DOCHARNESS-RSI: FAILURE-GUIDED HARNESS EVOLUTION AROUND FROZEN VISION-LANGUAGE MODELS FOR LONG MULTIMODAL DOCUMENT UNDERSTANDING

Anonymous authors
Paper under double-blind review

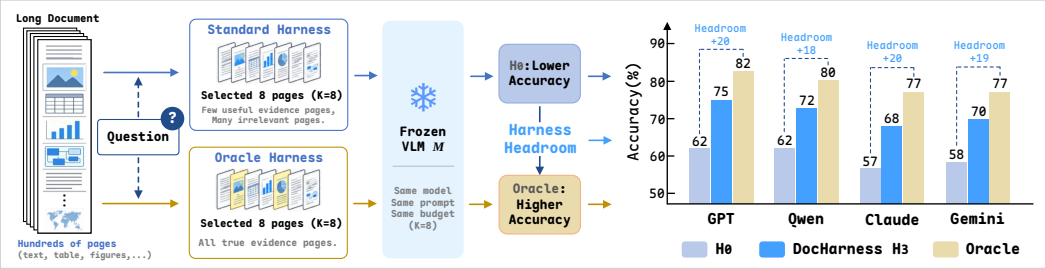

Figure 1: **Motivation of DocHarness-RSI.** Left: Under a fixed page budget and frozen VLM M, the standard harness H_0 selects irrelevant pages, whereas the oracle harness exposing true evidence achieves much higher accuracy. Right: The accuracy gap between H_0 and the oracle reveals a notable harness headroom of up to +20 points across frontier VLMs. Our evolved harness DocHarness H_3 substantially narrows this gap without modifying model parameters.

ABSTRACT

Answering questions over long multimodal documents requires locating sparse evidence scattered across complex layouts and hundreds of pages, all within a fixed page and context budget. Frontier Vision-Language Models (VLMs) reason well once the right pages are in view, yet the harness that selects and organizes those pages is typically a static retriever or a heuristic truncation rule, leading to missed evidence, incomplete cross-page coverage, and diluted context. We argue that this harness, rather than the model, is the primary bottleneck, and that optimizing it around a frozen VLM unlocks substantial headroom with no parameter updates. We present DocHarness-RSI, which treats the context-construction harness as an executable program and improves it recursively. The inner harness implements a Retrieve-Complete-Purify procedure: multi-channel retrieval surfaces candidate pages, question-conditioned expansion completes cross-page evidence chains, and utility-based purification removes low-value pages so the budget is spent on what matters. The outer loop is failure-guided: it inspects residual errors on development data, traces each one to the responsible harness stage, mutates only that stage, and admits a candidate program only after it passes dual-gated verification. Because the search is anchored to the currently binding bottleneck, effort migrates across stages as earlier failures are resolved, which keeps the harness compact while steadily raising both evidence recall and context purity. On long-document understanding benchmarks, the evolved harness delivers consistent gains over strong retrieval and truncation baselines, showing that executable harness evolution is an effective optimization layer for frozen VLM agents. Code will be released.

1 INTRODUCTION

Recent advances in Vision-Language Models (VLMs) have demonstrated strong capabilities in visual reasoning, chart understanding, and document analysis (Liu et al., 2023; Lee et al., 2023).

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.

However, when inputs scale from single pages to complex documents spanning tens or hundreds of pages, such as financial filings, technical reports, and slide decks, exhaustively processing the full document incurs prohibitive computational overhead, excessive latency, and high inference costs (Blau et al., 2024; Ma et al., 2024; Deng et al., 2025). Consequently, a fundamental question emerges regarding what document content should be presented to the model to enable accurate and cost-effective reasoning. Real-world documents exhibit high structural heterogeneity, where critical evidence is sparsely distributed across text, tables, figures, and cross-page layouts (Ma et al., 2024; Deng et al., 2025). Under practical resource and budget constraints, long-document understanding is fundamentally a context-construction problem.

Existing approaches primarily follow two lines. One line expands native context windows to ingest entire documents (Ding et al., 2024b). However, full-context input incurs heavy computational overhead and suffers from attention degradation on information located in the middle of long sequences (Liu et al., 2024). Another line uses Retrieval-Augmented Generation (RAG) to select query-relevant passages (Lewis et al., 2020). Yet standard RAG treats the model as a passive recipient of static context. When answers require synthesizing evidence across page boundaries or reading non-textual elements, single-pass retrieval often misses vital information (Asai et al., 2024).

