DATARESEARCHBENCH: BENCHMARKING VISUAL DATA SOURCING WITH GENERALIST AGENTS ON THE OPEN WEB

Anonymous authors
Paper under double-blind review

ABSTRACT

Current agents are increasingly capable of retrieving information and synthesizing answers, but many research workflows require constructing data resources that are not already available in a suitable form. We introduce DataResearchBench to evaluate agents' capability of transforming high-level requirements into structured visual data collections through source discovery, acquisition, verification, and annotation. The benchmark contains 46 tasks across seven domains and requires agents to acquire images from public web resources without a predefined candidate pool. It also studies reference images as visual specifications for requirements that are difficult to express through text alone. We evaluate collected data using duplicate and provenance checks together with human-reviewed task-specific rubrics that separate base validity from additional quality. Experiments with five generalist agents (e.g., Codex, Claude Code) reveal a substantial gap between finding valid examples and constructing consistently valid collections. The strongest agent reaches 78.3% BPR (Base Pass Rate) under visual specification, yet only 45.7% of tasks contain five valid images. Visual specifications improve most agents, while fixed-backbone comparisons reveal substantial framework-dependent variation in collection quality and sourcing behavior. Larger collection requests yield more valid images overall but lower validity and quality at the largest scale, and multi-task execution further exposes differences in agent orchestration. DataResearchBench provides a testbed for studying how specification, framework, and workload shape reliable visual data construction. We will open-source the dataset and code and continuously update DataResearchBench to support emerging research needs across a broader range of data types.

1 INTRODUCTION

Advances in multimodal large language models (MLLMs) have enabled agents to browse websites, retrieve information from the open web, and synthesize external evidence into answers and reports (Yang et al., 2023; Mialon et al., 2024; Zhou et al., 2024; Koh et al., 2024; Wei et al., 2025; Du et al., 2026). Yet many research workflows require constructing data resources that do not already exist in a suitable form. We refer to the capability of discovering, acquiring, verifying, and organizing such data from open-world sources as *data research*. Unlike retrieving evidence for a single answer, data research requires the acquired items themselves to satisfy downstream requirements for relevance, quality, provenance, and annotation.

Data research is challenging because the target specification and collection process are open-ended. Natural-language requirements may underspecify visual properties such as viewpoint, composition, or style, while reference images can complement text by illustrating desired content (Vo et al., 2019; Liu et al., 2021). We treat such references as *visual specifications* that agents must translate into search, verification, and selection decisions. Success is inherently collection-level: finding a few suitable examples is easier than constructing a complete collection, especially as collection size or the number of sourcing objectives increases. This motivates a central question: *How reliably can generalist agents construct visual data collections from open-world sources, and how does this reliability change with visual specifications, agent frameworks, and collection workloads?*

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

To study this problem, we introduce **DataResearchBench**, a benchmark for agentic visual data research from open web resources. It contains 46 tasks across seven domains, derived from data requirements in multimodal benchmark and dataset studies. Agents must acquire images without a predefined candidate pool or search strategy while preserving source information and providing task-specific annotations. We evaluate Text-only and Visual Specification settings, differing only in whether three human-selected reference images accompany the task description. Because open-web data sourcing admits no unique target collection, submissions are evaluated using duplicate and provenance checks together with human-reviewed, task-specific rubrics that separate minimum validity from additional quality.

Our experiments show that reliable data sourcing depends on more than retrieving additional images. Visual specifications improve most agents but affect task-level completion unevenly. Fixed-backbone comparisons reveal substantial framework-dependent differences in collection quality and sourcing behavior. Larger collection requests produce more valid images in absolute terms but lower validity and quality at the largest scale, while joint multi-task execution reveals distinct orchestration patterns across agents. Overall, DataResearchBench shows how requirement specification, agent framework, and workload shape the quality of constructed visual data collections.

2 Related Work

2.1 Agentic dataset construction.

DataComp evaluates multimodal dataset curation through downstream model performance under a standardized training protocol (Gadre et al., 2023). Recent work has begun to study more agentic forms of data construction. AutoData introduces a multi-agent system and the Instruct2DS benchmark for collecting structured records from live web sources (Ma et al., 2025), while DatasetResearch evaluates agents that discover existing datasets or synthesize data for downstream requirements (Li et al., 2025). DatasetAgent instead develops a specialized multi-agent system for collecting, processing, cleaning, and annotating real-world images for vision datasets (Sun et al., 2025). DataResearchBench instead evaluates generalist agents that acquire visual instances directly from the open web without predefined candidate pools or prescribed sources. We assess the validity and quality of the resulting collections, together with the agents' sourcing behavior.

2.2 Task-oriented web and research agents.

Web-agent benchmarks study browser-based task completion, including long-horizon interaction in WebArena (Zhou et al., 2024), visually grounded tasks in VisualWebArena (Koh et al., 2024), real-world web navigation in WebVoyager (He et al., 2024), persistent search in BrowseComp (Wei et al., 2025), and long-horizon search with synthesis and attribution in Mind2Web 2 (Gou et al., 2025). Deep Research benchmarks further emphasize iterative search and evidence synthesis, evaluating citation-supported reports (Du et al., 2026), reasoning-intensive research search (Java et al., 2026), and multimodal report generation with visual evidence (Huang et al., 2026). RMIR represents a related multimodal retrieval setting, identifying target images from a predefined candidate pool using image–text queries (Li et al., 2026). DataResearchBench shifts the evaluation target from completing a web task, producing a research answer, or selecting from an existing retrieval space to carrying out sustained open-world data acquisition across an entire collection.

3 DataResearchBench

Figure 1 summarizes DataResearchBench, from task construction and open-web data sourcing to evaluation. Tasks are derived from research-driven visual data requirements, refined through agent filtering and human curation, and executed by generalist agents under Text-only or Visual Specification inputs. The resulting collections are evaluated for sample quality and task completion.

3.1 Data Research Task

For a task τ, the *Text-only* input consists of a natural-language requirement q_τ and requested collection size N_τ. The *Visual Specification* setting additionally provides three human-selected reference

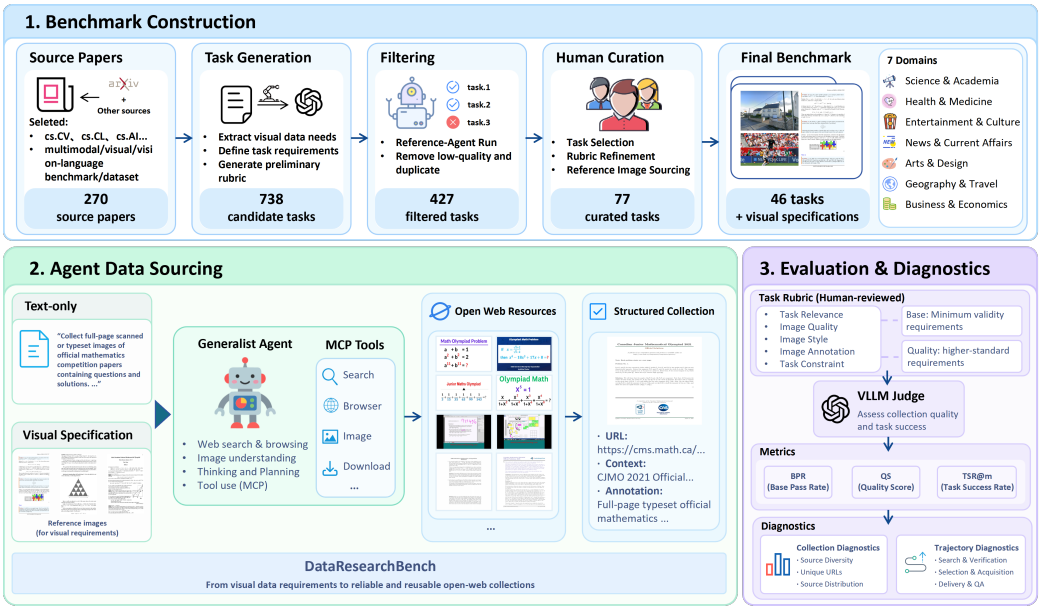


Figure 1: Overview of DataResearchBench. The benchmark construction pipeline produces 46 visual data sourcing tasks across seven domains. Generalist agents acquire structured image collections from the open web under Text-only or Visual Specification inputs. Human-reviewed task-specific rubrics evaluate collection validity and quality, complemented by task-level metrics and diagnostics of sources and agent trajectories.

images, $\mathcal{V}_\tau = \{v_{\tau,j}\}_{j=1}^{3}$, illustrating intended visual characteristics and acceptable variation. The agent must discover, acquire, verify, and organize N_τ real images from public web resources without a predefined candidate pool, target website, or search query. Images must be acquired rather than generated or synthesized, and reference images cannot be submitted. A complete submission is

$$\mathcal{D}_\tau = \{(x_{\tau,i}, u_{\tau,i}, c_{\tau,i}, a_{\tau,i})\}_{i=1}^{N_\tau}, \tag{1}$$

where $x_{\tau,i}$ is the acquired image, $u_{\tau,i}$ its image or source-page URL, $c_{\tau,i}$ the source context, and $a_{\tau,i}$ a factual task-specific annotation. Source context may include the original title, caption, or surrounding text and is stored separately from the agent-generated annotation.

