MANGABENCH: CAN MULTIMODAL MODELS REASON ACROSS THOUSANDS OF MANGA PAGES?

Anonymous authors
Paper under double-blind review

ABSTRACT

As Multimodal Large Language Models (MLLMs) scale to context windows of up to one million tokens, many long-context benchmarks increasingly test whether models can effectively use their context. However, evaluating model understanding across long, connected visual narratives remains an open challenge, and recent benchmarks fail to evaluate model reasoning over content beyond the context window limit. To address this gap, we introduce MangaBench, a benchmark for evaluating reasoning and understanding over long visual stories. Its tasks require models to aggregate evidence and to track entities, relationships, and states across volumes. Manga is a particularly challenging medium because its information-dense panels depict selected narrative moments, and later scenes may depend on previous visual details. MangaBench pairs manually constructed questions with an evaluation of multimodal models using native agents and an RLM-inspired reader. We compare task performance across model–harness configurations. Our evaluation shows that even frontier models using these harnesses struggle with counting and listing, and with state tracking across volumes. These results highlight the need for models and agent harnesses that can more effectively gather visual evidence to reason over extended narratives.

1 INTRODUCTION

How many pages does it take to settle a rivalry? In manga—Japanese comics—the answer can be in the thousands. Following these stories requires readers to connect visual and textual evidence across thousands of pages, tracking characters and relationships alike. Manga offers a natural, content-rich setting for evaluating long-context understanding in Multimodal Large Language Models (MLLMs): can they identify relevant details from distant scenes and use them to make sense of the story?

Building such a benchmark presents two challenges: procuring suitable source material and providing reliable annotations. The source material must contain visual details beyond dialogue and narration, with characters and relationships that develop across thousands of pages. To our knowledge, no synthetic dataset provides this combination at the scale of multi-volume manga. To evaluate understanding of these narratives, annotators must first read and understand the story in depth to construct and answer questions that involve combining evidence across scenes, tracking changes, and interpreting the medium's visual and narrative conventions. Manga provides such narratives, but constructing questions and reliable reference answers requires expert knowledge of the story.

Existing manga benchmarks examine visual and narrative understanding at different scales: MangaUB (Ikuta et al., 2025) evaluates single- and multi-panel understanding, while MangaVQA (Baek et al., 2026) focuses on two-page spreads. Both works draw on the Manga109 dataset (Matsui et al., 2017; Aizawa et al., 2020), a collection of individual volumes rather than extended multi-volume narratives. Re:Verse (Baranwal et al., 2025) extends this to chapter-level and multi-page understanding within a single volume. What remains less clear is how well models locate and combine evidence across multiple volumes.

To this end, we introduce MangaBench, a benchmark of 954 manually authored questions over eight manga series, averaging 3,768.4 pages per series. Eight expert annotators read all included volumes of their assigned series and wrote questions designed to require visual information beyond dialogue and narration. For each question, they provided a reference answer, supporting page references,

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

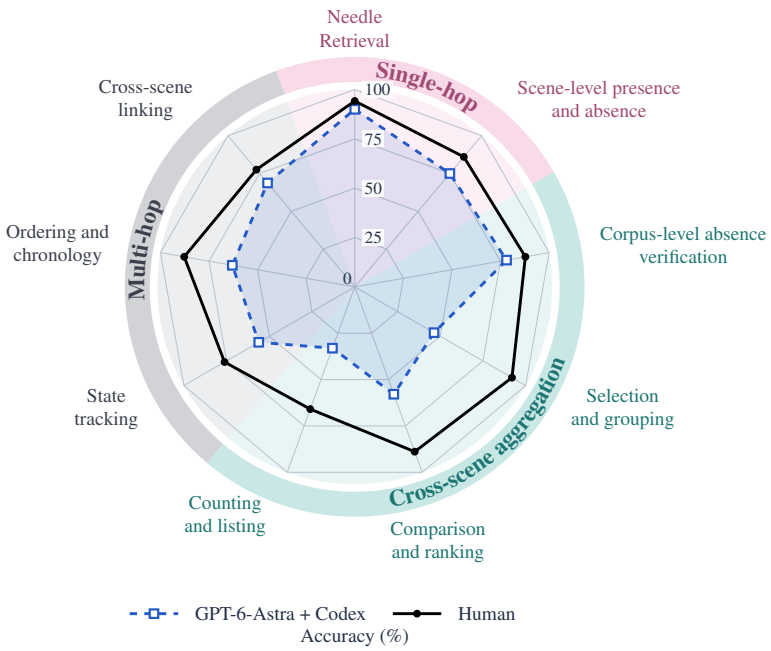


Figure 1: Task types grouped into three families: single-hop tasks require one self-contained piece of evidence; cross-scene aggregation combines independent observations; multi-hop tasks connect dependent observations across scenes. The radar chart compares human accuracy with judge-assisted accuracy for the best model–harness configuration, GPT-6-Astra with Codex.

and an explanation of how the evidence supports the answer. For long-range questions, this explanation also states why each cited scene is needed. A second annotator answered each question independently, then checked its evidence and connections between scenes. MangaBench extends manga evaluation to multi-volume narratives, with annotations that identify the visual evidence and cross-scene connections required for each answer.

We evaluate multimodal models and agent harnesses on MangaBench, using frontier models such as Claude Opus 5.5 and GPT-6-Astra. We compare native agents using their available tools with the MangaBench reader, inspired by Recursive Language Models (RLMs; Zhang et al., 2025), which delegates page inspections to visual readers and combines their findings. We compare task accuracy, with input ablations and closed-book baselines. Figure 1 shows the task breakdown for GPT-6-Astra with Codex alongside human performance. On the full corpus, both native agents and the RLM reader achieve higher accuracy on needle retrieval than on counting, listing, and state tracking. Our analysis of selected failures reveals errors in both gathering evidence and using it to construct an answer. Readers sometimes miss or misread relevant occurrences on inspected pages, while final answers can omit recovered findings or associate events with the wrong entities or times.

These findings point to two priorities for future MLLMs: (1) extracting all relevant evidence and (2) preserving how entities change across scenes. One promising direction is to develop memory mechanisms that retain individual events and state changes, link them to the correct entities and source pages, and support aggregation across scenes.

2 Related Work

Benchmarks for visual media vary in whether their images form a single narrative, and in whether answers require evidence from separate portions of the narrative. Multi-image benchmarks test retrieval, comparison, aggregation, and tracking over image collections or interleaved image–text inputs (Song et al., 2024; Wang et al., 2024; 2025b; Das et al., 2026); images in these benchmarks do not necessarily form an evolving story. Video benchmarks test temporal reasoning, aggregation, and memory (Tapaswi et al., 2016; Wu et al., 2024; Fu et al., 2026; Wang et al., 2026; Jain et al.,

Manga Title	# Volumes	# Chapters	# Pages	# Questions
Blue Giant	10	80	2,034	17
Blue Period	17	76	3,255	102
Chainsaw Man	22	210	4,063	92
Golden Kamuy	31	314	6,155	513
Jujutsu Kaisen	31	276	5,713	193
Land of the Lustrous	13	108	2,570	5
Manchuria Opium Squad	14	132	2,628	15
Steel Ball Run	24	95	3,729	17
Total	162	1,291	30,147	954
Average	20.3	161.4	3,768.4	119.2

Table 1: MangaBench corpus. Pages refer to material available to the reader for each series.

2026), but shortcut solvability has also been documented in video evaluation (Cores et al., 2025; Li et al., 2026). SagaQA (Pennec et al., 2026) specifically targets cross-episode reasoning over extended television narratives. Comics and manga benchmarks evaluate similar capabilities: inference across panels (Iyyer et al., 2017), structural and perceptual analysis (Vivoli et al., 2024), and single- and multi-panel understanding (Ikuta et al., 2025). MangaVQA (Baek et al., 2026) evaluates contextual understanding over two-page manga spreads, while Re:Verse (Baranwal et al., 2025) extends evaluation to multi-page understanding within a single manga volume of 11 chapters and 308 pages.

Long-context text benchmarks provide a different perspective, demonstrating that context length alone does not guarantee long-range reasoning. Needle in a Haystack (Kamradt, 2023) extends the input while keeping the required evidence localized. Recent benchmarks isolate reasoning capabilities through task construction: RULER (Hsieh et al., 2024) includes multi-hop tracing and aggregation, NoLiMa (Modarressi et al., 2025) reduces reliance on literal matching, and OOLONG (Bertsch et al., 2025) emphasizes aggregation over long texts. Similar benchmarks examine reasoning capabilities over long-form narratives: NarrativeQA (Kočiský et al., 2018) evaluates comprehension of books and movie scripts, while NovelHopQA (Gupta et al., 2025) tests multi-hop question answering over long narrative excerpts. Related narrative benchmarks examine plots, recurring entities, and events separated by long narrative distances (Xu et al., 2024; Karpinska et al., 2024; Wang et al., 2025a; Du et al., 2026). Broader long-context suites include LongBench (Bai et al., 2024), LongBench v2 (Bai et al., 2025), and NarraBench (Hamilton et al., 2026) argues that narrative events, perspective, and revelation remain underrepresented. Compared to text-native narrative evaluation, MangaBench additionally requires grounding the relevant entities and events in manga pages.