Recent tool-augmented models demonstrate that context acquisition and organization decisively govern downstream performance (Yao et al., 2022; Schick et al., 2023; Sun et al., 2025a; Saad-Falcon et al., 2024), suggesting that failures on complex documents often stem from poor context construction rather than model reasoning deficits (Karpukhin et al., 2020). To verify this, our controlled pilot experiment in Fig. 1 compares a standard heuristic harness against an Oracle context built from ground-truth evidence pages under an identical page budget K and frozen VLM answerer. Across multiple frontier VLMs, isolating optimal evidence pages yields dramatic accuracy gains. This reveals substantial harness headroom, proving that: **The frozen model already possesses the reasoning capacity to answer the questions, but existing context constructors fail to supply the right evidence under bounded capacities.**

While VideoHarness-RSI (Xu & Chen, 2026) demonstrated recursive context optimization for video, documents present fundamentally distinct challenges. Unlike continuous video timelines, documents feature discrete page boundaries, heterogeneous layouts, and severe context competition under finite budgets. Analyzing document QA reveals three recurrent failure modes (Ding et al., 2024a; Deng et al., 2025; Ma et al., 2024), namely Evidence Miss from omitted non-textual or semantically distant pages, Cross-page Incompleteness from truncated multi-page chains, and Context Dilution from uninformative pages crowding the budget K. Document context curation must therefore resolve accessibility, completeness, and purity simultaneously.

To address these challenges, we propose DocHarness-**RSI**, a **R**ecursive harness **S**elf-**I**mprovement framework for long multimodal documents. DocHarness-RSI optimizes context construction around a frozen VLM via a dual-loop design. The inner harness implements a modular Retrieve-Complete-Purify (RCP) pipeline, where Retrieve fuses multi-channel features, Complete conditionally expands neighboring pages to recover evidence chains, and Purify filters redundant pages to maximize context density under budget K. The outer loop drives failure-guided evolution by diagnosing residual context errors on development data, restricting code mutations strictly to the identified bottleneck operator, and admitting candidates via dual-gated verification. As earlier bottlenecks are resolved, search effort dynamically shifts across operators while recognizing component saturation, effectively preventing pipeline bloat. Our main contributions are summarized as follows:

- We formalize document context construction as an executable program optimization layer around frozen VLMs, empirically demonstrating substantial harness headroom under a fixed page budget.
- We introduce the modular Retrieve-Complete-Purify pipeline to systematically address Evidence Miss, Cross-page Incompleteness, and Context Dilution in long documents.
- We establish a recursive optimization loop that diagnoses empirical failures, restricts programmatic search to diagnosed bottlenecks, and prevents development overfitting via dual-gated verification.
- Extensive evaluations show that DocHarness-RSI achieves state-of-the-art performance on MMLongBench-Doc over leading methods, while exhibiting strong zero-shot generalization on LongDocURL.

2 Related Work

2.1 Long-Document Understanding

Processing lengthy documents requires managing multimodal evidence across extensive page sequences. Early methods adopted hierarchical architectures that pair page-level OCR and visual encoders with small language model decoders (Dong et al., 2024), but suffered from OCR redundancy and error accumulation (Xu et al., 2020). Recently, multimodal large models have enabled two prominent directions. The first develops end-to-end multi-page architectures that dynamically compress visual tokens and scale instruction tuning (Appalaraju et al., 2021; Blau et al., 2024). The second employs Multimodal RAG (MM-RAG) to retrieve relevant page snapshots or text chunks (Zhang et al., 2024; Ding et al., 2024a). However, existing MM-RAG pipelines rely on hand-crafted heuristics that frequently drop non-adjacent evidence or dilute context under strict page budgets.

2.2 Tool-Augmented and Agentic Document Reasoning

Equipping models with tools enables active exploration that mimics human reading behaviors (Wang et al., 2024; Zhang et al., 2026). In document processing, agentic frameworks such as DocAgent and WorldMM (Sun et al., 2025a; Yeo et al., 2026) leverage hierarchical planning and graph-based retrieval to navigate multi-page corpora. Recent findings confirm that active tool navigation substantially outperforms passive context ingestion across complex enterprise formats like tables and slides Qiu et al. (2026); Shankar et al. (2024); Xie et al. (2024). Nevertheless, current agentic workflows remain manually engineered and do not adaptively optimize the executable context-curation code from downstream task feedback.