3.2 Benchmark Construction

Source-paper collection. We retrieve relevant papers from arXiv using queries related to multimodal, vision–language, and visual benchmark and dataset research, covering `cs.CV`, `cs.CL`, and `cs.AI`. After canonicalizing arXiv identifiers and removing duplicate versions, this process yields 235 papers. We add 35 manually selected representative papers to broaden application coverage, resulting in a source pool of 270 papers.

Task generation. GPT-5-mini(OpenAI, 2025) analyzes each paper to identify its visual data types, data sources, dataset construction procedures, annotation requirements, and quality-control considerations. It then formulates standalone data-sourcing tasks that can be executed using public web resources. These tasks reflect the underlying data requirements of the source studies rather than require reproduction of their original datasets. This process produces 738 candidate tasks.

Preliminary rubric construction. For each candidate task, GPT-5.5(OpenAI, 2026b) identifies likely failure cases across five dimensions: *Task Relevance*, *Image Quality*, *Image Style*, *Image Annotation*, and *Task Constraint Compliance*, covering content match, visual usability and presentation, annotation correctness, and task-specific restrictions. It then drafts task-specific *base* criteria for minimum sample validity and optional *quality* criteria for stronger samples. Criteria are included only when applicable, and the resulting preliminary rubrics are used for reference-agent filtering before being revised during human curation.

Reference-agent filtering. All 738 candidates are executed by an OpenClaw agent powered by GPT-5.5 without reference images. For each returned image, the judge considers the five preliminary base criteria jointly and assigns a score in $\{1, 3, 5\}$ according to criterion satisfaction, with 5 being the maximum rating. We average these image scores within each task and remove candidates whose mean exceeds 4.0. This step reduces the candidate set to 427 tasks.

Human curation. Reviewers retain 77 tasks based on fidelity to the source studies, meaningful data needs, public-web feasibility, clear and verifiable requirements, and non-redundancy. For each task, we refine the description and rubric and select three representative reference images from the source dataset when available, or otherwise from public web resources. Three reviewers cross-check each rubric and revise it until agreement. From this pool, we select 46 core tasks across seven domains based on task quality, challenge, coverage, non-redundancy, and evaluation feasibility, yielding 138 human-selected reference images. The full task catalog and domain distribution are provided in Appendix B.2.

4 Evaluation Protocol

4.1 Evaluation Overview

We evaluate each submitted collection through four stages: submission validation, duplicate checking, provenance auditing, and task-specific rubric assessment. The first three stages determine whether a submitted sample is eligible for evaluation, while the rubric assessment produces criterion-level base and quality judgments. We additionally analyze the recorded execution trajectories to characterize how agents search, verify, select, acquire, and quality-check candidate images.

4.2 Collection Integrity and Provenance Checks

Duplicate checking. We check each task-level collection for duplicates among submitted images and for overlap with the benchmark reference images. Exact matches are detected using file and decoded-pixel hashes, while perceptually similar pairs are identified using 64-bit perceptual hashes with a Hamming-distance threshold of 6. Candidate pairs are visually reviewed, and confirmed duplicates are marked invalid. All comparisons are restricted to images within the same task.

Provenance audit. We audit whether each submitted sample is supported by its recorded source information through three checks: whether the submitted image matches an image available from the reported source, whether the recorded source context is faithful to independently retrieved source content, and whether the submitted image is consistent with that context. Each check returns *pass*, *fail* or *error*; an explicit *fail* or *error* in any check invalidates the sample.

4.3 Rubric-based Semantic Evaluation

For each retained instance, GPT-5.5 receives the task description, the finalized rubric, the acquired image $x_{\tau,i}$, its recorded source URL $u_{\tau,i}$, source context $c_{\tau,i}$, and annotation $a_{\tau,i}$. The evaluator returns a binary decision for each criterion. The five base criteria are assessed jointly in one call, and the applicable quality criteria are assessed jointly in a separate call. Final scores are computed from these criterion-level decisions rather than from a free-form overall judgment.

4.4 Evaluation Metrics

We evaluate both image-level quality and task-level completion. Base Pass Rate and Quality Score summarize the validity and quality of acquired instances, while Task Success Rate at threshold m measures how consistently agents construct collections containing multiple valid images.

Base Pass Rate (BPR). Let $v_{\tau,i} \in \{0, 1\}$ indicate whether requested slot i passes submission validation and is not invalidated by the duplicate or provenance checks. Let $\mathcal{K}_\tau^{\text{base}}$ index the applicable base criteria for task τ, and let $b_{\tau,i,k} \in \{0, 1\}$ denote the corresponding semantic judgment. The final image-level base validity is

$$B_{\tau,i} = v_{\tau,i} \prod_{k \in \mathcal{K}_\tau^{\text{base}}} b_{\tau,i,k}. \tag{2}$$

For an evaluated task set $\mathcal{T}$, BPR is the proportion of all requested images that are base-valid:

$$\text{BPR} = \frac{\sum_{\tau \in \mathcal{T}} \sum_{i=1}^{N_\tau} B_{\tau,i}}{\sum_{\tau \in \mathcal{T}} N_\tau}. \tag{3}$$

Quality Score (QS). Let $\mathcal{K}_\tau^{\text{quality}}$ index the applicable quality criteria and let $h_{\tau,i,k} \in \{0,1\}$ denote their judgments. Quality credit is assigned only to base-valid images:

$$Q_{\tau,i} = \begin{cases} \dfrac{1}{|\mathcal{K}_\tau^{\text{quality}}|} \displaystyle\sum_{k \in \mathcal{K}_\tau^{\text{quality}}} h_{\tau,i,k}, & B_{\tau,i} = 1 \text{ and } |\mathcal{K}_\tau^{\text{quality}}| > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

QS averages these scores over all requested images:

$$\text{QS} = \frac{\sum_{\tau \in \mathcal{T}} \sum_{i=1}^{N_\tau} Q_{\tau,i}}{\sum_{\tau \in \mathcal{T}} N_\tau}. \tag{5}$$

Task Success Rate (TSR@m). A task meets threshold m when at least m of its requested images are base-valid:

$$\text{TSR@}m = \frac{1}{|\mathcal{T}|} \sum_{\tau \in \mathcal{T}} \mathbb{I}\left[\sum_{i=1}^{N_\tau} B_{\tau,i} \geq m\right]. \tag{6}$$

Unlike BPR and QS, which aggregate image-level outcomes, TSR@m assigns each task a single success decision at the specified threshold.

4.5 Trajectory Analysis

We analyze recorded execution traces to characterize agent sourcing behavior. Qwen3.5-9B(Qwen Team, 2026) categorizes each event as Planning, Reference Analysis, Candidate Search, Source Verification, Candidate Selection, Image Acquisition, Annotation and Delivery, Quality Assurance, Environment and Workflow, or Other. We report event counts and proportions. For grouped analysis, we retain Planning and combine related activities into Search & Verification, Selection & Acquisition, and Delivery & QA. For multi-task runs, events are additionally labeled as task-specific, shared across multiple tasks, or unassigned.

4.6 Human Validation of the Judge

Since criterion-level judgments are produced by an LLM judge, we validate its reliability against human judgments following prior LLM- and MLLM-as-a-Judge studies (Zheng et al., 2023; Chen et al., 2024). Three human reviewers independently assess a random sample covering approximately 5% of the main benchmark outputs using the finalized task-specific base criteria, without access to the automated decisions. The mean agreement with the judge on image-level base validity is 85.3%. Full sampling, annotation, and per-reviewer agreement details are provided in Appendix A.4.

5 Experiments

5.1 Experimental Setup

We evaluate five generalist agents: Claude Code (Anthropic, 2026a) with Claude Opus 5 (Anthropic, 2026b), Codex (OpenAI, 2026a) with GPT-5.6-Sol (OpenAI, 2026c), Grok Build (SpaceXAI,

Table 1: Main results on DataResearchBench under the Text-only and Visual Spec. settings. Domain columns report domain-level BPR, and the final two columns report overall BPR and QS. All values are percentages. The best result in each column within each setting is shown in bold, including ties. Domain abbreviations are defined in the Appendix B.2.