MangaBench evaluates how models understand manga, from interpreting individual scenes to combining evidence across volumes. All manga series used in MangaBench span thousands of pages, providing the long narrative context needed to evaluate these tasks.

3 MangaBench Design

3.1 Manga Corpus

Our corpus comprises eight manga series, six of them in their official English translations, and two in the original Japanese, covering genres such as adventure, fantasy, thriller, and slice-of-life. We select multi-volume narratives with recurring characters and relationships that develop across chapters. Their detailed artwork supports tracking objects and visual changes across distant scenes. We did not identify openly licensed manga with the narrative scale required for this evaluation. We do not distribute the copyrighted pages, but provide evaluation code, page references, and source-edition details to support evaluation using independently obtained copies. The corpus contains 162 volumes, 1,291 chapters, and 30,147 pages in total. We preserve the original chapter and page order for each series. Each page is scanned at a resolution of 1120×1600 pixels. Scans are stored in volume PDFs, with one scan per PDF page. Table 1 reports corpus statistics and question counts by series. Appendix A lists the included volumes and source editions and explains how pages are counted and referenced.

Question family	What the question requires	Count (%)
Single-hop *When Kishibe says "They're finally away from Quanxi, eh?", what is held in his right hand?*	Locate and interpret one self-contained scene.	282 (29.6%)
Cross-scene aggregation *In which volumes, between 10 and 16, do Asa and Yoshida share scenes together?*	Combine independently interpretable observations from multiple scenes.	327 (34.3%)
Multi-hop *During a news interview, a woman says that Chainsaw Man saved her on a train. When this event happened, who was he fighting?*	Connect dependent observations across scenes.	345 (36.2%)

Table 2: Question families grouped by how evidence should be found and combined.

3.2 Question Design and Annotation

Question Types and Distribution. MangaBench contains 954 questions that assess models' ability to locate and combine visual and textual evidence. Based on how they use this evidence, we group questions into three families described in Table 2: single-hop, cross-scene aggregation, and multi-hop. The task types within these families are shown in Figure 1. Appendix B reports question counts by task type and answer format, while Section 4.3 describes the answer formats and scoring.

Design Criteria. Annotators were instructed to write questions requiring visual information that cannot be derived from dialogue or narration alone, together with a verifiable reference answer. For cross-scene questions, each supporting scene must contribute information needed to answer the question. Questions specify the relevant characters, objects, or story intervals where needed. Subjective judgments, ambiguous wording, and illegible evidence are excluded.

Annotation. Questions were authored by eight expert annotators who read all included volumes for their assigned series. Each question includes a reference answer, covered page range, task labels, and supporting page references. A short explanation states how each cited scene supports the answer. Questions for the two Japanese series were written by two annotators who read Japanese. Each question underwent an inter-annotator review: another annotator first answered independently without seeing the reference answer, then checked the wording, evidence, and connections between scenes. For exhaustive lists and absence questions, another annotator reviewed the full scope for omitted instances or counterexamples. Disagreements prompted a review of the wording and cited pages; a third annotator reviewed unresolved cases, and ambiguous questions were removed.

Quality Control. Following inter-annotator review, the paper authors conducted an additional quality-control review. Overall, we retained 954 of 1,026 candidate questions (93.0%). Among the retained questions, approximately 15% were revised before acceptance. Questions were excluded primarily for unresolved ambiguity, insufficient supporting evidence, or duplication of the same fact using substantially the same evidence.

Evidence Distribution. Across all 954 questions, the mean number of distinct cited pages is 5.52 (median: 3). The median distance between the first and last cited page is 136.5 pages, measured over 954 questions. Appendix B reports these statistics by question family, alongside chapter spans and the proportion of questions with evidence in multiple chapters or volumes.

3.3 Implications of Corpus Scale

Input Constraints. MangaBench requires locating relevant evidence within thousands of pages. Even our smallest corpus, *Blue Giant* (2,034 pages), requires an estimated 1.14 million visual tokens under the lowest estimate in Table 9, exceeding the listed context limits of around one million tokens. Models must therefore inspect the corpus across multiple requests and retain relevant findings between requests. Appendix C details the token estimates and additional PDF input limits.

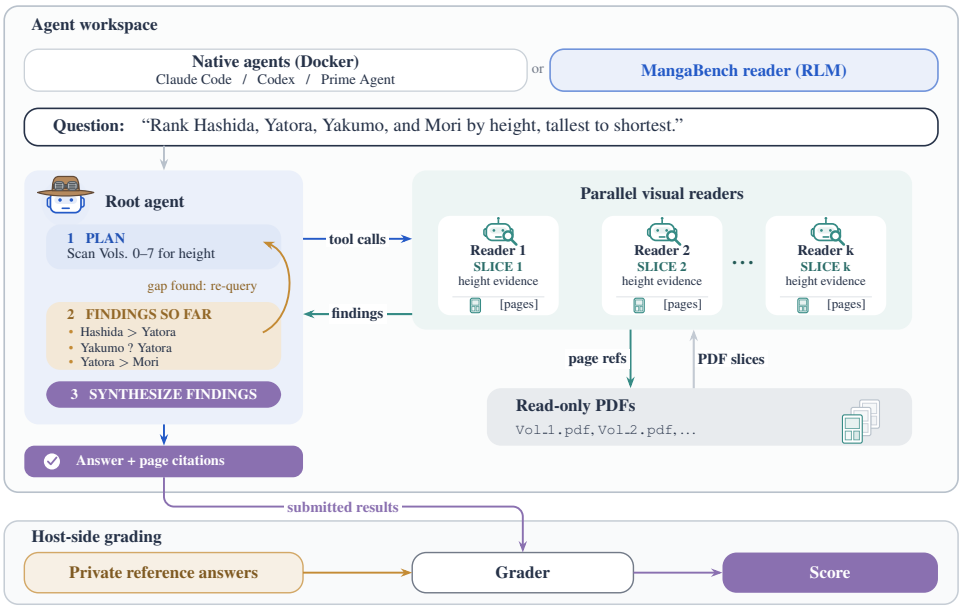


Figure 2: MangaBench evaluation workflow. Native agents or the RLM reader inspect read-only manga PDFs and return answers with page citations. In the illustrated RLM example, the root agent assigns page slices to parallel visual readers, combines pairwise height comparisons, and requests missing evidence before ranking the characters. The root retains the readers' reports rather than the PDFs. Answers are scored on the host against private references, which are inaccessible to agents.

Annotation Effort. Manual annotation is time-consuming at this scale, where each series averages 3,768.4 pages in length. Each annotator produced around 6–7 questions per hour. Even with familiarity with the story, the paper authors spent 200 additional person-hours validating every question and reference answer. In particular, listing and absence questions are difficult to validate because the authors had to check the entire specified scope for missed instances or counterexamples.

4 Experimental Setup

4.1 Models and Agent Harnesses

We evaluate five frontier models: Claude Opus 5, Claude Opus 5.5, GPT-5.6-Sol, GPT-6-Astra, and Gemini 3.8 Flash, using `xhigh` reasoning where supported. Model-specific settings are recorded in the released repository. We compare native agents and the RLM-based MangaBench reader, which find and combine evidence across the corpus through multiple calls. Neither harness has a total token or cost budget, but model context and output limits apply per request.

Native Agents. We evaluate Claude Code, Codex, and Prime Agent. Each agent runs in an isolated container with read-only volume PDFs. They choose how to inspect pages using installed shell, Python, PDF-processing, and OCR tools, plus native subagent features where available, and retain findings in a persistent session. Network access is restricted to model inference.

MangaBench Reader (RLM). Inspired by Recursive Language Models (RLMs; Zhang et al., 2025), our reader uses a root agent to coordinate parallel visual readers, all using the same model and reasoning setting. Figure 2 illustrates this process: given the questions and chapter index, the root agent assigns reading tasks over page or chapter ranges. Each reader then receives a PDF slice in a fresh context and returns its findings with page citations. The root agent keeps reader reports instead of manga pages in its context without harness-side compaction, and uses them to guide further reads and revise its answer. The root agent can answer early or use up to 16 tool-use rounds, followed by a final answer request with tools disabled. Appendix D.2 describes the PDF slicing limits, per-call output limits, and retry policy for the RLM reader.

4.2 Evaluation Protocol

Given the cost of evaluating thousands of pages, native agents receive all selected questions for a series in one shared session, following NovelQA (Wang et al., 2025a). For model–RLM configurations, we group questions into batches of up to 20. Agents may reuse findings across questions within a session, but findings are not carried between sessions. Agents receive question IDs, text, and answer types. Reference answers and private evidence annotations remain inaccessible.

