016

From my limited knowledge, an object's Hausdorff dimension is a bit like the number of dimensions a geometric object has, but a little bit more flexible.

6. [Fractals & the Fractal Dimension](https://www.vanderbilt.edu/AnS/psychology/cogsci/chaos/workshop/Fractals.html)

Fractals, which are irregular geometric objects, require a third meaning: The Hausdorff Dimension. If we take an object residing in Euclidean dimension D and ...
7. [How to compute the dimension of a fractal | plus.maths.org](https://plus.maths.org/content/how-compute-dimension-fractal)
Date published: Jul 18, 2024

...

</tool_response>

Step 3

<think>
Perfect! Now I have the information I need. From the search results, I can see that:
1. The Cantor set is constructed by recursively removing the middle third from a line segment
2. The Hausdorff dimension of the Cantor set is log2/log3, which equals approximately 0.6309...
However, I need to verify if the image actually shows the Cantor set or a different fractal. The image search initially identified it as a "binary fractal tree," but the question specifically mentions "removing

the middle third from a line segment."
Let me search more specifically to understand what fractal this might actually be, or if there's a connection between the Cantor set and tree-like structures.
</think>
<tool_call>
{"name": "web_search", "arguments": {"queries": ["Cantor set tree structure Hausdorff dimension", "fractal tree middle third removal dimension", "Cantor dust Hausdorff dimension"]}}
</tool_call>
<tool_response>

A Google search for 'Cantor set tree structure Hausdorff dimension' found 10 results:

Web Results
1. [Hausdorff dimension of Cantor set - Mathematics Stack Exchange](https://math.stackexchange.com/questions/151656/hausdorff-dimension-of-cantor-set)
Date published: May 30, 2012

We want to proof the Hausdorff dimension of C is α:=log2/log3. So we calculate the d-dimensional Hausdorff measure Hd(C) for all d to determine ...

2. [Hausdorff dimension of a subset of Cantor set - MathOverflow](https://mathoverflow.net/questions/88576/hausdorff-dimension-of-a-subset-of-cantor-set)
Date published: Feb 15, 2012

... Hausdorff dimension is topological entropy divided by logλ, where λ is the contraction ratio at each step of the construction of the Cantor set.

3. [Cantor set - Wikipedia](https://en.wikipedia.org/wiki/Cantor_set)

In mathematics, the Cantor set is a self-similar set of points lying on a single line segment that has a number of unintuitive properties.

4. [[PDF] Exploring Hausdorff Dimension - DigitalCommons@SHU](https://digitalcommons.sacredheart.edu/cgi/viewcontent.cgi?article=1072&context=acadfest)
Date published: Apr 21, 2017

The Hausdorff dimension of the Cantor Set is log2 log3. = 0.6309... An intuitive proof would go as follows. Proof. The Cantor Set, call it F ...

5. [Soft bounds on the Hausdorff dimension of dynamical Cantor sets](https://matheuscmss.wordpress.com/2017/02/04/soft-bounds-on-the-hausdorff-dimension-of-dynamical-cantor-sets/)
Date published: Feb 4, 2017

In some applications of dynamical Cantor sets, it is important to dispose of estimates on their Hausdorff dimensions: for instance, the ...

6. [[PDF] Hausdorff Dimension of Cantor Sets - ResearchGate](https://www.researchgate.net/publication/305084244_Hausdorff_Dimension_of_Cantor_Sets)
Date published: Jul 9, 2016
This essay is a trial to determine conditions which characterize a local dimension function. In fact, the problem reduces to the ...