Agents	Art	Bus.	Ent.	Geo.	Health	News	Sci.	BPR	QS
Text-only									
Claude Code	**55.0**	**90.0**	**80.0**	80.0	**77.5**	**68.0**	**67.1**	**73.0**	**67.8**
Grok	50.0	50.0	62.9	**90.0**	**77.5**	56.0	57.1	63.0	58.0
Codex	35.0	40.0	57.1	60.0	72.5	44.0	**67.1**	58.3	50.8
Kimi Code	50.0	20.0	51.4	45.0	67.5	64.0	45.7	50.4	45.7
MiniMax	25.0	10.0	31.4	50.0	35.0	64.0	27.1	33.5	28.3
Visual Spec.									
Claude Code	**75.0**	**85.0**	**85.7**	**80.0**	**85.0**	**64.0**	**74.3**	**78.3**	**71.5**
Grok	60.0	60.0	60.0	75.0	82.5	52.0	72.9	68.3	62.6
Codex	40.0	75.0	77.1	70.0	50.0	40.0	55.7	57.8	50.7
Kimi Code	60.0	30.0	68.6	70.0	65.0	56.0	42.9	54.8	48.6
MiniMax	20.0	40.0	22.9	50.0	62.5	36.0	45.7	41.7	36.5

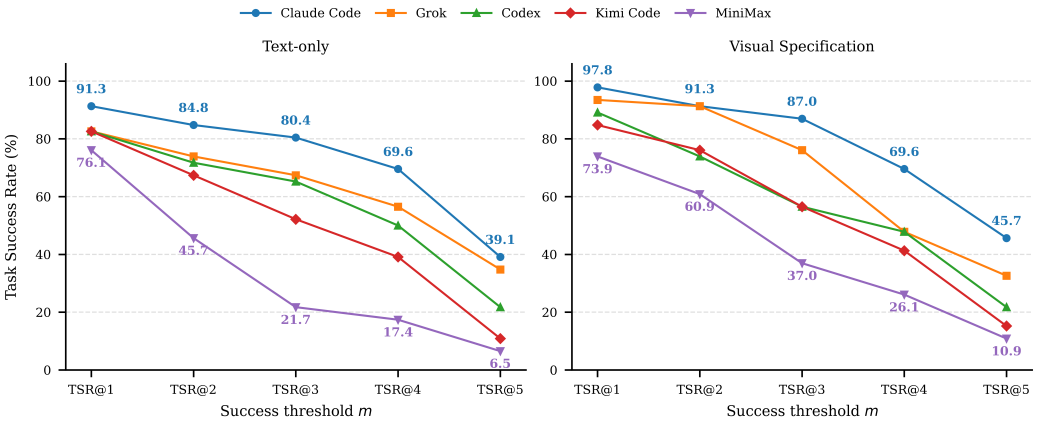


Figure 2: Task success across increasing collection thresholds. TSR@m measures the percentage of tasks containing at least m base-valid images among the five requested images.

2026b) with Grok-4.6 (SpaceXAI, 2026a), Kimi Code (Kimi, 2026b) with Kimi K3 (Kimi, 2026a), and MiniMax Code (MiniMax, 2026b) with MiniMax M3 (MiniMax, 2026a). We additionally evaluate Pi (Earendil Works, 2026) and OpenClaw (OpenClaw Contributors, 2026) in controlled framework comparisons. All agents use high reasoning effort, identical task-independent instructions, and fresh isolated environments. The main evaluation covers all 46 tasks with five requested images per task; the *Text-only* and *Visual Specification* settings differ only in whether the three reference images are provided.

All agents use the same MCP-based tool suite (Model Context Protocol Contributors, 2025) for image search, browsing, downloads, arXiv retrieval, and local image processing. Native model browsing and search are disabled, while agents choose their own queries, sources, and candidates. We impose no task-level limits on time, tokens, tool calls, or model expenditure.

5.2 Main Benchmark Results

Table 1 shows clear performance differences across agents and domains. Claude Code achieves the strongest aggregate BPR and QS in both settings, followed by Grok, Codex, Kimi Code, and MiniMax. Under Visual Specification, Claude Code also obtains the highest BPR in all seven domains, whereas the Text-only setting exhibits more domain-specific variation, with different agents leading or tying in Geography & Travel, Health & Medicine, and Science & Academia.

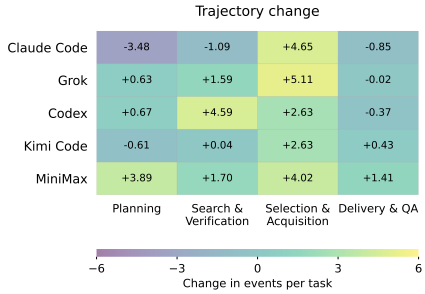


Figure 3: Changes in agent trajectories under Visual Specification relative to Text-only on the same 46 tasks.



Figure 4: Search and verification activity across agent frameworks with fixed backbone models.

A larger gap appears in base validity than in the quality of samples that already pass. Under Visual Specification, BPR ranges from 41.7% to 78.3%, while QS/BPR ranges only from 87.3% to 91.7%. Figure 2 further shows that this gap becomes more pronounced at the collection level: Claude Code finds at least one base-valid image on 45 of 46 tasks, but obtains five valid images on only 21. Across the other agents, TSR@5 ranges from 10.9% to 32.6%. Thus, the main difficulty lies not only in finding valid examples, but in maintaining validity across an entire collection.

Visual specifications improve aggregate BPR and QS for four of the five agents, but their effect on task completion varies by system and threshold. For example, Grok improves in BPR while slightly decreasing at TSR@5, and MiniMax improves at higher thresholds despite a lower TSR@1. Claude Code improves at most thresholds, while Kimi Code improves across all five. Higher average validity therefore does not always translate into higher task success at every collection threshold. Model API cost comparisons are reported in Appendix C.5.

5.3 Effect of Visual Specifications

The effect of visual specifications is strongly task dependent. Aggregate scores can therefore obscure substantial changes in task-level outcomes. Codex provides a clear example: its overall BPR changes only from 58.3% under Text-only to 57.8% under Visual Specification, while TSR@1 increases from 82.6% to 89.1% and TSR@3 decreases from 65.2% to 56.5%, with TSR@5 unchanged. Visual guidance can therefore alter how valid samples are distributed across tasks even when aggregate validity remains nearly unchanged.

Trajectory changes show a more consistent behavioral effect. As shown in Figure 3, *Selection & Acquisition* increases for all five agents, with an average increase of 3.81 events per task. In contrast, changes in planning and *Delivery & QA* are small on average and vary in direction across agents. *Search & Verification* increases for most agents, but less uniformly. The most consistent effect of visual guidance is therefore greater activity around evaluating and acquiring candidate images rather than a uniform increase across the entire workflow.

The same visual guidance is incorporated differently across agents. Claude Code increases selection and acquisition while reducing both planning and search and verification. Codex instead shows its largest increase in *Search & Verification*, together with additional acquisition, despite almost no change in aggregate BPR. Kimi Code changes little outside selection and acquisition, whereas MiniMax increases activity across all four behavior groups. These differences show that visual specifications do not induce a single sourcing strategy. Their effect depends on how the agent integrates visual information into candidate discovery, verification, and acquisition.

5.4 Effect of Agent Framework

We evaluate Kimi K3 with Kimi Code, Pi, and OpenClaw, and GPT-5.6-Sol with Codex, Pi, and OpenClaw. OpenClaw achieves the highest scores on all five reported metrics for both backbones under both input settings (Table 2). Under Text-only with Kimi K3, its BPR reaches 60.9%, com-

Table 2: Framework comparison with fixed backbone models. Each cell reports Text-only / Visual Spec. performance. BPR, QS, and TSR are percentages.