Agents inspect read-only volume PDFs. Native agents can access the full corpus supplied for each series, while RLM access follows the volume range configured for each batch. These comparisons therefore include differences in batching and available context. We do not add separate distractor pages, since the available volumes already contain pages outside each question's supporting evidence. When a question asks what was known by a specific chapter, the answer must exclude information learned later, including when later volumes are accessible. Reported resource usage includes failed attempts and retries. Appendix D provides the prompts and describes answer-format corrections, restarts, and other evaluation procedures.

We supplement evaluation with a human baseline, where annotators answered questions written by others, with full corpus access and no time limit. The authors manually scored their answers.

4.3 Metrics and Scoring

Metrics. Our primary metric is accuracy, with each question weighted equally. Missing or unidentifiable answers and execution failures count as incorrect. No execution failures occurred in the reported evaluation runs, so we omit a separate execution-failure breakdown. We further break down accuracy by task type and answer format to identify where models perform well or struggle.

Scoring. Each question receives either full credit or none. Boolean, multiple-choice, and integer answers must give the correct value or selection; text answers may use equivalent wording for the same answer. Lists must contain every reference item and no extra items, but ordering and duplicates do not matter. Answers for rankings and sequences must preserve the reference ordering. Page citations affect scoring only when the question explicitly asks for them; otherwise, they only support inspection of the answer. Appendix E details the answer parsing and initial deterministic checks.

Exact matching may reject correct, free-form text answers expressed differently from the reference answer. We therefore use an LLM judge (GPT-6-Astra, reasoning effort `high`) to apply scoring rules while accepting equivalent wording. The judge evaluates every completed answer and decides the final score. If judging fails, we retain the deterministic score. The authors reviewed and agreed with the judge on 98.0% of 954 decisions. Reported scores use the judge's decisions, including disagreements with the authors. Appendix E.1 provides the judge prompt and procedure.

5 Results

5.1 Main Results

Table 3 exhibits a large gap between model and human performance. GPT-6-Astra with Codex achieves the highest overall accuracy at 62.6%, followed by Claude Opus 5.5 with Claude Code at 61.4%. The strongest configuration remains 21.4 percentage points below the human accuracy of 84.0%. Every configuration achieves its highest accuracy on single-hop questions with the best result reaching 81.9%. The largest gap to humans is in cross-scene aggregation: the best model accuracy is 47.4%, compared with 80.1% for humans. Gathering and combining evidence across scenes therefore remains a major challenge even for frontier models.

The table also shows that harness performance depends on the model. The RLM reader exceeds Prime Agent in overall accuracy by 14.7 percentage points for Gemini 3.8 Flash and 5.6 points for GPT-5.6-Sol. GPT-6-Astra performs best with Codex, exceeding its RLM accuracy by 2.7 points, while both Claude models perform better with Claude Code than with Prime Agent. These comparisons describe complete model–harness configurations, including differences in system prompts, tool interfaces, and batching. The observed performance differences thus cannot be attributed to the reading architecture alone.

Model	Harness	Accuracy (%)			
		Overall	Single-hop	Cross-scene aggregation	Multi-hop
Claude Opus 5	Claude Code	51.0	65.6	44.3	45.5
	Prime Agent	44.1	57.1	34.9	42.3
	RLM	—	—	—	—
Claude Opus 5.5	Claude Code	61.4	**81.9**	**47.4**	58.0
	Prime Agent	51.4	68.1	39.1	49.3
	RLM	—	—	—	—
Gemini 3.8 Flash	Prime Agent	33.5	48.2	30.9	24.1
	RLM	48.2	71.3	39.4	37.7
GPT-5.6-Sol	Codex	47.9	68.8	41.9	36.5
	Prime Agent	47.6	70.6	37.6	38.3
	RLM	53.2	69.9	43.7	48.7
GPT-6-Astra	Codex	**62.6**	80.9	**47.4**	**62.0**
	Prime Agent	60.2	79.4	46.8	57.1
	RLM	59.9	79.4	47.1	55.9
Human	—	**84.0**	**89.4**	**80.1**	**83.2**

Table 3: Overall performance on MangaBench. The comparison covers the same 954 questions for each configuration. Bold values indicate highest accuracy for a specific question family.

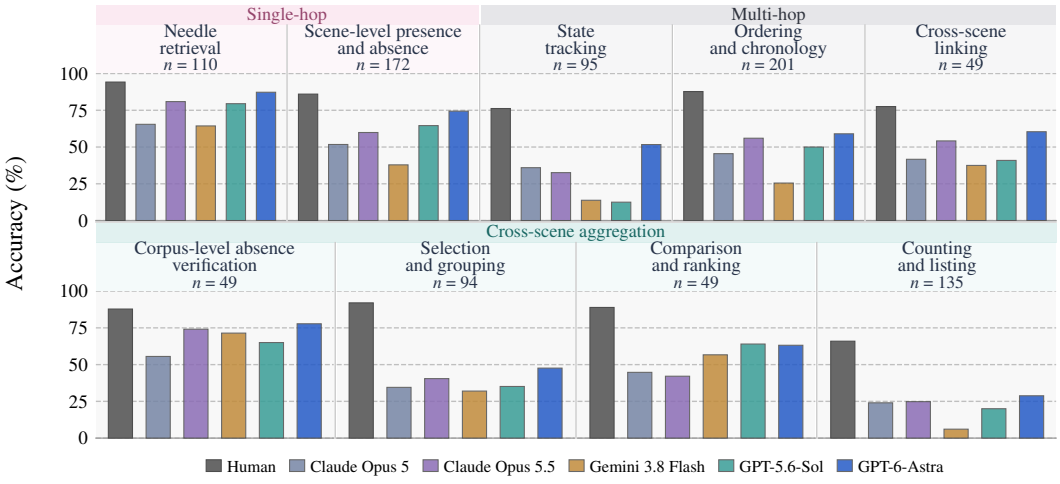


Figure 3: Accuracy with Prime Agent by task type. n denotes the number of questions per task.

5.2 Results by Question Type

Figure 3 compares all five models using Prime Agent, the only harness supporting all evaluated models. A consistent weakness across models is assembling complete answers from distributed observations: counting and listing trail needle retrieval by 42.4–58.4 percentage points across the five models. Even the strongest model, GPT-6-Astra, reaches 87.3% on retrieval but only 28.9% on counting and listing. The largest gap to humans occurs in selection and grouping, where the best model reaches 46.8% against 92.6% for humans, a 45.8-point difference. **These results show that strong retrieval performance does not ensure reliable enumeration or selection across scenes.**

State tracking presents a distinct weakness within multi-hop reasoning. Every model scores lower on state tracking than on either ordering and chronology or cross-scene linking. For GPT-6-Astra, these accuracies are 51.6%, 58.7%, and 61.2%, respectively. Broad multi-hop averages therefore obscure the additional difficulty of tracking changes and persistence in an entity's state. These patterns motivate separate tests of evidence completeness and state tracking alongside retrieval.

Condition	Accuracy (%)				Est. cost
	Overall	Single-hop	Cross-scene aggregation	Multi-hop	(USD)
Full input	**58.5**	65.1	**54.5**	**51.9**	$23.18 + 0.00$
Text-only	53.9	**66.3**	41.8	46.2	$11.56 + 36.29$
Vision-only	51.8	60.5	47.3	42.3	$11.65 + 44.71$
Closed-book	54.4	64.0	52.7	40.4	$0.90 + 0.00$

Table 4: *Jujutsu Kaisen* input conditions with GPT-5.6-Sol on 193 questions: 86 single-hop, 55 aggregation, and 52 multi-hop. The first three rows use Codex, while closed-book makes a direct API call. All use GPT-6-Astra (`high`) binary judgments. Costs show main-agent usage + additional metered usage, excluding grading.

5.3 Input Ablations

We vary the available input to examine how textual and visual evidence contribute to performance, and how much can be answered without the manga. We compare four conditions using GPT-5.6-Sol with `xhigh` reasoning effort on the same 193 *Jujutsu Kaisen* questions. Full input provides the original pages. Text-only provides OCR-extracted text with page references, including dialogue, narration, signs, and sound effects, but no images or added captions. Vision-only provides pages at their original dimensions with detected text removed by masking and inpainting. Closed-book provides only the questions and answer-format instructions, testing what the model can answer from prior knowledge and question cues. The first three conditions use shared Codex sessions; closed-book uses one direct API batch without corpus access or tools.

Full input achieves the highest overall accuracy at 58.5%, compared with 53.9% for text-only and 51.8% for vision-only (Table 4). Its advantage is concentrated in aggregation and multi-hop questions, suggesting that text and images provide complementary evidence for reasoning across scenes. By contrast, single-hop accuracy is similar with text-only and full input (66.3% versus 65.1%).