...
</tool_response>

Step 4

<think>
Now I have clear information. The Cantor set (and Cantor dust) has a Hausdorff dimension of log2/log3 ≈ 0.6309.
However, I still need to reconcile this with the image. The image search initially identified it as a "binary fractal tree" with recursive branching. But the question specifically mentions "removing the middle third from a line segment."
Let me think about this more carefully. There are a few possibilities:
1. The question might be misdescribing the figure - it could actually be a Cantor set or Cantor dust
2. The figure could be a tree-like representation of the Cantor set
3. It could be a different fractal construction
Looking at the search results, I found that "Cantor dust" is mentioned with a Hausdorff dimension of 1.2619 (which is 2×log2/log3). This is interesting - it's the Hausdorff dimension of the 2-dimensional Cantor set.
But the most fundamental Cantor set (1-dimensional) has Hausdorff dimension log2/log3 ≈ 0.6309.
Let me search more specifically to understand what this tree-like structure might represent and what its dimension is.
</think>
<tool_call>
{"name": "web_search", "arguments": {"queries": ["binary fractal tree Hausdorff dimension", "fractal tree recursive branching dimension formula", "Hausdorff dimension tree fractal construction"]}}
</tool_call>
<tool_response>

A Google search for 'binary fractal tree Hausdorff dimension' found 10 results:

Web Results
1. [List of fractals by Hausdorff dimension - Wikipedia](https://en.wikipedia.org/wiki/List_of_fractals_by_Hausdorff_dimension)

Presented here is a list of fractals, ordered by increasing Hausdorff dimension, to illustrate what it means for a fractal to have a low or a high dimension.

2. [What is the path cardinality of an infinite binary tree with fractal ...](https://mathoverflow.net/questions/142200/what-is-the-path-cardinality-of-an-infinite-binary-tree-with-fractal-boundary-of)
Date published: Sep 15, 2013

Clearly, FIBT has path cardinality larger than countably infinite, as it has at least the cardinality of a set of positive Hausdorff dimension.

3. [Symmetric fractal trees in three dimensions - ScienceDirect.com](https://www.sciencedirect.com/science/article/abs/pii/S0960077906003894)

Below are definitions of dimension and of space-filling. The Hausdorff dimension d of a fractal tree ...

4. [[PDF] Fractals in Probability and Analysis - Stony Brook

...
</tool_response>

Step 5

<think>
Now I have more information. From the search results, I can see that: 1. For fractal trees, the Hausdorff dimension is calculated using the formula D =

log(N)/log(R), where: N = number of self-similar parts after each iteration. R = scaling factor (how much each branch is scaled down) 2. The image search identified this as a "binary fractal tree" with recursive branching. 3. The question mentions "removing the middle third from a line segment" which is characteristic of the Cantor set construction. There seems to be a disconnect between the question description and the actual figure. Let me think about this more carefully. If this is a binary fractal tree (as identified by the image search), and if it's constructed by recursively removing the middle third, what would be its dimension? Actually, let me reconsider the question. It says: "constructed by recursively removing the middle third from a line segment and listed by Hausdorff dimension" This could mean: - The figure is constructed using the middle third removal method (Cantor set construction) - OR it's a tree-like representation of such a construction
But the image clearly shows a tree structure, not the Cantor set. Let me search for information about whether there's a connection between middle third removal and tree structures, or if this might be a different fractal construction.
</think>
<tool_call>
{"name": "web_search", "arguments": {"queries": ["middle third removal fractal tree construction", "Cantor set tree visualization Hausdorff dimension", "fractal tree construction method middle third"]}}
</tool_call>
<tool_response>

A Google search for 'middle third removal fractal tree construction' found 10 results:

Web Results
1. [Cantor set - Wikipedia](https://en.wikipedia.org/wiki/Cantor_set)

The most common construction is the Cantor ternary set, built by removing the middle third of a line segment and then repeating the process with the remaining ...

2. [Three steps of iterative construction of the middle third Cantor set.](https://www.researchgate.net/figure/Three-steps-of-iterative-construction-of-the-middle-third-Cantor-set_fig4_394720209)

Geometrically, the middle third Cantor set is constructed by repeatedly removing middle third open intervals from the unit interva ...

3. [Fractals & the Fractal Dimension](https://www.vanderbilt.edu/AnS/psychology/cogsci/chaos/workshop/Fractals.html)

...