Backbone	Framework	BPR	QS	TSR@3	TSR@4	TSR@5
Kimi K3	Kimi Code	50.4 / 54.8	45.7 / 48.6	52.2 / 56.5	39.1 / 41.3	10.9 / 15.2
	Pi	45.7 / 48.7	42.3 / 45.1	47.8 / 50.0	26.1 / 32.6	13.0 / 15.2
	OpenClaw	**60.9 / 58.7**	**55.0 / 52.1**	**69.6 / 63.0**	**54.3 / 50.0**	**17.4 / 23.9**
GPT-5.6-Sol	Codex	58.3 / 57.8	50.8 / 50.7	65.2 / 56.5	50.0 / 47.8	21.7 / 21.7
	Pi	60.0 / 56.1	54.0 / 49.4	67.4 / 56.5	41.3 / 43.5	30.4 / 21.7
	OpenClaw	**65.2 / 61.3**	**56.0 / 54.9**	**69.6 / 60.9**	**58.7 / 52.2**	**37.0 / 32.6**

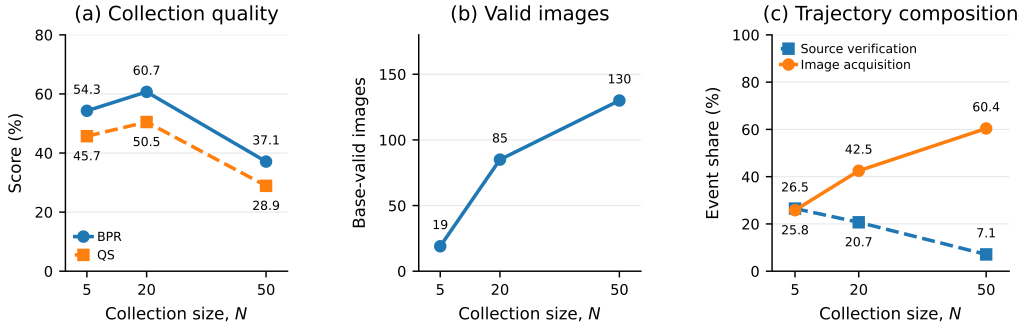


Figure 5: Effect of collection size on Codex under Visual Spec., with $N = 5, 20, 50$ images requested per task on the same seven tasks. (a) Collection validity and quality, measured by BPR and QS. (b) Total number of images satisfying all base criteria across the seven tasks. (c) Shares of trajectory events devoted to source verification and image acquisition, computed within each task and then averaged.

pared with 50.4% for Kimi Code, while its TSR@5 is 17.4%, compared with 10.9%. With GPT-5.6-Sol under Text-only, OpenClaw achieves a TSR@5 of 37.0%, compared with 21.7% for Codex.

Kimi Code achieves higher BPR and QS than Pi in both settings. With GPT-5.6-Sol, Pi outperforms Codex on these metrics under Text-only, whereas Codex leads under Visual Specification.

Trajectory analysis shows that OpenClaw consistently performs more *Search & Verification* across backbones and input settings (Figure 4). With Kimi K3 under Visual Specification, it exceeds Kimi Code in these activities on 95.6% of the matched tasks, while its total tool activity is higher on only 51.1%. These patterns are consistent with the hypothesis that candidate discovery and source verification contribute to collection quality.

5.5 Scaling with Collection Size

We evaluate Codex under *Visual Specification* on the same seven tasks, one per domain, with collection sizes $N \in \{5, 20, 50\}$. Figure 5(a)(b) shows that larger requests yield more valid images, but normalized collection validity and quality peak at N=20. Increasing the collection size from $N = 5$ to $N = 20$ improves BPR from 54.3% to 60.7% and QS from 45.7% to 50.5%. At $N = 50$, however, the two metrics fall to 37.1% and 28.9%, respectively. The total number of base-valid images continues to grow across the three collection sizes, from 19 to 85 and then to 130.

The trajectory analysis in Figure 5(c) shows that Codex increasingly prioritizes image acquisition over source verification. The task-averaged share of acquisition events rises from 25.8% at $N = 5$ to 60.4% at $N = 50$, while the share of source-verification events falls from 26.5% to 7.1%. Verification also becomes less widespread: it is recorded in all seven tasks at $N = 5$, five tasks at $N = 20$, and only one task at $N = 50$. Between the two larger quotas, acquisition events increase from 308 to 643, whereas source-verification events decrease from 179 to 43. The reduction in verification activity coincides with the decline in collection validity and quality, suggesting that reduced source checking may contribute to the degradation.

Table 3: Performance under single-task and Joint-5 execution on the same 45 tasks. All values are percentages.

Agent	Setting	BPR	QS	TSR@3	TSR@4	TSR@5
Codex	Single	58.2	51.0	57.8	48.9	22.2
	Joint-5	45.3	37.7	48.9	33.3	4.4
Kimi Code	Single	55.1	48.8	57.8	42.2	15.6
	Joint-5	56.4	51.8	62.2	44.4	20.0
MiniMax	Single	41.3	36.1	35.6	26.7	11.1
	Joint-5	29.8	24.7	28.9	11.1	4.4

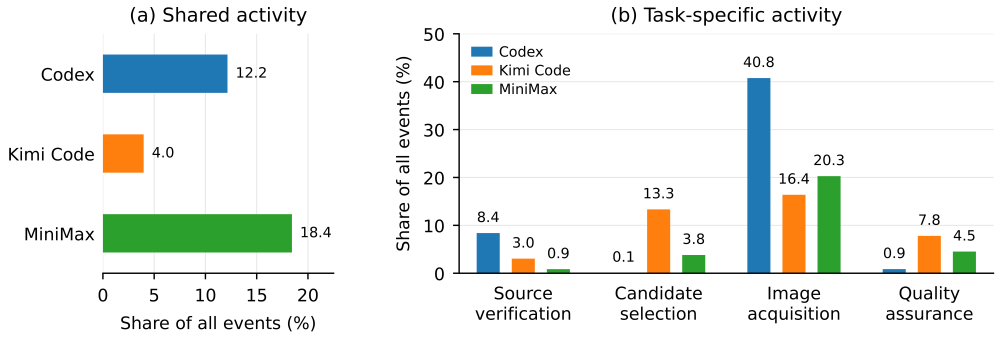


Figure 6: Joint-5 trajectory profiles across nine groups of five tasks per agent. (a) Proportion of events shared across multiple tasks. (b) Proportions of four selected task-specific activities. All percentages use the total number of unique events pooled across the nine groups as the denominator, with shared and unassigned events included exactly once.

5.6 Multi-task Data Sourcing

We compare Codex, Kimi Code, and MiniMax under independent single-task execution and *Joint-5*, where five sourcing tasks are issued in one request. Both settings use the same 45 tasks under *Visual Specification*, with five images requested per task. As shown in Table 3, joint execution affects agents differently. Kimi Code improves in both BPR and QS, whereas Codex and MiniMax experience degradation under Joint-5. The results indicate that multi-task execution introduces additional coordination challenges whose impact depends on the agent framework.

The trajectory profiles reveal distinct ways of organizing the joint request (Figure 6). Kimi Code delegates constituent tasks to separate subagents, resulting in the lowest proportion of shared activity and greater task-specific emphasis on candidate selection and quality assurance. Codex is substantially more acquisition-heavy, whereas MiniMax shows the largest amount of shared activity and little task-specific source verification. The results suggesting that task decomposition may help maintain collection quality in multi-task settings.

6 Conclusion

We introduced DataResearchBench to evaluate whether generalist agents can transform high-level requirements into structured visual data collections with source information and factual annotations. Human-reviewed, task-specific rubrics assess sample validity and quality, complemented by task-level success and trajectory analyses. Our results identify consistent base validity as a central challenge, with collection outcomes varying across visual specifications and agent frameworks. Our scaling study reveals expanding acquisition alongside contracting verification coverage, while joint-task results highlight decomposition with dedicated subagents as a promising design direction. In summary, DataResearchBench provides a testbed for evaluating complete agents and investigating how verification, candidate selection, and task orchestration can support more reliable data sourcing.

AI USE STATEMENT

We used generative AI tools for literature review, manuscript drafting and language refinement. Large Language models were also used for candidate-task generation, preliminary rubric drafting, agent execution, automated judging, and trajectory classification, as described in the paper. The authors reviewed and revised AI-assisted manuscript content. Benchmark tasks and rubrics were finalized through human curation, and automated base-validity judgments were compared with human assessments. The authors retain responsibility for the accuracy and integrity of the final text, analyses, and research artifacts.

ETHICS STATEMENT

This work studies task-directed visual data collection from public web resources rather than exhaustive website crawling. The main evaluation requests five images per task, with larger quotas examined on a seven-task subset. Our MCP tools do not include dedicated mechanisms for circumventing website anti-bot protections, and requests remain subject to website-side access restrictions. Source URLs and contextual information are retained for provenance.

REPRODUCIBILITY STATEMENT

Section 3.2 describes benchmark construction, while Appendix B presents the task catalog and representative examples. Section 5.1 details the experimental settings and common tool interface. Appendix A documents the judging prompts, score computation, human validation, and diagnostic procedures. Appendix C reports complete benchmark metrics, run-to-run stability, additional collection-size scaling results, and source and cost diagnostics.