Closed-book nevertheless reaches 54.4% overall and approaches full input on single-hop and aggregation questions. The largest gap is on multi-hop questions (40.4% versus 51.9%). **Substantial accuracy is therefore possible without corpus access, so correct answers alone do not establish visual grounding.** Extraction and masking errors, together with differences in the closed-book harness and prompt, limit interpretation of these comparisons as isolated effects of input modality.

5.4 Downsampled Images

We test whether an entire manga can fit in a single model context. We arrange all 2,781 *Blue Period* pages from Volumes 0–15 into 464 contact sheets, each containing up to six pages in a three-column, two-row grid. We preserve page order, reduce each sheet to 448×448 or 672×672 pixels, and apply JPEG compression. All 102 questions follow the images in one request, without reading tools. The resulting inputs use approximately 155,000 and 291,000 tokens, respectively. This preserves coverage of the whole series, but makes small text and visual details harder to read.

With either compressed input, the model answers more than half the questions correctly (53.9–55.9%), with far fewer tokens and shorter runtime than RLM (Table 5). All configurations remain below 50% on aggregation and multi-hop. Larger sheets yield only two additional correct answers for about 70% more total tokens. JPEG quality, corpus access, reading procedure, and question/reference versions also differ, so this comparison does not isolate the effect of resolution.

5.5 Failure Analysis

We manually audit 139 questions selected from Codex failures across four series, alongside corresponding RLM responses. We classify failures into four categories: evidence not inspected, occurrences missed or misread, findings omitted, and incorrect entity or time associations. Cases with insufficient trace evidence were left unresolved.

Configuration	Accuracy (%)				Runtime	Total tokens
	Overall	Single-hop	Cross-scene aggregation	Multi-hop		
Full context, 448 × 448, Q30	53.9	71.4	40.4	16.7	7m 40s	181,869
Full context, 672 × 672, Q8	55.9	71.4	42.6	**33.3**	5m 09s	309,899
RLM, original pages	**67.6**	**91.8**	**48.9**	16.7	37m 30s	8,378,370

Table 5: Blue Period downsampling with GPT-5.6-Sol (`high`) on 102 questions: 49 single-hop, 47 aggregation, and 6 multi-hop. Scores use saved binary judgments. Q denotes JPEG quality. Runtime and tokens cover answer generation.

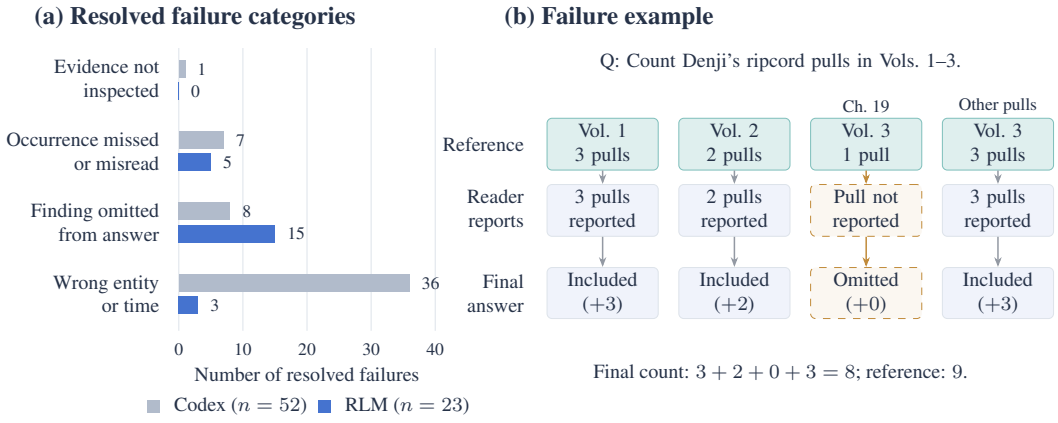


Figure 4: Failure analysis. (a) Counts of resolved failures for GPT-6-Astra using Codex ($n = 52$) and the RLM reader ($n = 23$) across four series. Each failed answer receives one category; unresolved cases and correct answers are excluded. (b) A Gemini 3.8 Flash RLM trace for counting Denji's ripcord pulls in *Chainsaw Man*, Volumes 1–3. The reader omits the Chapter 19 pull, so the final answer correctly sums the reported events but gives eight instead of the reference count of nine.

Among failures with an identifiable cause, Codex most often linked evidence to the wrong entity or time (36/52), while RLM most often omitted recovered information from its final answer (15/23), as seen in Figure 4(a). Because the questions were selected from Codex failures, they do not establish which harness performs better overall.

These examples show that errors can arise both when reading the pages and when writing the final answer. In Figure 4(b), Gemini's reader misses the Chapter 19 ripcord pull. The final answer adds the reported events correctly, but counts eight instead of nine. In a separate *Jujutsu Kaisen* ranking question, Codex correctly notes that Hazenoki and Yuta both have 35 points, yet its final ranking does not show that they are tied. The model found the evidence but failed to use it in its answer.

These failures show that reliable answers require both accurate page reading and consistent use of found evidence. Recording events with entities, timing, and source pages could help models check for missing information. Improving how models preserve and use evidence across scenes is therefore a promising direction for addressing failures in counting, listing, and state tracking.

6 Conclusion

We introduce MangaBench, a benchmark for evaluating multimodal models' ability to locate and combine visual and textual evidence across long manga narratives. Its corpus averages 3,768.4 pages per series. Evaluating frontier models with native agents and a specialized reader, we find that they struggle with counting, listing, and state tracking. Future work should examine whether larger context windows and improved memory mechanisms for gathering and combining evidence can overcome these limitations.

7 AUTHOR STATEMENTS

AI USE STATEMENT

In this work, we used generative AI tools to assist with experimental design and harness implementation, and to judge model answers for the reported scores (see Section 4.3). We also used generative AI tools to generate a synthetic manga, *I Caused a Latte Rebellion*, to demonstrate the evaluation procedure without users needing to supply their own scans. We have not used generative AI tools for data annotation or results interpretation. Cleaning and reformatting datasets, developing theoretical models or conceptual frameworks, proposing or refining hypotheses, formulating mathematical claims or writing proofs, translation and qualitative data analysis are not applicable to this work. Additionally, we used generative AI tools to edit the paper to improve readability and to create graphs. The core research idea was developed by the authors. We have reviewed all AI-assisted work: the harness code was reviewed and tested for correctness by the authors, all text edits and graphs were validated by authors to ensure technical content remains unchanged, and all AI-generated suggestions were carefully judged before adoption. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

We are dedicated to ensuring that MangaBench serves exclusively non-commercial academic endeavors. The benchmark uses 30,147 pages from eight manga titles, all of which are protected by copyright. We purchased all volumes and scanned pages ourselves, and we use the scans solely for non-commercial research under the text and data mining exception in Section 29A of the UK Copyright, Designs and Patents Act 1988. As such, we affirm that we will not redistribute the scans. We release only our question–answer annotations together with page identifiers, so researchers can assemble the evaluation corpus from their scans of lawfully obtained copies. Annotations for the held-out split are kept private to avoid contamination. We plan to host a public leaderboard, which accepts model configurations and containerized agents for evaluation on our infrastructure, with inference routed through supported providers and reference answers kept private.

All model inference, including direct API calls, Claude Code, Codex, and Prime Agent requests routed through our gateway, was performed under enterprise agreements with Anthropic, OpenAI, and Google, under terms that prohibit use of submitted content for model training; providers retain inputs only temporarily (up to 30 days for OpenAI and Anthropic, and 55 days for Google) for abuse monitoring. Page images are transmitted solely to obtain model outputs and are not shared with any other party. For native-agent evaluations, we use Claude Code and Codex through enterprise accounts. Submitted content is covered by the providers' commercial data protections, including no model training by default. Retention and deletion follow the applicable enterprise agreements and account settings (Anthropic, 2026a; OpenAI, 2026a;b).

Annotators had read their assigned series before joining the project. They were paid £14 per hour, above the UK National Living Wage, for writing questions and for inter-annotator review.

REPRODUCIBILITY STATEMENT

We provide the evaluation harness and the question–answer annotations for the public split as anonymized supplementary material. The underlying manga are copyrighted and we do not release page images; instead, we provide page identifiers and a script that builds the benchmark from purchased copies of the listed volumes, including details on the purchased copies' publisher and edition (Appendix A). We also provide a small synthetic manga, *I Caused a Latte Rebellion*, with two volumes, three example questions and public reference answers. Users can inspect recorded runs, including a cross-scene height-ranking error, or rerun the workflow using their own API key. and can reproduce the cross-scene aggregation failure. The annotation procedure is described in Section 3.2. All prompts are listed in Appendix D.1. The final API access date is September 25 2026, with model and agent identifiers described in Section 4.1. All model–harness pairs and reasoning settings are described in Appendix D.2. We plan to provide hosted evaluation on the held-out split through a public leaderboard. Submitted agents will run in network-isolated containers with read-only access to the scans, with outbound traffic limited to inference requests through our gateway, and only aggregate scores will be published.