</tool_response>

Step 6

<think>
Now I have a clearer picture. From the search results, I can see that:
1. The Cantor set is constructed by removing the middle third from a line segment
2. The Cantor set has Hausdorff dimension log2/log3 ≈ 0.6309
3. Fractal trees are constructed using different methods (like dividing into thirds and replacing the middle third with an equilateral triangle, or binary branching)
The key insight is that the question mentions "constructed by recursively removing the middle third from a line segment" - this is the Cantor set construction. However, the image shows a tree-like structure.
There are a few possibilities:
1. The question is misdescribing the figure
2. The figure is a tree-like representation of the Cantor set
3. It's a different fractal construction
But looking at the search results more carefully, I found this important information:
"The Cantor ternary construction is a tree of nested subsets."
This suggests that the Cantor set can be represented as a tree structure.
However, the most fundamental

Cantor set (1-dimensional) has Hausdorff dimension log2/log3 ≈ 0.6309.
Let me think about what the question is really asking. It says: "What is the dimension of this figure, which is constructed by recursively removing the middle third from a line segment and listed by Hausdorff dimension?"
If this figure is indeed constructed by removing the middle third from a line segment (Cantor set construction), then the Hausdorff dimension is log2/log3.
But if it's a fractal tree constructed using a similar method (removing middle thirds and creating branches), then the dimension would be calculated differently.
Given that the image search identified it as a "binary fractal tree" and the question mentions "removing the middle third," I need to determine which interpretation is correct.
Actually, let me reconsider.
The most straightforward answer to the question is that if a fractal is constructed by recursively removing the middle third from a line segment, it's the Cantor set, and its Hausdorff dimension is log2/log3.
But the image doesn't look like the Cantor set - it looks like a tree. However, the question might be using "removing the middle third" metaphorically or the figure might be a different construction.
Let me search for the exact Hausdorff dimension of the Cantor set to be certain.
</think>
<tool_call>

{"name": "web_search", "arguments": {"queries": ["Cantor set Hausdorff dimension exact value log2/log3", "Hausdorff dimension of Cantor ternary set formula"]}}
</tool_call>
<tool_response>

A Google search for 'Cantor set Hausdorff dimension exact value log2/log3' found 10 results:

Web Results
1. [Set with Hausdorff dimension $s = \log 2 / \log 3$ but $H^s = \infty](https://math.stackexchange.com/questions/3686906/set-with-hausdorff-dimension-s-log-2-log-3-but-hs-infty)
Date published: May 22, 2020

dimH(X)=s where s=log2log3, but Hs(X)=∞. Here I am using the notations from Fractal Geometry by Kenneth Falconer. From exercise 4.9 of the book.

2. [[PDF] Exploring Hausdorff Dimension - DigitalCommons@SHU](https://digitalcommons.sacredheart.edu/cgi/viewcontent.cgi?article=1072&context=acadfest)
Date published: Apr 21, 2017

This is a clear sign we need to use Hausdorff dimension! Theorem 4.1. The Hausdorff dimension of the Cantor Set is log2 log3. = 0.6309...

3. [List of fractals by Hausdorff dimension - Wikipedia](https://en.wikipedia.org/wiki/List_of_fractals_by_Hausdorff_dimension)

Presented here is a list of fractals, ordered by increasing Hausdorff dimension, to illustrate what it means for a fractal to have a low or a high dimension.

4. [[PDF] Chapter 4 Techniques for calculating dimensions](https://www.ma.imperial.ac.uk/~jswlamb/M345PA46/F03%20chap%204.pdf)

As a general rule, we get upper bounds for Hausdorff measures and dimensions by finding effective coverings by small sets, and lower bounds by putting measures.

5. [[PDF] Hausdorff measure of cartesian product of Cantor sets - arXiv](https://arxiv.org/pdf/2510.09701)
Date published: Oct 9, 2025

For the case of the ternary Cantor set, its Hausdorff dimension is known to be log3 2, and it was shown by Felix Hausdorff himself in 1919 [4] ...

6. [[PDF] Introduction to Fractal Geometry and Cantor Sets](https://www.aqouyang.com/article_view?id=WQKCJYQIK24)
Date published: Feb 18, 2024

The Hausdorff dimension HH of a set A is defined as the smallest value α for which the above outer measure exists with a finite nonzero value.