REFERENCES

Anthropic. Claude code. https://github.com/anthropics/claude-code, 2026a. Version 2.1.232, used in our experiments.

Anthropic. Introducing claude opus 5, 2026b. URL https://www.anthropic.com/news/claude-opus-5. Released July 24, 2026.

Dongping Chen, Ruoxi Chen, Shilin Zhang, Yinuo Liu, Yaochen Wang, Huichi Zhou, Qihui Zhang, Yao Wan, Pan Zhou, and Lichao Sun. MLLM-as-a-Judge: Assessing multimodal llm-as-a-judge with vision-language benchmark. *arXiv preprint arXiv:2402.04788*, 2024.

Mingxuan Du, Benfeng Xu, Chiwei Zhu, Licheng Zhang, Xiaorui Wang, and Zhendong Mao. DeepResearch Bench: A comprehensive benchmark for deep research agents. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/465f22be10e07b301c6ed58f0472f704-Abstract-Conference.html.

Earendil Works. Pi coding agent. https://pi.dev/news/releases/0.84.4, 2026. Version 0.84.4, released August 28, 2026.

Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Boyu Gou, Zanming Huang, Yuting Ning, Yu Gu, Michael Lin, Weijian Qi, Andrei Kopanev, Botao Yu, Bernal Jimenez Gutierrez, Yiheng Shu, Chan Hee Song, Jiaman Wu, Shijie Chen, Hanane Moussa, Tianshu Zhang, Jian Xie, Yifei Li, Tianci Xue, Zeyi Liao, Kai Zhang, Boyuan Zheng, Zhaowei Cai, Viktor Rozgic, Morteza Ziyadi, Huan Sun, and Yu Su. Mind2web 2: Evaluating agentic search with agent-as-a-judge. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-5778.

Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. WebVoyager: Building an end-to-end web agent with large multimodal models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6864–6890, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.371. URL https://aclanthology.org/2024.acl-long.371/.

Peizhou Huang, Zixuan Zhong, Zhongwei Wan, Donghao Zhou, Samiul Alam, Xin Wang, Zexin Li, Zhihao Dou, Li Zhu, Jing Xiong, Chaofan Tao, Yan Xu, Dimitrios Dimitriadis, Tuo Zhang, and Mi Zhang. MMDeepResearch-Bench: A benchmark for multimodal deep research agents. *arXiv preprint arXiv:2601.12346*, 2026. URL https://arxiv.org/abs/2601.12346.

Abhinav Java, Ashmit Khandelwal, Sukruta Midigeshi, Aaron Halfaker, Amit Jayant Deshpande, Navin Goyal, Ankur Gupta, Nagarajan Natarajan, and Amit Sharma. Characterizing deep research: A benchmark and formal definition. In *International Conference on Learning Representations*, pp. 10315–10349, 2026.

Kimi. Kimi k3: Open frontier intelligence, 2026a. URL https://www.kimi.com/en/blog/kimi-k3.

Kimi. Kimi code cli. https://www.kimi.com/code/docs/en/kimi-code-cli/release-notes/changelog.html, 2026b. Version 0.36.1, released August 14, 2026.

Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Russ Salakhutdinov, and Daniel Fried. VisualWebArena: Evaluating multimodal agents on realistic visual web tasks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 881–905. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.50. URL https://aclanthology.org/2024.acl-long.50/.

Keyu Li, Mohan Jiang, Dayuan Fu, Yunze Wu, Xiangkun Hu, Dequan Wang, and Pengfei Liu. DatasetResearch: Benchmarking agent systems for demand-driven dataset discovery. *arXiv preprint arXiv:2508.06960*, 2025. URL https://arxiv.org/abs/2508.06960.

Yijiang Li, Kunal Kotian, Ali Marjaninejad, Meir Friedenberg, Kaushik Pavani, and Sunny Dasgupta. RMIR: A benchmark dataset for reasoning-intensive multimodal image retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Zheyuan Liu, Cristian Rodriguez-Opazo, Damien Teney, and Stephen Gould. Image retrieval on real-life images with pre-trained vision-and-language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2125–2134, October 2021. URL https://openaccess.thecvf.com/content/ICCV2021/html/Liu_Image_Retrieval_on_Real-Life_Images_With_Pre-Trained_Vision-and-Language_Models_ICCV_2021_paper.html.

Tianyi Ma, Yiyue Qian, Zheyuan Zhang, Zehong Wang, Xiaoye Qian, Feifan Bai, Yifan Ding, Xuwei Luo, Shinan Zhang, Keerthiram Murugesan, Chuxu Zhang, and Yanfang Ye. AutoData: A multi-agent system for open web data collection. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://papers.nips.cc/paper_files/paper/2025/hash/fd489a44f3bcb9f122e4931ef21d0c43-Abstract-Conference.html.

Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. GAIA: A benchmark for general ai assistants. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/25ae35b5b1738d80f1f03a8713e405ec-Abstract-Conference.html.

MiniMax. Minimax m3, 2026a. URL https://www.minimax.io/models/text/m3.

MiniMax. Minimax code. https://github.com/MiniMax-AI/minimax-code, 2026b. Version 3.0.68, used in our experiments.

Model Context Protocol Contributors. Model context protocol specification. https://modelcontextprotocol.io/specification/2025-06-18, 2025. Accessed September 2026.

OpenAI. GPT-5 Mini. https://developers.openai.com/api/docs/models/gpt-5-mini, 2025. OpenAI API model documentation.

OpenAI. Codex cli. https://github.com/openai/codex, 2026a. Version 0.151.0 (rust-v0.151.0), used in our experiments.

OpenAI. Introducing gpt-5.5, 2026b. URL https://openai.com/index/introducing-gpt-5-5/. Released April 23, 2026.

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

OpenClaw Contributors. Openclaw. https://github.com/openclaw/openclaw/releases/tag/v2026.7.1-2, 2026. Version 2026.7.1-2, commit 0790d9f, used in our experiments.

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

SpaceXAI. Introducing grok 4.6, 2026a. URL https://x.ai/news/grok-4-6. Released August 12, 2026.

SpaceXAI. Grok build. https://x.ai/build/changelog, 2026b. Version 1.0.5, released August 15, 2026.

Haoran Sun, Haoyu Bian, Shaoning Zeng, Yunbo Rao, Xu Xu, Lin Mei, and Jianping Gou. DatasetAgent: A novel multi-agent system for auto-constructing datasets from real-world images. *arXiv preprint arXiv:2507.08648*, 2025. URL https://arxiv.org/abs/2507.08648.

Nam Vo, Lu Jiang, Chen Sun, Kevin Murphy, Li-Jia Li, Li Fei-Fei, and James Hays. Composing text and image for image retrieval - an empirical odyssey. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6439–6448, 2019. doi: 10.1109/CVPR.2019.00660.

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

Rui Yang, Lin Song, Yanwei Li, Sijie Zhao, Yixiao Ge, Xiu Li, and Ying Shan. GPT4Tools: Teaching large language model to use tools via self-instruction. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-3149. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/e393677793767624f2821cec8bdd02f1-Abstract-Conference.html.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. *arXiv preprint arXiv:2306.05685*, 2023.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. WebArena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/4410c0711e9154a7a2d26f9b3816d1ef-Abstract-Conference.html.

A Evaluation Details

A.1 Submission Validation

Each submitted image is paired with metadata containing its source URL, source context, and task-specific annotation. We first verify that the image can be decoded and that the required metadata fields are present and nonempty. Invalid records receive zero base validity and zero quality score. Missing output slots are handled in the same way and remain in the denominator defined by the requested collection size.

For the main benchmark, each task contributes five requested slots. Valid image–metadata records are subsequently evaluated by the duplicate, provenance, and semantic checks described below.

A.2 Duplicate Checking and Provenance Audit

Duplicate checking. Duplicate detection is performed within each task, comparing submitted images both against other submitted images and against the benchmark reference images. We use file and decoded-pixel hashes to detect exact matches and 64-bit perceptual hashes to identify visually similar pairs, with a Hamming-distance threshold of 6. The detector produces 56 unique candidate pairs, most of which correspond to visually similar frames from the same video. After manual review, 18 submitted images are confirmed as duplicates and invalidated.

Provenance audit. We audit each submitted image–metadata record in three stages: (1) comparing the submitted image with imagery retrieved from the reported source; (2) comparing the submitted source context with independently retrieved source content; and (3) checking whether the submitted image is semantically consistent with that source context. Each stage returns *pass*, *fail*, or *error*. An explicit *fail* or *error* invalidates the sample, while processing errors are recorded separately.