REFERENCES

Kiyoharu Aizawa, Azuma Fujimoto, Atsushi Otsubo, Toru Ogawa, Yusuke Matsui, Koki Tsubota, and Hikaru Ikuta. Building a manga dataset "Manga109" with annotations for multimedia applications. *IEEE MultiMedia*, 27(2):8–18, 2020. doi: 10.1109/mmul.2020.2987895.

Anthropic. Data usage – Claude Code documentation, 2026a. URL https://code.claude.com/docs/en/data-usage. Accessed 22 September 2026.

Anthropic. What's new in Claude Opus 5 – Claude Platform Docs, 2026b. URL https://platform.claude.com/docs/en/models/opus-5/whats-new-opus-5. Accessed 22 September 2026.

Anthropic. Claude Opus 5.5 – Claude Platform Docs, 2026c. URL https://platform.claude.com/docs/en/models/opus-5-5/overview. Accessed 22 September 2026.

Anthropic. PDF support – Claude Platform Docs, 2026d. URL https://platform.claude.com/docs/en/build-with-claude/pdf-support. Accessed 22 September 2026.

Anthropic. Vision – Claude Platform Docs, 2026e. URL https://platform.claude.com/docs/en/build-with-claude/vision. Accessed 22 September 2026.

Jeonghun Baek, Kazuki Egashira, Shota Onohara, Atsuyuki Miyai, Yuki Imajuku, Hikaru Ikuta, and Kiyoharu Aizawa. MangaVQA and MangaLMM: A benchmark and specialized model for multimodal manga understanding. In *Findings of the Association for Computational Linguistics: EACL 2026*, pp. 5349–5370, 2026. doi: 10.18653/v1/2026.findings-eacl.284.

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. LongBench: A bilingual, multitask benchmark for long context understanding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3119–3137, 2024. doi: 10.18653/v1/2024.acl-long.172.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. LongBench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3639–3664, 2025. doi: 10.18653/v1/2025.acl-long.183.

Aaditya Baranwal, Madhav Kataria, Naitik Agrawal, Yogesh S. Rawat, and Shruti Vyas. Re:Verse - can your VLM read a manga? In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 3820–3830, 2025.

Amanda Bertsch, Adithya Pratapa, Teruko Mitamura, Graham Neubig, and Matthew R. Gormley. Oolong: Evaluating long context reasoning and aggregation capabilities. *arXiv preprint arXiv:2511.02817*, 2025.

Daniel Cores, Michael Dorkenwald, Manuel Mucientes, Cees G. M. Snoek, and Yuki M. Asano. Lost in time: A new temporal benchmark for VideoLLMs. In *36th British Machine Vision Conference (BMVC)*, 2025.

Anurag Das, Adrian Bulat, Alberto Baldrati, Ioannis Maniadis Metaxas, Bernt Schiele, Georgios Tzimiropoulos, and Brais Martinez. More images, more problems? a controlled analysis of VLM failure modes. *arXiv preprint arXiv:2601.07812*, 2026.

Guancheng Du, Yong Hu, Wenqing Wang, Yaming Yang, and Jiaheng Gao. SagaScale: A realistic, scalable, and high-quality long-context benchmark built from full-length novels. *arXiv preprint arXiv:2601.09723*, 2026.

Chaoyou Fu, Haozhi Yuan, Yuhao Dong, Yi-Fan Zhang, Yunhang Shen, Xiaoxing Hu, Xueying Li, Jinsen Su, Chengwu Long, Xiaoyao Xie, Yongkang Xie, Xiawu Zheng, Xue Yang, Haoyu Cao, Yunsheng Wu, Ziwei Liu, Xing Sun, Caifeng Shan, and Ran He. Video-MME-v2: Towards the next stage in benchmarks for comprehensive video understanding. *arXiv preprint arXiv:2604.05015*, 2026.

Google. Gemini 3.8 Flash – Gemini API documentation, 2026a. URL https://ai.google.dev/gemini-api/docs/models/gemini-3.8-flash. Accessed 22 September 2026.

Google. Media resolution – Gemini API documentation, 2026b. URL https://ai.google.dev/gemini-api/docs/media-resolution. Accessed 22 September 2026.

Google. Document understanding – Gemini API documentation, 2026c. URL https://ai.google.dev/gemini-api/docs/document-processing. Accessed 22 September 2026.

Abhay Gupta, Kevin Zhu, Vasu Sharma, Sean O'Brien, and Michael Lu. NovelHopQA: Diagnosing multi-hop reasoning failures in long narrative contexts. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 26134–26151, 2025. doi: 10.18653/v1/2025.emnlp-main.1328.

Sil Hamilton, Matthew Wilkens, and Andrew Piper. NarraBench: A comprehensive framework for narrative benchmarking. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3786–3801, 2026. doi: 10.18653/v1/2026.eacl-long.176.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. RULER: What's the real context size of your long-context language models? In *First Conference on Language Modeling*, 2024. URL https://arxiv.org/abs/2404.06654.

Hikaru Ikuta, Leslie Wöhler, and Kiyoharu Aizawa. MangaUB: A manga understanding benchmark for large multimodal models. *IEEE MultiMedia*, 2025. doi: 10.1109/MMUL.2025.3550451.

Mohit Iyyer, Varun Manjunatha, Anupam Guha, Yogarshi Vyas, Jordan Boyd-Graber, Hal Daumé III, and Larry S. Davis. The amazing mysteries of the gutter: Drawing inferences between panels in comic book narratives. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7186–7195, 2017.

Rahul Jain, Keval Doshi, Burak Uzkent, and Garin Kessler. Narrative aligned long form video question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 8765–8774, June 2026.

Greg Kamradt. Needle in a haystack—pressure testing LLMs. GitHub repository, 2023. URL https://github.com/gkamradt/needle-in-a-haystack.

Marzena Karpinska, Katherine Thai, Kyle Lo, Tanya Goyal, and Mohit Iyyer. One thousand and one pairs: A "novel" challenge for long-context language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 17048–17085, 2024. doi: 10.18653/v1/2024.emnlp-main.948.

Tomáš Kočiský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. The NarrativeQA reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328, 2018. doi: 10.1162/tacl_a_00023.

Baiqi Li, Kangyi Zhao, Ce Zhang, Chancharik Mitra, Jean de Dieu Nyandwi, and Gedas Bertasius. TimeBlind: A spatio-temporal compositionality benchmark for video LLMs. *arXiv preprint arXiv:2602.00288*, 2026.

Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. Sketch-based manga retrieval using Manga109 dataset. *Multimedia Tools and Applications*, 76(20):21811–21838, 2017. doi: 10.1007/s11042-016-4020-z.

Ali Modarressi, Hanieh Deilamsalehy, Franck Dernoncourt, Trung Bui, Ryan A. Rossi, Seunghyun Yoon, and Hinrich Schuetze. NoLiMa: Long-context evaluation beyond literal matching. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 44554–44570, 2025.

OpenAI. Authentication – Codex documentation, 2026a. URL https://learn.chatgpt.com/docs/auth. Accessed 22 September 2026.

OpenAI. ChatGPT Work admin FAQ, 2026b. URL https://learn.chatgpt.com/docs/enterprise/work-admin-faq. Accessed 22 September 2026.

OpenAI. File inputs – OpenAI API documentation, 2026c. URL https://developers.openai.com/api/docs/guides/file-inputs. Accessed 22 September 2026.

OpenAI. GPT-5.6 Sol – OpenAI API documentation, 2026d. URL https://developers.openai.com/api/docs/models/gpt-5.6-sol. Accessed 22 September 2026.

OpenAI. GPT-6 Astra – OpenAI API documentation, 2026e. URL https://developers.openai.com/api/docs/models/gpt-6-astra. Accessed 22 September 2026.

OpenAI. Images and vision – OpenAI API documentation, 2026f. URL https://developers.openai.com/api/docs/guides/images-vision. Accessed 22 September 2026.

Galann Pennec, Zhengyuan Liu, Nicholas Asher, Philippe Muller, and Nancy F. Chen. SagaQA: A multi-hop reasoning benchmark for long-form narrative understanding in TV series, 2026. URL https://arxiv.org/abs/2606.03301.

Dingjie Song, Shunian Chen, Guiming Hardy Chen, Fei Yu, Xiang Wan, and Benyou Wang. MileBench: Benchmarking MLLMs in long context. In *Proceedings of the First Conference on Language Modeling*, 2024.

Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. MovieQA: Understanding stories in movies through question-answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4631–4640, 2016.