7. [[PDF] Hausdorff Dimension, February 2012](https://sites.math.rutgers.edu/~bumby/nyntsem/NYFeb2012.pdf)
Date published: Feb 16, 2012

Note that the measure of each interval in C is its length to the power log2=log3. The description we seek will has this as the dimension of the ...

8. [Hausdorff Dimension - an overview | ScienceDirect Topics](https://www.sciencedirect.com/topics/engineering/hausdorff-dimension)

is the Hausdorff dimension of C. For a cube D* = 3, for a square D* = 2, for a line segment D* = 1, for the tertiary Cantor set D* = log2/log3 (see [16]).

9. [[PDF] HAUSDORFF DIMENSION AND LOCAL STRUCTURE OF FRACTAL](https://conservancy.umn.edu/bitstreams/88994c90-fbae-416c-bab7-bd48033ebd6f/download)
Date published: Aug 13, 2024

In [Page82], we can use Mass distribution method to find Hausdorff dimension of F × F □ R2 is log 4/log 3, where F is the middle third Cantor ...

10. [[PDF] Several Methods for Calculating and Estimating the Hausdorff ...](https://pdfs.semanticscholar.org/0c6c/d354da56b9a7b7681631458dfc7739d54fed.pdf)

Specifically, the Hausdorff dimension of the Cantor set is denoted bylog2 log3. If we attempt to cover the Cantor set with sets of dimensions 0, i.e., with.
======
A Google search for 'Hausdorff dimension of Cantor ternary set formula' found 9 results:

Web Results
1. [Hausdorff dimension of Cantor set - Math Stack

...
</tool_response>

Step 7

<think>
Perfect! Now I have confirmed the exact Hausdorff dimension of the Cantor set. Multiple sources confirm that:
The Hausdorff dimension of the Cantor set is log2/log3 ≈ 0.6309
This is the standard result for the Cantor ternary set, which is constructed by recursively removing the middle third from a line segment.
However, I still need to reconcile this with the image. The image search identified it as a "binary fractal tree" with recursive branching. But the question specifically mentions "removing the middle third from a line segment."
There are a few possibilities:

1. The question is misdescribing the figure - it's actually the Cantor set
2. The figure is a tree-like representation of the Cantor set construction
3. It's a different fractal construction that uses a similar method

Given that the question explicitly states "constructed by recursively removing the middle third from a line segment," and this is the defining characteristic of the Cantor set, the answer should be the Hausdorff dimension of the Cantor set, which is log2/log3.

However, I should verify if there's any connection between the Cantor set and tree structures, or if this might be a different fractal altogether.

Let me think about this more carefully. The Cantor set is a 1-dimensional fractal. If the question is asking about a tree-like structure constructed using the middle third removal method, it could be:

- A 2-dimensional Cantor set (Cantor dust) which has Hausdorff dimension 2×log2/log3 ≈ 1.2619
- Or a different construction

But the most straightforward interpretation is that the question is referring to the Cantor set construction, and the Hausdorff dimension is log2/log3.

Let me do one more search to see if there's any specific connection between this tree structure and the Cantor set.

</think>

<tool_call>

{"name": "web_search", "arguments": {"queries": ["Cantor set tree structure visualization", "ternary Cantor set tree representation", "Cantor set as tree fractal"]}}

</tool_call>

<tool_response>

A Google search for 'Cantor set tree structure visualization' found 10 results:

Web Results

1. [Cantor Set - Wolfram Demonstrations Project](https://demonstrations.wolfram.com/CantorSet/)
The Cantor set is obtained from[0,1] by removing the open middle third and iteratively removing the middle third of each remaining interval.
2. [[PDF] Cantor sets, binary trees, and Lipschitz ...]
...

</tool_response>

Step 8

<think>

Excellent! Now I found the key information.