Table 4 summarizes the audit over 5,410 image–metadata records. Most records pass each individual check, while 295 receive at least one failed provenance check.

A.3 Rubric-based Semantic Evaluation

The semantic evaluation uses GPT-5.5 as an image-and-annotation judge. For each retained image–metadata pair, the judge receives the task description, source information, paired annotation, and the task-specific rubric. The judge independently evaluates the five base criteria and the applicable quality criteria, returning one binary decision for each criterion.

The *Image Annotation* criterion evaluates the correctness and task relevance of annotation information associated with an image, as specified by the task-specific rubric. Depending on the task, this may include the paired textual annotation or labels, captions, and other annotations embedded in the image. The evaluator is instructed to assess the image and annotation jointly and all judgments are returned in a fixed JSON format to enable deterministic aggregation.

The complete system and user prompt templates are provided in Listing 1 and Listing 2.

Listing 1: System prompt template used for rubric-based semantic evaluation.

```
You are a strict image-and-annotation quality judge for benchmark
construction.

You will receive one image, its paired annotation, a task description,
and five scoring standards.

Judge whether the image and annotation satisfy the given standards.

For "Image Annotation", follow the task-specific scoring standard to
    determine whether to assess the paired annotation text, labels,
    captions, or other annotations embedded in the image, or both.

For the other dimensions, judge the image and use the annotation only
```

```
as supporting context when relevant.

Return only valid JSON.

Each dimension value must be an object with exactly two keys:
"judgment" and "reason".

The "judgment" value must be exactly "yes" or "no".

{reason_instruction}

If a standard is "None", "none", empty, or not applicable,
mark that dimension as "yes".

Do not add markdown or extra keys.
```

Listing 2: User prompt template used for rubric-based semantic evaluation.

```
Task ID: {task_id}
Task name: {task_name}
Task description: {task_description}

Source URL:
{source_url}

Original source context:
{source_context}

Paired annotation:
{annotation}

Judgment type: {judge_type}

For each of the five dimensions, judge whether this image and its
paired annotation satisfy the corresponding {judge_type} standard.

Standards:
{standards_json}

Return exactly this JSON object shape:

{
  "Task Relevance": {"judgment": "yes", "reason": ""},
  "Image Quality": {"judgment": "yes", "reason": ""},
  "Image Style": {"judgment": "yes", "reason": ""},
  "Image Annotation": {"judgment": "yes", "reason": ""},
  "Task Constraint Compliance": {"judgment": "yes", "reason": ""}
}
```

A.4 Human Validation of the Judge

We randomly sample 116 image–metadata pairs from the five main agents, with 58 examples from Text-only and 58 from Visual Specification. Three human reviewers independently assess each sample using the finalized task-specific base criteria without access to the automated judge's decisions.

Agreement between each reviewer and the automated judge on final image-level base validity is 86.21%, 81.03%, and 88.79%, respectively, corresponding to a mean agreement of 85.3%. These results indicate substantial agreement between the automated judge and human reviewers.

A.5 Agent Instruction Prompt

All evaluated agents receive the same task-independent instruction for the image data collection task. The instruction specifies the collection objective, MCP tool usage, output requirements, and anno-

Table 4: Provenance-audit outcomes across all evaluated image–metadata records.

Audit stage	Pass	Fail	Error
Image–source match	5,392	17	1
Source-context fidelity	5,270	140	0
Image-context consistency	5,261	148	1
Overall	5,114	295	1

tation constraints. Task-specific information, including the task name, task description, requested image count, and reference images when applicable, is provided separately.

Listing 3 shows the complete instruction provided to agents. The same instruction is used for both *Text-only* and *Visual Specification* settings; the only difference is whether reference images are included as additional task inputs.

Listing 3: Task-independent instruction provided to evaluated agents.

```
Execute this data collection task immediately. Collect image data from
websites according to the provided task name, task description, and
reference images when available.

For every collected image, preserve its source context and write a
factual, task-relevant annotation.

You must use the provided MCP tools to obtain any images.

Do not generate or synthesize images.

Continue until all required images and their corresponding annotation
JSON files have been saved to the output directory.

Before collecting, identify meaningful coverage dimensions based on the
task description and any available reference images.

Determine the appropriate coverage dimensions and allocate the required
number of images across them according to the specific task and the
required image count.

Each image must have one JSON metadata file with the same basename.
The metadata file must contain:

image_file:
   Exact image filename including extension.

source_url:
   Original image or source-page URL.

source_context:
   Relevant original title, caption, surrounding text, or other
   source context used to understand the image.

annotation:
   Factual annotation derived from the image and its original source
   context, written for the supplied task.

Do not invent source context or unsupported facts. Keep source context
separate from the task-relevant annotation.
```

A.6 Trajectory Analysis Details

Framework-specific parsers convert recorded execution logs into a common sequence of text and tool events. Streamed or repeated log fragments are consolidated, and tool results are associated with their corresponding calls. Qwen3.5-9B assigns each event one primary behavior category using the

Table 5: Mapping from trajectory categories to grouped behaviors.

Primary category	Analysis group
Planning	Planning
Reference Analysis	–
Candidate Search	Search & Verification
Source Verification	Search & Verification
Candidate Selection	Selection & Acquisition
Image Acquisition	Selection & Acquisition
Annotation and Delivery	Delivery & QA
Quality Assurance	Delivery & QA
Environment and Workflow	–
Other	–

Table 6: Distribution of the core benchmark tasks across application domains.

Domain	Abbr.	Tasks	Share (%)
Science & Academia	Sci.	14	30.43
Health & Medicine	Health	8	17.39
Entertainment & Culture	Ent.	7	15.22
News & Current Affairs	News	5	10.87
Art & Design	Art	4	8.70
Geography & Travel	Geo.	4	8.70
Business & Economics	Bus.	4	8.70
Total		46	100.00

task description, surrounding events, tool arguments, and tool-result context. Classification follows the purpose of the action; for example, reading a webpage to establish provenance is labeled *Source Verification*, whereas judging whether an image should be retained is labeled *Candidate Selection*.

For the grouped analyses in the main text, the primary categories are mapped as follows:

The Visual Specification comparison uses all recorded events and reports changes in grouped event counts on matched tasks. Framework comparisons use tool events on the 45 tasks with complete trajectories across the compared configurations. The collection-size analysis uses task-averaged behavior shares together with raw event counts and task coverage. For Joint-5 execution, each event is additionally attributed to one constituent task, shared across multiple tasks, or left unassigned; each event contributes once to the pooled statistics.

A.7 Source Diagnostics

Source diagnostics are computed from the provenance metadata returned with submitted images. We extract the network location from each nonempty source URL, convert it to lowercase, and remove a leading `www.`; other subdomains are retained. Each submitted metadata record contributes one source count, including repeated URLs. Missing and malformed source URLs are tracked separately. These normalized domains are used for the source-distribution analysis reported in Appendix C.

B Benchmark Details

B.1 Benchmark Examples

Figures 7, 8, and 9 presents a representative task from different application domains. The example includes the natural-language data requirement and the reference images provided in the Visual Specification setting. The example illustrates the diversity of visual forms and sourcing requirements covered by DataResearchBench.

task 011 *PCB / electronic schematic rendered image collection*

task_description:
Collect images of electronic schematic diagram for testing visual localization, text referring, and connectivity reasoning.

Task Relevance:
base: Images must primarily depict electronic schematic diagrams, not photos of physical boards. The circuit content should be the main visual subject and must include recognizable symbols, wires/traces, component connections, and logical relationships.
quality: Images should include moderately complex to dense schematic diagrams where multiple components, crossings, branches, or multi-part modules require genuine structural reasoning. High-quality samples may include visually similar symbols, long-distance net references, repeated component patterns, or mixed schematic conventions, as long as connectivity can still be resolved from the image itself.

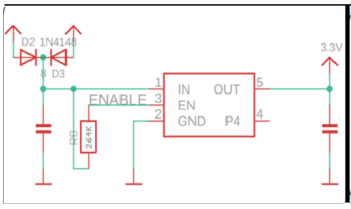
Image Quality:
base: Images must be clear enough that small labels, connection lines, junctions, symbol boundaries, and wire endpoints are visually distinguishable. Reject images with blur, severe compression, unreadable text, extreme low contrast, heavy cropping of schematic content, or glare/perspective artifacts that prevent reliable reading of connectivity or text targets.
quality: Images should preserve fine-grained engineering detail across the full schematic, with crisp text, clear junction dots, legible pin labels, and unambiguous wire crossings. High-quality samples should avoid watermarks, overlaid logos, editor chrome occluding content, and other interference on top of the schematic region.