Emanuele Vivoli, Marco Bertini, and Dimosthenis Karatzas. CoMix: A comprehensive benchmark for multi-task comic understanding. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/fe79898dcf078ec54b6feeea10ebb751-Abstract-Datasets_and_Benchmarks_Track.html.

Cunxiang Wang, Ruoxi Ning, Boqi Pan, Tonghui Wu, Qipeng Guo, Cheng Deng, Guangsheng Bao, Xiangkun Hu, Zheng Zhang, Qian Wang, and Yue Zhang. NovelQA: Benchmarking question answering on documents exceeding 200K tokens. In *International Conference on Learning Representations*, 2025a.

Fei Wang, Xingyu Fu, James Y. Huang, Zekun Li, Qin Liu, Xiaogeng Liu, Mingyu Derek Ma, Nan Xu, Wenxuan Zhou, Kai Zhang, Tianyi Yan, Wenjie Mo, Hsiang-Hui Liu, Pan Lu, Chunyuan Li, Chaowei Xiao, Kai-Wei Chang, Dan Roth, Sheng Zhang, Hoifung Poon, and Muhao Chen. MuirBench: A comprehensive benchmark for robust multi-image understanding. In *International Conference on Learning Representations*, 2025b.

Weiyun Wang, Shuibo Zhang, Yiming Ren, Yuchen Duan, Tiantong Li, Shuo Liu, Mengkang Hu, Zhe Chen, Kaipeng Zhang, Lewei Lu, Xizhou Zhu, Ping Luo, Yu Qiao, Jifeng Dai, Wenqi Shao, and Wenhai Wang. Needle in a multimodal haystack. In *Advances in Neural Information Processing Systems*, volume 37, pp. 20540–20565, 2024. doi: 10.52202/079017-0649.

Ziyang Wang, Yue Zhang, Shoubin Yu, Ce Zhang, Zengqi Zhao, Jaehong Yoon, Hyunji Lee, Gedas Bertasius, and Mohit Bansal. EgoMemReason: A memory-driven reasoning benchmark for long-horizon egocentric video understanding. In *Third Conference on Language Modeling*, 2026. URL https://arxiv.org/abs/2605.09874.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. LongVideoBench: A benchmark for long-context interleaved video-language understanding. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2024.

Zhe Xu, Jiasheng Ye, Xiaoran Liu, Xiangyang Liu, Tianxiang Sun, Zhigeng Liu, Qipeng Guo, Linlin Li, Qun Liu, Xuanjing Huang, and Xipeng Qiu. DetectiveQA: Evaluating long-context reasoning on detective novels. *arXiv preprint arXiv:2409.02465*, 2024.

Alex L. Zhang, Tim Kraska, and Omar Khattab. Recursive language models, 2025. URL https://arxiv.org/abs/2512.24601.

A Corpus Documentation

Table 6 summarises the source editions and volume coverage. All series use purchased collected editions.

Manga Title	Volumes	Language	Publisher and edition
Blue Giant	1–10	English	Seven Seas Entertainment; five 2-in-1 omnibuses
Blue Period	1–17	English	Kodansha Comics; English paperback edition
Chainsaw Man	1–22	English	VIZ Media; Shonen Jump manga edition
Golden Kamuy	1–31	English	VIZ Media; VIZ Signature edition
Jujutsu Kaisen	0–30	English	VIZ Media; Shonen Jump manga edition
Land of the Lustrous	1–13	English	Kodansha Comics; English paperback edition
Manchuria Opium Squad	1–14	Japanese	Kodansha; tankōbon edition
Steel Ball Run	1–24	Japanese	Shueisha; Jump Comics collected edition, monochrome

Table 6: Source editions and volume coverage of the corpus.

We scan every page of each included volume, including title pages, and exclude no material. Page counts include all scanned pages. Two-page spreads are scanned as two separate pages and count as two pages. Pages retain their original order within each volume. Each 1120×1600 scan occupies one PDF page, and the pixel dimensions refer to the embedded scan. Page references use the printed page numbers in the source edition. For pages without printed numbers, we count from the first page of the volume, starting at one.

B Question and Evidence Distributions

Table 7 reports question counts and percentages by task type and answer format in the current question set. Task types follow Figure 1. Each question is counted once under its primary task type and once under its answer format; percentages use all 954 questions as their denominator. Multiple-choice questions are combined because the recorded answer format does not distinguish single-answer from multiple-answer items.

Task type or answer format	Count	Share (%)
Task type		
Needle retrieval	110	11.5
Scene-level presence and absence	172	18.0
Counting and listing	135	14.2
Selection and grouping	94	9.9
Comparison and ranking	49	5.1
Corpus-level absence verification	49	5.1
State tracking	95	10.0
Ordering and chronology	201	21.1
Cross-scene linking	49	5.1
Answer format		
Multiple-choice	91	9.5
True/false	34	3.6
Integer	66	6.9
Exact-answer	482	50.5
List	281	29.5

Table 7: Question distribution by task type and answer format. Percentages are rounded to one decimal place.

Table 8 reports the evidence distribution separately for single-hop, cross-scene aggregation, and multi-hop questions with assigned task labels. Evidence pages are the number of distinct pages cited in each question's annotations. Chapter span counts indexed chapters from the earliest to latest cited chapter, including both endpoints. Page gap counts page-to-page steps from the first to the last cited page in corpus order. We use corpus indices to resolve chapter order and page offsets where the coordinate mapping is supported.

Statistic	Single-hop	Cross-scene aggregation	Multi-hop
Evidence pages (mean)	1.4	11.5	4.77
Evidence pages (median)	1 [1, 1]	4 [3, 7.5]	3 [2, 4.5]
Chapter span (median)	1 [1, 1]	44.5 [16, 90.5]	59.5 [28.5, 144.5]
Page gap (median)	0 [0, 0]	944 [249, 2257]	542 [102.5, 1596]
Multiple chapters (%)	1.4	97.5	100
Multiple volumes (%)	3.5	82.5	74.1

Table 8: Evidence distribution for 282 single-hop, 327 cross-scene aggregation, and 345 multi-hop questions. Brackets give [Q1, Q3] for median rows.

C PDF Input Requirements and Context Limits

Table 9 estimates how many visual tokens would be needed to read all $2{,}034$ pages of *Blue Giant*, our smallest corpus, in one request. Each PDF page contains a 1120×1600 scan. Even the lowest estimate, about 1.14 million tokens, exceeds the listed context windows of around one million tokens. We take these context limits from the official specifications for Gemini (Google, 2026a), Claude (Anthropic, 2026b;c), and GPT (OpenAI, 2026d;e).

Gemini 3's default PDF setting allocates approximately 560 visual tokens per page (Google, 2026b). For the other models, we assume that each page is rendered at the scan resolution, without additional crops or downscaling. Note that these are estimates and not measured API token counts. We then count the cells in the model's visual grid, rounding up at the page edges:

$$T_{\text{page}} \approx \left\lceil \frac{1120}{s} \right\rceil \cdot \left\lceil \frac{1600}{s} \right\rceil, \qquad T_{\text{corpus}} \approx 2034 \cdot T_{\text{page}},$$

Model	Visual tokens/page	Corpus visual tokens	Context limit
Claude Opus 5	$\approx 2{,}320^{\dagger}$	$\approx 4{,}718{,}880$	1,000,000
Claude Opus 5.5	$\approx 2{,}320^{\dagger}$	$\approx 4{,}718{,}880$	1,000,000
Gemini 3.8 Flash	≈ 560	$\approx 1{,}139{,}040$	1,048,576
GPT-5.6-Sol	$\approx 1{,}750^{\dagger}$	$\approx 3{,}559{,}500$	1,050,000
GPT-6-Astra	$\approx 1{,}750^{\dagger}$	$\approx 3{,}559{,}500$	1,050,000

Table 9: Estimated visual tokens for *Blue Giant*'s 2,034 PDF pages and published context limits. † assumes pages are rendered at 1120×1600 without additional crops or downscaling. Gemini's limit covers input tokens only. Estimates are based on information available on 22 September 2026.

Here, s is the width of each grid cell in pixels. Claude Opus 5 and 5.5 use $s = 28$ (Anthropic, 2026e), while GPT at high detail uses $s = 32$ (OpenAI, 2026f). This gives 2,320 and 1,750 visual tokens per page, respectively, before GPT's billing multiplier. The estimates exclude extracted text, prompts, and request overhead. Actual token counts depend on how the provider processes the PDF.

Even if all 2,034 pages fit within the token budget, Claude and Gemini's native APIs would still prevent us from sending them in one request. Gemini accepts at most 1,000 PDF pages per request (Google, 2026c); Claude accepts 600 with a one-million-token context window (Anthropic, 2026d). File size is a separate constraint: Gemini limits PDFs to 50 MB, Claude limits request payloads to 32 MB, and OpenAI limits the combined input files to 50 MB (Google, 2026c; Anthropic, 2026d; OpenAI, 2026c). The listed GPT models also cap input at 922,000 tokens within their larger context window. The MangaBench reader handles these constraints by spreading page reads across multiple requests, subject to the configured page and encoded file-size limits.