2.3 Automated Harness Optimization and Program Synthesis

Automated workflow discovery and program-level optimization have recently emerged to refine model wrappers without modifying model weights. Frameworks like ADAS (Hu et al., 2025) and AFlow (Zhang et al., 2025) optimize agent structures via execution feedback, connecting to LLM-guided program search (Romera-Paredes et al., 2024). Meta-Harness (Lee et al., 2026) synthesizes model harness code from prior traces, while VideoHarness (Xu & Chen, 2026) frames context selection as an executable program around frozen models for long video. Our proposed DocHarness-RSI extends this paradigm to multimodal documents, leveraging a modular Retrieve-Complete-Purify pipeline to achieve failure-guided recursive self-improvement around frozen vision-language models under bounded page budgets.

3 Method

3.1 Problem Formulation

Let $D = \{p_1, \ldots, p_N\}$ denote a long multimodal document comprising an ordered sequence of N physical pages, where each page $p_i = (I_i, T_i)$ contains visual rendering I_i and textual content T_i. Given a question q associated with D, exhaustively reading the entire document is computationally impractical when N substantially exceeds a context budget K. We therefore define an executable context harness $H \in \mathcal{H}_{\text{exec}}$ that maps the document and question to a bounded page context:

$$C_H = H(D, q; K), \tag{1}$$

where $C_H \subseteq D$ denotes the bounded document context constructed for the question and $|C_H| \leq K$. The final answer prediction is produced by a protocol-frozen Vision-Language Model (VLM) M:

$$\hat{y} = M(C_H, q), \tag{2}$$

where the model weights, decoding hyper-parameters, and task prompts of M remain strictly invariant across all evaluations. Rather than modifying the parameters of M, we optimize the context-construction logic over the executable harness space $\mathcal{H}_{\text{exec}}$. The optimization objective is:

$$H^* = \arg \max_{H \in \mathcal{H}_{\text{exec}}} \mathbb{E}_{(D,q,y)\sim\mathcal{D}} \left[\mathbb{I}\left(M(H(D, q; K), q) = y \right) \right], \tag{3}$$

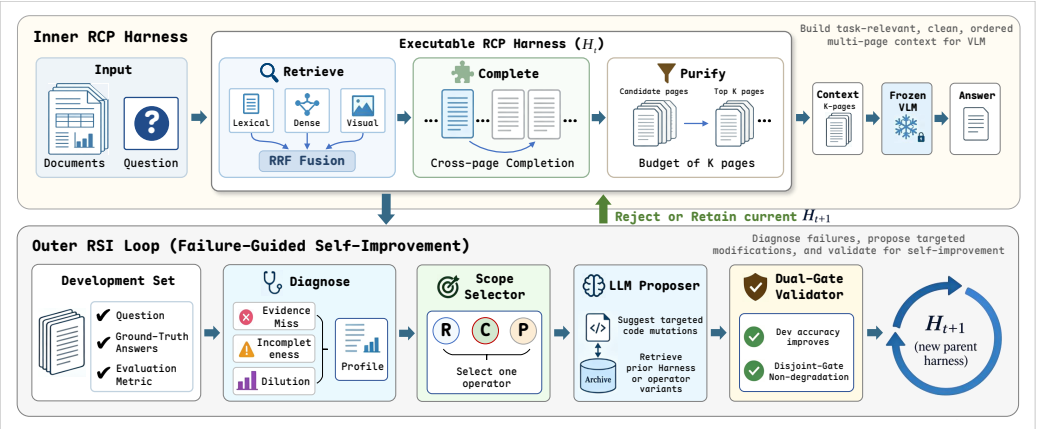

Figure 2: **Overview of the DocHarness-RSI framework.** Top: The inner executable RCP harness constructs a task-relevant, ordered context through Retrieve, Complete, and Purify before querying a protocol-frozen VLM. Bottom: The outer failure-guided RSI loop evaluates execution outcomes on a development set, profiles residual errors across the three failure modes, restricts LLM-based code mutation to the primary bottleneck operator via scope selection, and admits improved programs into the recursive lineage through dual-gated validation.

where $\mathbb{I}(\cdot)$ is the indicator function. During search, Eq. (3) is evaluated empirically on a development set $\mathcal{D}_{\text{dev}}$. This formulation cleanly decouples context construction from the intrinsic reasoning capability of the downstream model: As shown in Fig. 2, under a fixed page budget, DocHarness-RSI optimizes which multimodal evidence the frozen model observes.

3.2 Document-Specific RCP Harness

Unlike sequential video frames (Wang et al., 2024; Zhang et al., 2026), long documents scatter evidence across heterogeneous page modalities and layouts. Under a constrained page budget K, context construction suffers from three primary failures, including **(i) Evidence Miss** from single-modality retrieval omissions, **(ii) Cross-page Incompleteness** from truncated multi-page chains, and **(iii) Context Dilution** from redundant pages crowding out valid evidence. To systematically resolve these bottlenecks, as outlined in Algorithm 1, we formulate the inner harness H through three modular operators, **R**etrieve, **C**omplete, and **P**urify (RCP).