Image Style:
base: Images should be rendered engineering schematics. Natural photos, handwritten sketches, artistic circuit illustrations, PCB layout screenshots, or images dominated by software UI, documentation pages, tables, plots, or other non-diagram content are not acceptable. Combined or multi-panel images are not allowed unless a single panel clearly dominates and serves as one coherent schematic target.
quality: None

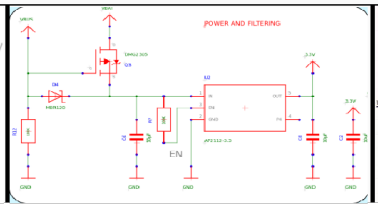
Image Annotation:
base: Annotation information, if provided, must correctly match the schematic content. Any referenced components, nets, pins, or text targets must be visible and consistent with the rendered schematic.
quality: None

Task Constraint Compliance:
base: Images must not contain direct answer cues for the intended reasoning tasks, such as arrows, circles, highlighted nets, selected components, callout boxes, tutorial overlays, or embedded solution text that explicitly identifies a target object or connection.
quality: Images should also avoid hidden shortcuts from instructional contexts, including search highlights, hover tooltips, exam markers, step-by-step annotations, or color-coded emphasis that reveals key text references or connectivity paths without requiring visual reasoning.

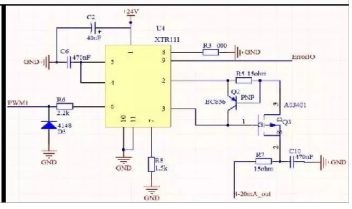
Reference 1



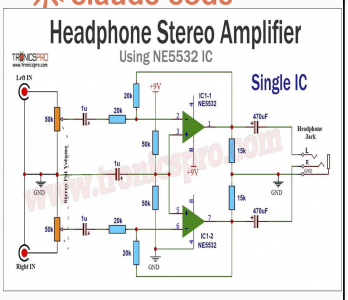
Reference 2



Reference 3



claude code

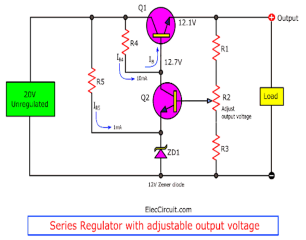


"source_context": "TRONICSpro article 'Stereo Headphone Amplifier Circuit Diagram using NE5532', posted October 20 2023 by Tronicspro. The image is captioned 'Headphone Amplifier Circuit Diagram using NE5532'. The page lists the components……
"annotation": "Hobbyist-style illustrated circuit diagram, 1200x697 px, in a pictorial rendering convention rather than an EDA-tool export: green wires, light-blue rectangles for resistors, orange rectangles for potentiometers……

image quality(quality): Diagonal large watermark covers circuit and interconnect areas

base ✓ quality 0.66

codex

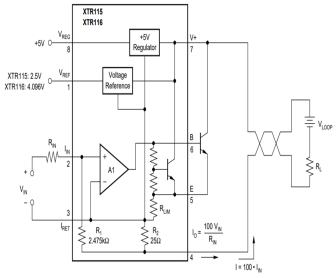


"source_context": "ElecCircuit page titled \"Power Supply Circuit Diagram & Basic Principles for Beginners.\" The page explains linear supplies and includes the figure captioned \"Series Regulator with adjustable ……
"annotation": "Instructional schematic of an adjustable series linear regulator. A 20 V unregulated input feeds series transistor Q1, whose output node is labeled 12.1 V and connects to the load. Q2 is tied to the R1-R2-R3 output-sensing divider; R2 is explicitly labeled as the output-voltage adjustment……

image constraint(base/quality): The blue current path arrows, numerical values, and adjustment tips provide instructional reasoning shortcuts.

base ⊗ quality 0

kimi code



"source_context": "Image attached to the Electrical Engineering Stack Exchange question \"RTD sensor circuit design\" (https://electronics.....
"annotation": "RTD temperature-sensor signal-conditioning schematic. A \"1mA Current Source\" block built from NMOS transistors Q1/Q2 (Q_NMOS_DGS) and resistor R9 drives the RTD element labeled Rt1_1 Thermistor. The resulting voltage feeds a two-stage op-amp chain (U1A and U1B, TEMP_EN power pin) with 300R level-shift resistors R1_1/R1_2, 5K11 feedback/gain resistors R2, R3, R4, producing analog output TEMP_AO. Dashed boxes group functional stages……

base ✓ quality 1.0

Figure 7: Task011's information and result.



Figure 8: Task030’s information and result.



Figure 9: Task053's information and result.

B.2 BENCHMARK TASK CATALOG

DataResearchBench contains 46 visual data-sourcing tasks spanning seven application domains. We use *Art* for Art & Design, *Bus.* for Business & Economics, *Ent.* for Entertainment & Culture, *Geo.* for Geography & Travel, *Health* for Health & Medicine, *News* for News & Current Affairs, and *Sci.* for Science & Academia. Table 6 summarizes the domain distribution and abbreviations used in the paper.

Table 7 lists all 46 core tasks. Task identifiers retain their construction-stage indices and are therefore not consecutive.

Table 7: Complete core task set of DataResearchBench.

ID	Task Summary	Domain
001	Scientific experimental data visualization images	Sci.
002	Screenshots of user posts from social media platforms	Ent.
003	Official mathematics competition collection	Sci.
005	Collection of multi-view, multi-modality radiology case image sets	Health
006	Search-based multi-turn VQA with real-scene images and QA pairs	Art
007	Aligned high-resolution aerial/satellite imagery collection for terrace farmland scene segmentation	Sci.
008	Multimodal Clinical Medical Imaging Sample Collection	Health
009	Parsable tables and quantitative figures from academic papers with contextual text	Sci.
010	Street-view street and environment images for visual cue combination	Geo.
011	PCB / electronic schematic rendered image collection for visual grounding, text referring, and connectivity reasoning	Sci.
012	High-fidelity scanned/screenshot single-question exam images with multilingual scripts and original layouts	Sci.
013	Human hand joint motion understanding	Sci.
015	Online puzzle or brain-teaser game level screenshots and initial-state images	Sci.
016	Public multimodal clinical image sample collection for testing visual dependence	Health
019	News events and public figures images	News
020	Visual misalignment image collection	Ent.
021	Clinical image and report collection	Health
023	Synthetic sandbox-style moral dilemma rendered images	Ent.
024	Cluttered tabletop grasping scenes with language instructions and object/grasp annotations	Sci.
025	Laboratory static image collection: safety signage and transparent vessels	Health
026	Collect children/family social story illustrations	Art
027	Global street-view panorama collection for single-frame geolocation	Geo.
028	Collect real-world visual anomaly photos (biological or structural abnormalities)	Health
029	Bi-temporal high-resolution building instances with pre/post-disaster imagery	Geo.
030	Visual Rebus Puzzle Understanding	Ent.
034	Slide Layout Failure Images	Art
035	Collect mobile app full-screen UI screenshots	Sci.
040	Desktop full-page screenshots of editorial, blog, and portfolio pages	Art.
041	Political meme collection for harmfulness analysis	News
044	Synthetic Hateful Image Collection for Localization and Dehatification	Ent.

Table 8: Complete TSR results for the five main agents. Entries are percentages, with successful-task counts in parentheses. Each setting contains 46 tasks with five requested images per task.