D Evaluation Details

D.1 Prompts

The following blocks reproduce the native-agent task prompt and the RLM prompt components. Series names, questions, answer-format instructions, chapter indexes, and page mappings are supplied separately where required. Native agents keep their own system prompts and tool interfaces. Reference answers and private evidence annotations are never included. Typographic emphasis is added for readability.

Native-Agent Task Prompt. The same task prompt is given to each native agent. The question file contains only question IDs, question text, and answer types.

You are running the MangaBench baseline evaluation.

Read every question in `/questions/questions.json`. Use the full-resolution volume PDFs under `/manga` as the source of truth. You have **one persistent session for the entire batch** and may use any installed tool or native subagent feature. Network access is intentionally unavailable except for model inference.

Do not stop until **every question has exactly one result**. Write `/output/results.json` as a JSON array and no other format. Each element must contain exactly:

```
{"id":"the question id",
  "answer":"the concise final answer",
  "evidence":["human-readable volume/page citation"]}
```

Requirements:
- Preserve each question id exactly.
- Return every requested id once, with no extra ids.
- answer must be a non-empty string. For multiple choice, use only the requested letter or letters. For lists, separate items with semicolons. For counts, give the integer. For yes/no, answer yes or no.
- evidence must be an array of strings. Cite the volume and PDF page for the panels used. Use an empty array only when the answer is that the event is absent and no positive page proves it.

- `results.json` must contain valid JSON with no Markdown fences.
- **Do not search for an answer key.** No answer key exists in this environment.

Before finishing, parse `results.json` yourself and verify its ids against `/questions/questions.json`.

RLM Root Agent: Base Instructions. The series name replaces {`series`} below. In shared sessions, the batch instructions that follow replace the single-question output format. The root agent also receives the chapter index and the question batch.

You answer a hard question about the manga "{series}" by ORCHESTRATING vision sub-calls.

You cannot see any page yourself. `research(vol, ch, pages, look_for)` examines targeted pages or one chapter. `research_range(start_vol, end_vol, look_for)` performs one exhaustive pass over every page in an inclusive volume range, splitting only when request limits require it. Decompose the question, read the TEXT reports, and iterate until you can answer. **Tell each call precisely what to look for.**

Each vision leaf is returned as `[Checked: ...]` followed by `MATCH`, `UNCERTAIN`, or `NO MATCHES`. `NO MATCHES` applies only to the pages in the immediately preceding `[Checked: ...]` line. **Combine all leaf blocks before deciding that something is absent from a chapter or range.**

You have a chapter index (volumes → chapters → page counts and 1-based volume-page ranges) to decide where to look. The research tool's page indices are 0-based within a chapter. Questions and gold data may cite 1-based volume pages; use the displayed ranges to convert them. Be efficient but thorough. For a genuinely exhaustive "every", "never", or "across the series" question, issue ONE `research_range` call with a single complete predicate rather than separate whole-chapter calls. Include all candidates/options in that predicate so the corpus is not rescanned once per hypothesis. Use targeted `research` calls to verify uncertain pages afterwards.

Answer DIRECTLY — no preamble, no restating the question, no hedging. But be **SPECIFIC and COMPLETE**: include every relevant detail and all qualifying instances you found (for "list every X", list them all). Cut padding, not substance. Then a line "Evidence:" citing the volumes/chapters/pages you relied on.

Your response MUST end with a final line in the EXACT format specified in the ANSWER FORMAT block of the question. That final "`ANSWER:`" line is the only thing graded — put the answer there in the canonical form and nothing else (no markdown, no explanation). Reasoning and Evidence go ABOVE it.

RLM Root: Batch Instructions. These instructions are appended to the base prompt when questions share one session. Each question is accompanied by its required answer format.

You are answering a batch of questions in **ONE shared session**. Plan the initial reading across all questions together; omit `question_ids` on the first turn to check the whole batch. Reuse the resulting evidence across answers. Combine all checked blocks before deciding what remains unresolved. A leaf saying it cannot answer the whole question from its small excerpt does not itself justify rereading that chapter.

On follow-ups, supply `question_ids` for only the unresolved questions. Use explicit research pages (or `research_joint`) and state the specific ambiguity to verify. Whole-chapter and volume-range scans overlapping previously read pages are rejected; use the existing findings instead of restarting a scan. For incomplete coverage use `retry_research(call_id)`, which restores the original task and reads only its missing pages. Do not rewrite its predicate or reread successful pages to repair a timeout. Findings retain question IDs and page citations; every answer may reuse relevant shared evidence.

Return only a JSON object with this structure:

```
{"answers": [{
  "id": "exact question ID",
  "model_answer": "reasoning and evidence followed by the question's
ANSWER: line"
}]}
```

Include every supplied question ID exactly once, no other IDs. Each `model_answer` is a string and must end with that question's requested ANSWER: format. Do not put a single ANSWER: line outside the JSON. **This batch output format replaces the single-question final response format.**

Visual-Reader Instructions. Each visual reader receives a PDF slice, a mapping from its pages to source page references, and a reading task assigned by the root agent. The following instructions specify how it returns findings.

Return only concise evidence lines in one of these forms:
MATCH | <Vol/Ch/chapter-page citation(s)> | <directly relevant visible evidence>
UNCERTAIN | <Vol/Ch/chapter-page citation(s)> | <relevant ambiguity and why it is unresolved>

If no supplied page contains relevant evidence, return exactly:

NO MATCHES

Use one line per distinct finding. Do not restate the task, summarize pages with no matches, list irrelevant near-matches, or give a final answer based on pages outside this PDF. Do not use Markdown.

D.2 Evaluation Settings

Native agents use one shared session per series. RLM session boundaries and question order depend on the batch configuration. Runs grouped by volume limit use batches of up to 20 questions and restrict access to the corpus through that volume. Findings are not carried between sessions. We evaluate supported model–harness combinations, using the highest supported reasoning setting other than `max`. Table 10 records the compatibility checks completed; table 11 lists the RLM settings used for the reported results.

Model	Native agents			RLM
	Claude Code	Codex	Prime Agent	Reader
Claude Opus 5	✓	X	✓	—
Claude Opus 5.5	✓	X	✓	—
Gemini 3.8 Flash	X	X	✓	✓
GPT-5.6-Sol	X	✓	✓	✓
GPT-6-Astra	X	✓	✓	✓

Table 10: Model–harness compatibility matrix. ✓: included in our evaluation; X: model incompatible with harness; —: not evaluated.

Setting	Used RLM default
Root tool-use rounds	16, followed by a final answer request without tools
Pages per visual-reader request	At most 100, subject to the file-size limit
Encoded PDF size per request	At most 16 MiB after base64 encoding
Visual-reader output allowance	25,000 tokens per request
Root output allowance in shared sessions	32,768 tokens, increased to at most 65,536 after output exhaustion
Transient request failures	At most two retries after the initial attempt

Table 11: Default RLM settings in the evaluation code, used for our reported results. Output allowances include reasoning tokens where the API counts them toward the output limit.

For the main RLM evaluation, PDF slices preserve the original page resolution; the provider controls subsequent rendering and preprocessing. Larger reading tasks are split into requests satisfying the page and file-size limits. If the root agent exhausts its output allowance, it retries with a larger

allowance and the same completed reading history. These output-limit retries are separate from retries for transient connection or provider errors.

For native agents, invalid or incomplete answer files trigger one continuation of the same session to correct the output. We use Claude Code 2.1.215, Codex 0.151.0, and Prime Agent 0.8.1. Codex allows up to eight request retries and eight stream retries. Prime is configured with a 1,000,000-token context limit and a 128,000-token output limit. Other request limits and retry settings retain the defaults of the pinned agent versions.

Interrupted runs are not automatically restarted. If an RLM run is resumed, fully recorded sessions are retained and any incomplete batch is rerun from the beginning. Answers from different attempts are not combined; only the latest recorded batch attempt is scored. Optional answer-only recovery is not part of the default protocol.

E SCORING RULES

Each answer receives 1 if correct and 0 otherwise, with no partial credit. Table 12 defines correctness for each answer format. We extract the final `ANSWER:` line from reader responses or use the native agent's structured answer field, then normalize formatting. For text and ranking answers without an `ANSWER:` line, we compare the whole response.

Answer format	**Requirement for full credit**
True/false	Matches the reference value. Yes/no is the same as true/false.
Single-answer MCQ	Selects the correct option.
Multiple-answer MCQ	Selects all correct options and no others.
Integer	Gives the reference integer.
List	Contains every reference item and no extras, allowing equivalent wording. Order and duplicates do not matter.
Text	Matches the reference in meaning, allowing equivalent wording.
Rankings and sequences	Gives the same answer while preserving the reference ordering.