Retrieve: Mitigating Evidence Miss. Single-modality retrieval frequently overlooks evidence residing in complex figures or fine-grained text. To maximize recall, each page $p \in D$ is represented via complementary frozen features $\{f_{\text{lex}}(p), f_{\text{dense}}(p), f_{\text{vis}}(p)\}$ across channels $\mathcal{C} = \{\text{lex}, \text{dense}, \text{vis}\}$. Let $\pi_c(p)$ denote the rank of page p under channel c based on its similarity to question q. We fuse these disparate rankings via Reciprocal Rank Fusion:

$$S_{\text{ret}}(p) = \sum_{c \in \mathcal{C}} \frac{w_c}{k_{\text{rrf}} + \pi_c(p)}, \tag{4}$$

where $w_c \geq 0$ is the channel weight and k_{rrf} is a smoothing constant. The top-M pages ranked by S_{ret} form the candidate evidence pool $E_{\text{ret}} \subset D$.

Complete: Resolving Cross-page Incompleteness. A retrieved page often serves merely as an anchor for multi-page procedures or split tables, yielding incomplete evidence in isolation. Given pool E_{ret}, we identify high-confidence seed pages $E_{\text{seed}} \subseteq E_{\text{ret}}$ and augment candidates via a conditional neighborhood operator $\mathcal{N}(\cdot)$:

$$E_{\text{comp}} = E_{\text{ret}} \cup \mathcal{N}(E_{\text{seed}}, q). \tag{5}$$

The expansion neighborhood around seed pages is defined as:

$$\mathcal{N}(E_{\text{seed}}, q) = \bigcup_{p_i \in E_{\text{seed}}} \{p_j \in D \mid |j - i| \leq \delta(q)\}, \tag{6}$$

where $\delta(q)$ denotes an expansion radius inferred from question continuity cues. For isolated facts, $\delta(q) = 0$ to avoid squandering the context budget on unneeded pages.

Algorithm 1 RCP Context Construction and Inference

Require: Question q; long document $D = \{p_1, \ldots, p_N\}$; page budget K; frozen VLM M.
Ensure: Bounded page context C_H; answer prediction $\hat{y}$.

Phase 1: Multi-Channel Retrieval ▷ Mitigate Evidence Miss
for each channel $c \in \{\text{lex}, \text{dense}, \text{vis}\}$ **do**
  Compute page rank list π_c based on similarity between $f_c(p)$ and $f_c(q)$
end for
for each page $p \in D$ **do**
  $S_{\text{ret}}(p) \leftarrow \sum_c \frac{w_c}{k_{\text{rrf}} + \pi_c(p)}$ ▷ Reciprocal Rank Fusion, Eq. (4)
end for
$E_{\text{ret}} \leftarrow$ Top-$M(S_{\text{ret}})$; Select high-confidence seeds $E_{\text{seed}} \subseteq E_{\text{ret}}$
Phase 2: Cross-Page Completion ▷ Mitigate Cross-page Incompleteness
Estimate question continuity radius $\delta(q) \in \{0, 1, 2\}$
$\mathcal{N}(E_{\text{seed}}, q) \leftarrow \bigcup_{p_i \in E_{\text{seed}}} \{p_j \in D \mid |j - i| \leq \delta(q)\}$ ▷ Eq. (6)
$E_{\text{comp}} \leftarrow E_{\text{ret}} \cup \mathcal{N}(E_{\text{seed}}, q)$
Phase 3: Multi-Objective Purification ▷ Mitigate Context Dilution
Initialize selected context set $\mathcal{S} \leftarrow \emptyset$
while $|\mathcal{S}| < K$ **and** $E_{\text{comp}} \setminus \mathcal{S} \neq \emptyset$ **do**
  $p^* \leftarrow \arg\max_{p \in E_{\text{comp}} \setminus \mathcal{S}} S_{\text{pur}}(p \mid \mathcal{S})$ ▷ Multi-objective utility, Eq. (7)
  $\mathcal{S} \leftarrow \mathcal{S} \cup \{p^*\}$
end while
$C_H \leftarrow \text{SortByPageIdx}(\mathcal{S})$ ▷ Restore ascending physical order
Phase 4: Downstream Inference
$\hat{y} \leftarrow M(C_H, q)$ ▷ Protocol-frozen VLM inference, Eq. (2)
return $C_H, \hat{y}$