From the search results, I can see:

1. From result #5 in the first search: "Often trees are represented by joining any element to each of its immediate successors. Figure 1 shows part of the Cantor tree, where the horizontal segments."
2. From result #5 in the third search: "The Cantor Set is one of the simplest fractals, and also one of the earliest to have been studied in detail. As a one-dimensional structure, it was easy to..."
3. From result #1 in the third search: "One of the first examples of a fractal is the Cantor Set, which was discovered by Georg Cantor in 1883."

The key insight is that the Cantor set can be represented as a tree structure. The Cantor

```
tree is a tree where each node
has two children, representing
the binary choices in the
ternary (base-3) representation
of numbers in the Cantor set.
So the image appears to be
showing the Cantor tree - a tree
structure representing the
Cantor set.  The "removing the
middle third" construction is
the underlying process that
generates this tree structure.
Therefore, the Hausdorff
dimension of this figure (the
Cantor tree, which represents
the Cantor set) is the Hausdorff
dimension of the Cantor set,
which is:
**log2/log3 ≈ 0.6309**
This is the standard result for
the Cantor ternary set.
</think>
<answer>
Based on my research, I can now
provide a comprehensive answer
to this question.
The figure shown is actually the
**Cantor tree**, which
represents the Cantor set
construction.  The Cantor set is
constructed by recursively
removing the middle third from a
line segment - starting with
[0,1], then removing the middle
third (1/3, 2/3), then removing
the middle third of each
remaining interval, and so on.
The Cantor set can be
represented as a tree structure
where:
- The root represents the
interval [0,1]
- Each node branches into two
children representing the left
and right thirds of the interval
- This creates a binary tree
structure where each level
represents one iteration of the
construction
The Hausdorff dimension of the
Cantor set (and therefore of the
Cantor tree representing it) is
calculated using the formula for
self-similar fractals:
**D = log(N)/log(R)**
Where:
- N = number of self-similar
parts after each iteration = 2
(each interval splits into 2
parts)
- R = scaling factor = 1/3 (each
branch is 1/3 the length of the
parent)
Therefore:
**D = log2/log3 ≈ 0.6309**
This is the well-known Hausdorff
dimension of the Cantor ternary
set.
\boxed{log_2 / log_3} ✓
</answer>
```

D Ethical Considerations

D.1 Potential Risks

POINTS-Seeker-8B may inherit risks from both web retrieval and generative modeling. Retrieved content can be noisy, outdated, biased, or maliciously crafted, potentially causing incorrect answers or prompt-injection vulnerabilities. The agent may also encounter sensitive or copyrighted information, raising privacy and misuse concerns, and may still hallucinate when the evidence is insufficient. Deployment should therefore incorporate source verification, privacy filtering, prompt-injection defense, tool-use sandboxing, and human oversight in high-stakes scenarios.

D.2 License for Artifacts

We will release the code, model checkpoints, training/evaluation scripts, and curated data under appropriate open-source licenses. The code will be released under the Apache-2.0 license. The released model checkpoints will follow the license of the corresponding backbone model and will be provided for research use. The curated datasets and trajectories will be released under a research-friendly license, subject to the licenses and terms of the original data sources. For any third-party resources used in this work, including backbone models, benchmarks, and external tools, we comply with their respective licenses and usage terms.

D.3 Data Safety and Anonymization

During data curation, we did not intentionally collect private or sensitive personal information. For data derived from web sources, we avoid redistributing raw content when privacy or licensing concerns are unclear, and release only permitted artifacts or derived metadata when applicable.

D.4 Artifact Documentation

The data mainly covers knowledge-intensive multimodal search scenarios such as entities, landmarks, products, web pages, charts, documents, and multi-hop VQA. The language coverage is primarily English, with possible multilingual content inherited from public web sources and benchmarks. The artifacts are intended for research on multimodal search agents and long-horizon reasoning. We do not intentionally collect or annotate demographic attributes.

D.5 Use of AI Assistants.

We used AI assistants to help with language polishing and improving the clarity of the manuscript. The authors are fully responsible for all technical content, experimental results, analyses, and claims in the paper.