Table 12: Binary scoring rules by answer format.

A question counts as completed when an answer is returned, even if it cannot be parsed. Missing answers, unsupported inputs, and execution failures do not count as completed and receive zero credit. Completed responses are assessed by the judge as described below. Overall accuracy includes every question, while accuracy among completed responses includes only those with returned answers. Accuracy among completed responses is reported as unavailable if no responses are completed.

E.1 LLM JUDGE AND HUMAN REVIEW

LLM Judge. The LLM judge receives the question, answer type, reference answer, submitted answer, and the initial deterministic assessment. It evaluates every completed answer and can overturn the original assessment. For both harnesses, we supply the final answer: the last `ANSWER:` line for the RLM reader, or the structured answer field for native agents. Separate reasoning and evidence are omitted. If a reader response has no `ANSWER:` line, we supply the response so the judge can identify a clearly stated final answer, and record this fallback. An explicit empty answer remains empty. Equivalent wording is accepted, but the final answer must satisfy the scoring rules in Table 12 to receive full credit.

An answer that fails deterministic parsing can still be judged correct if its meaning is clear. Responses without a definite answer receive zero. The judge must return a correct or incorrect decision; it cannot abstain. If judging fails, we use the deterministic score instead without changing whether the original response counts as completed. Judge requests allow up to 2,000 output tokens, with up to two retries for transient provider errors. Invalid or incomplete judgments use the same fallback without further judging attempts. We use the following prompt, with typographic emphasis added for readability:

You score a benchmark answer using binary credit. **Your decision is the final score.**

Compare only the supplied question, answer type, reference answer, model response, and deterministic mark. Treat the reference as the grading target; do not use outside knowledge to replace or challenge it. **Assess the final answer, not the quality of the reasoning leading to it.** Do not use surrounding reasoning or evidence to repair an incorrect or missing final answer.

Award full credit only when the response gives the same answer as the reference. Accept equivalent wording, aliases that unambiguously identify the same entity, and harmless formatting differences. Boolean and integer answers must give the reference value. Multiple-choice answers must select exactly the reference option or option set. Lists must contain every reference item and no extra items; ignore order and duplicates. Rankings and sequences must express the reference ordering, including when an equivalent ordering is written in the opposite direction. Evaluate page citations only when the question requests them. Do not accept contradictory final answers. **NO partial credit:** an incomplete answer, a missing or extra list item, a partially correct option set, or a near-miss integer receives zero. If the response gives no definite answer, assign `WRONG` with zero credit.

Return `assessment`, `suggested_verdict`, `recommended_score`, and `reason` as JSON. You must choose `CORRECT` with `recommended_score 1` or `WRONG` with `recommended_score 0`; **do not abstain or return `null`.** Mark `assessment` as `consistent` only when both the verdict and credit match the deterministic mark; otherwise mark it `inconsistent`. Keep the reason short and specific.

Human Review. The authors manually checked every model answer against the reference using the same scoring rules as the judge. The human and judge decisions agreed on 935 out of 954 (98.0%) of answers, with all disagreements being false negatives where the judge rejected answers that the authors considered equivalent to the reference. Reported scores retain the judge's decisions, including disagreements with the authors.

F Additional Results by Task Type

Figures 5–7 complement the Prime Agent results in Figure 3 with task breakdowns for the RLM reader, Claude Code, and Codex. An X at a model's bar position marks an incompatible model–agent combination, following Table 10.

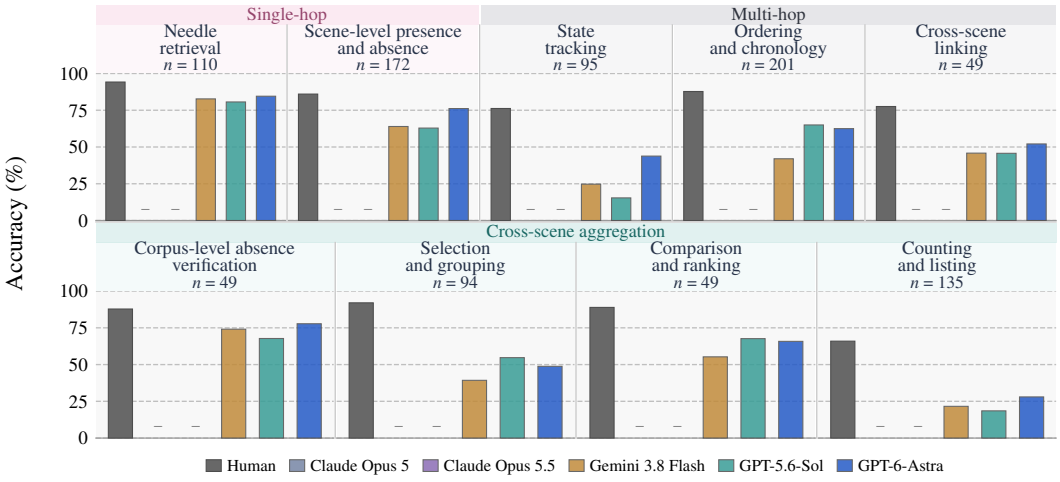


Figure 5: Accuracy with the RLM reader by task type. n denotes the number of questions per task. Claude Opus 5 and Opus 5.5 are not used with the RLM reader due to costs.

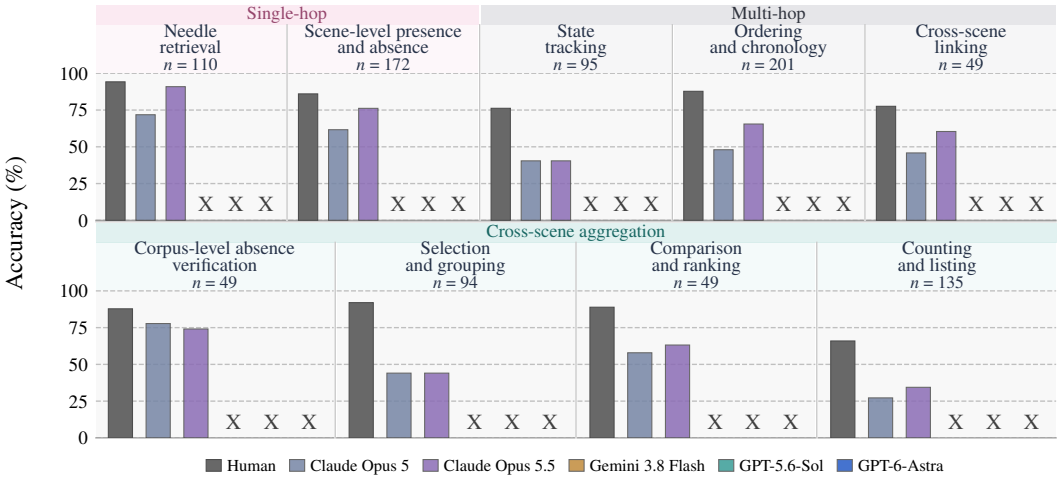


Figure 6: Accuracy with Claude Code by task type. n denotes the number of questions per task. Incompatible models are marked with X.

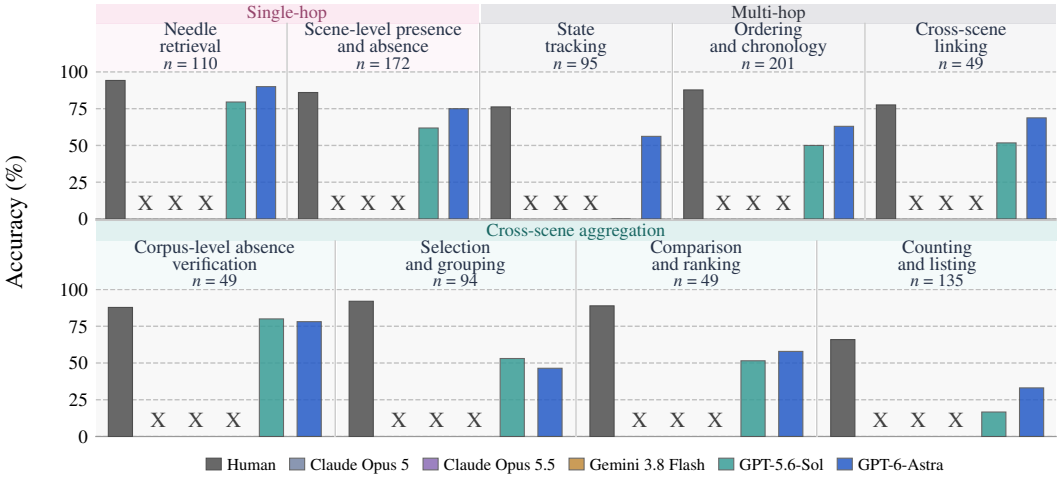


Figure 7: Accuracy with Codex by task type. n denotes the number of questions per task. Incompatible models are marked with X.