Agent	TSR@1	TSR@2	TSR@3	TSR@4	TSR@5
Text-only					
Claude Code	91.3 (42)	84.8 (39)	80.4 (37)	69.6 (32)	39.1 (18)
Grok	82.6 (38)	73.9 (34)	67.4 (31)	56.5 (26)	34.8 (16)
Codex	82.6 (38)	71.7 (33)	65.2 (30)	50.0 (23)	21.7 (10)
Kimi Code	82.6 (38)	67.4 (31)	52.2 (24)	39.1 (18)	10.9 (5)
MiniMax	76.1 (35)	45.7 (21)	21.7 (10)	17.4 (8)	6.5 (3)
Visual Spec.					
Claude Code	97.8 (45)	91.3 (42)	87.0 (40)	69.6 (32)	45.7 (21)
Grok	93.5 (43)	91.3 (42)	76.1 (35)	47.8 (22)	32.6 (15)
Codex	89.1 (41)	73.9 (34)	56.5 (26)	47.8 (22)	21.7 (10)
Kimi Code	84.8 (39)	76.1 (35)	56.5 (26)	41.3 (19)	15.2 (7)
MiniMax	73.9 (34)	60.9 (28)	37.0 (17)	26.1 (12)	10.9 (5)

ID	Task Summary	Domain
048	False or Misleading Claim Evidence Images from Public Social Posts	News
049	Collect still images showing on-screen stock tickers, charts, or price overlays	Bus.
050	Still images of presenter body language and hand gestures during stock recommendations	Bus.
053	Non-person political news images	News
058	Public candlestick chart screenshots for visual perception and numerical reasoning	Bus.
059	Map-based financial charts and research figure screenshots	Bus.
060	Geometry diagrams with clear symbols and measurements	Sci.
061	Scanned contest problem pages with equations and embedded diagrams	Sci.
062	Colorful Counting and Arithmetic Scene Images	Sci.
066	Traditional ceremonies, festivals, artifacts, and folk arts image collection	Ent.
068	Traditional Korean Foods and Snacks Image Collection	Ent.
070	AI-generated or edited visual paraphrases in news/social contexts	News
072	De-identified CT, MRI, and X-Ray radiology image collection	Health
073	Ophthalmic Fundus and OCT Still Images for Modality and Lesion Grading	Health
075	Aerial images of urban sports complexes and courts	Geo.
077	Aerial land-cover patches for spatial reasoning	Sci.

C Additional Experimental Results

C.1 Complete Main-Benchmark Metrics

The main text reports aggregate BPR, QS, and TSR trends across agents. We provide the complete task success values in Table 8, including the number of tasks satisfying each collection threshold. These values complement Figure 2 by providing the exact task counts underlying the threshold curves. The reported metrics incorporate duplicate checking, provenance auditing, and rubric-based semantic evaluation.

To separate collection validity from the quality of samples that already satisfy the base criteria, Table 9 reports BPR, QS, and conditional quality measured by QS/BPR. This ratio gives the average quality score among base-valid images. We compute the ratio from the unrounded aggregate scores and counts before rounding for display.

The complete TSR results show that collection-level completion decreases rapidly as the required number of valid images increases. The quality decomposition further shows that conditional quality

Table 9: Decomposition of base validity and conditional quality. BPR measures the fraction of requested images passing the validity checks and all base criteria. QS/BPR reports the average quality score among base-valid images. All entries are percentages.

	Text-only			Visual Spec.		
Agent	BPR	QS	QS/BPR	BPR	QS	QS/BPR
Claude Code	73.0	67.8	92.8	78.3	71.5	91.4
Grok	63.0	58.0	92.0	68.3	62.6	91.7
Codex	58.3	50.8	87.1	57.8	50.7	87.7
Kimi Code	50.4	45.7	90.6	54.8	48.6	88.8
MiniMax	33.5	28.3	84.6	41.7	36.5	87.3

Table 10: Codex performance across three independent runs. Each entry reports BPR / QS in percentages. Run 1 corresponds to the result reported in the main benchmark. The final column reports the mean across the three runs.

Setting	Run 1	Run 2	Run 3	Mean
Text-only	58.3 / 50.8	59.1 / 50.8	57.4 / 47.9	58.3 / 49.8
Visual Spec.	57.8 / 50.7	57.0 / 50.4	59.1 / 49.2	58.0 / 50.1

varies less than base validity across agents, indicating that consistently producing base-valid samples remains the dominant challenge.

C.2 Run-to-Run Stability

The main benchmark reports one execution per agent and input setting. To examine run-to-run variation, we independently repeat the Codex evaluation twice under the same task set and experimental configuration, resulting in three runs over the same 46 benchmark tasks. Table 10 reports the aggregate BPR and QS across the three runs.

The three executions show limited variation in aggregate performance. Across runs, BPR varies by 1.7 percentage points under *Text-only* and 2.2 percentage points under *Visual Specification*, calculated from the base-pass counts before rounding. QS varies by approximately 2.9 and 1.5 percentage points, respectively. These results show consistent aggregate performance for Codex across the three independent executions.

C.3 Collection-Size Scaling

The collection-size scaling experiment evaluates Codex under *Visual Specification* on tasks `003`, `006`, `008`, `010`, `019`, `023`, and `050`, with one task from each benchmark domain. The same seven tasks are used at all three collection sizes, $N \in \{5, 20, 50\}$. Table 11 reports the aggregate results. BPR and QS are computed using the same evaluation procedure as the main benchmark, with $7N$ requested images as the denominator.

Increasing the requested collection size from $N = 5$ to $N = 20$ improves both BPR and QS. At $N = 50$, the number of base-valid images continues to increase, but both normalized metrics fall below their $N = 5$ values. Larger requests thus produce more valid images while reducing the proportion of requested images that satisfy the evaluation criteria.

C.4 Multi-Task Evaluation Details

Under *Joint-5*, each request contains five tasks, with five images requested per task. The same grouping is used for all three agents. Task `077` forms a singleton group and is excluded from both the *Single* and *Joint-5* comparisons. Both settings therefore use the same 45 tasks, with 225 images requested per agent. Under *Single*, each of these tasks is executed independently. BPR and QS are computed over the 225 requested images, and TSR is computed over the 45 constituent tasks using the same evaluation procedure as the main benchmark.

Table 11: Collection-size scaling results for Codex under Visual Specification. Each setting contains the same seven tasks, with N images requested per task. Requested and base-valid image counts are aggregated across the seven tasks. BPR and QS are percentages.

N	Requested images	Base-valid images	BPR	QS
5	35	19	54.3	45.7
20	140	85	60.7	50.5
50	350	130	37.1	28.9

C.5 Source and Cost Diagnostics

We provide additional diagnostics on the sources used for data collection and the model API expenditure during evaluation. Source statistics are extracted from submitted provenance metadata and computed after normalizing the recorded source URLs. The source distributions characterize the types of public resources used by different agents.

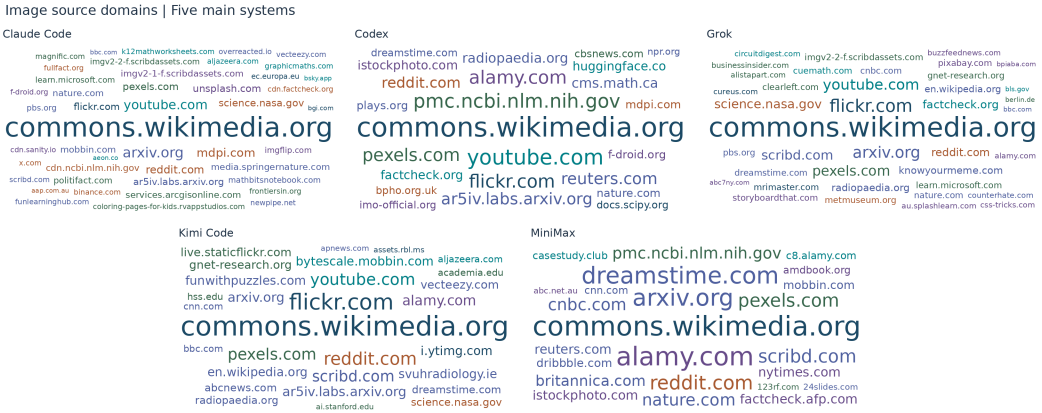


Figure 10: Recorded source-domain distributions for the five main agents. Each panel shows a word cloud over normalized source domains appearing in the submitted collections. Word size is proportional to the number of submitted images attributed to each domain.

Figure 10 shows that the evaluated agents rely on overlapping public sources while exhibiting different preferences among secondary sources. These distributions provide additional context for understanding how agents construct visual collections from the open web.

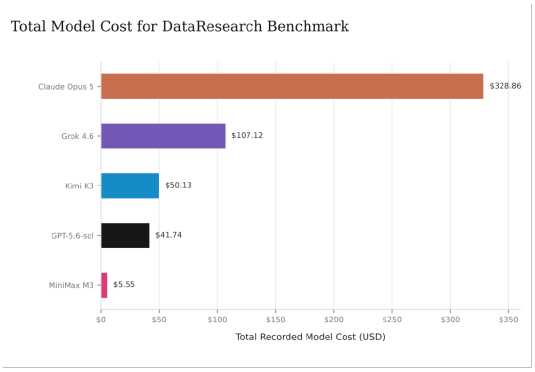


Figure 11: Total model API cost for evaluating each agent on the 46 benchmark tasks under both Text-only and Visual Specification.

Figure 11 reports the total model API expenditure for each evaluated agent. Reported costs include model API charges only and exclude search or other external tool-service costs. Costs are computed from logged input and output tokens using the public API prices available at the experiment date.