Purify: Eliminating Context Dilution. While retrieval and completion expand coverage, the resulting candidate pool often exceeds capacity ($|E_{\text{comp}}| > K$) and includes redundant pages. To assemble context C_H within budget K, we formulate purification as a multi-objective selection problem. For each candidate $p \in E_{\text{comp}}$ relative to the selected subset $\mathcal{S}$, we compute a composite utility score

$$S_{\text{pur}}(p \mid \mathcal{S}) = \lambda_r S_{\text{ret}}(p) + \lambda_c S_{\text{cov}}(p, q) + \lambda_d S_{\text{div}}(p, \mathcal{S}), \tag{7}$$

where $S_{\text{ret}}(p)$ denotes relevance from Eq. (4), $S_{\text{cov}}(p, q)$ measures question coverage, and $S_{\text{div}}(p, \mathcal{S})$ penalizes redundancy relative to $\mathcal{S}$. The λ_r, λ_c and λ_d are 1, 0.35 and 0.15, respectively. The purification operator greedily retains pages maximizing Eq. (7) up to budget K, and restores them to physical page order to preserve reading coherence for the downstream model M.

3.3 Failure-Guided Harness Evolution

While the inner RCP harness constructs context, the outer loop optimizes the harness code. Because unconstrained pipeline mutations yield an intractable search space and disrupt functioning modules, DocHarness-RSI diagnoses the dominant residual failure each iteration and confines program synthesis strictly to the bottlenecked RCP component.

Diagnostic Profiling and Scope Selection. At generation t, the incumbent harness H_t is evaluated on development set $\mathcal{D}_{\text{dev}}$ to produce a structured failure profile:

$$F_t = \text{Diagnose}(H_t, \mathcal{D}_{\text{dev}}), \tag{8}$$

which quantifies the empirical error distribution across Evidence Miss, Cross-page Incompleteness, and Context Dilution. The evolution controller identifies the primary bottleneck in F_t and assigns an active mutation scope:

$$s_t \in \{\mathcal{R}, \mathcal{C}, \mathcal{P}\}, \tag{9}$$

corresponding to the Retrieve, Complete, or Purify operator. Only the component within scope s_t is mutable during generation t, while all remaining pipeline logic and contracts remain frozen.

Targeted Harness Mutation. An outer-loop language model acts as the automated harness designer. Maintaining an evolution archive $\mathcal{A}_t = \{(H_\tau, \mathcal{S}_{\text{dev}}(H_\tau), F_\tau)\}_{\tau<t}$ of historical code, development scores, and diagnostic profiles, the designer samples J candidate harnesses:

$$\{H_t^{(j)}\}_{j=1}^{J} \sim P_{\text{mut}}(H_t, F_t, \mathcal{A}_t; s_t). \tag{10}$$

Restricting mutations strictly to scope s_t ensures localized programmatic repairs targeting the diagnosed bottleneck, preventing unconstrained rewrites of the broader pipeline.

Dual-Gated Evaluation and Retention. Candidate harnesses are evaluated end-to-end under the frozen inference protocol, selecting the variant with the highest development accuracy:

$$H_t^{\dagger} = \underset{H \in \{H_t^{(j)}\}_{j=1}^{J}}{\operatorname{argmax}} \; \mathcal{S}_{\text{dev}}(H). \tag{11}$$

To prevent overfitting to $\mathcal{D}_{\text{dev}}$, promotion incorporates an independent, document-disjoint gate set $\mathcal{D}_{\text{gate}}$ via a non-degradation indicator:

$$G(H_t^{\dagger}, H_t) = \mathbb{I}\left(\mathcal{S}_{\text{gate}}(H_t^{\dagger}) \geq \mathcal{S}_{\text{gate}}(H_t)\right). \tag{12}$$

The incumbent harness updates via the dual-gated rule:

$$H_{t+1} = \begin{cases} H_t^{\dagger}, & \text{if } \mathcal{S}_{\text{dev}}(H_t^{\dagger}) > \mathcal{S}_{\text{dev}}(H_t) \;\wedge\; G(H_t^{\dagger}, H_t) = 1, \\ H_t, & \text{otherwise.} \end{cases} \tag{13}$$

If no candidate satisfies both criteria, H_t is retained, and negative outcomes are logged into $\mathcal{A}_{t+1}$ to divert future mutations away from unpromising trajectories.

3.4 Recursive Self-Improvement

When an improved candidate becomes the new parent H_{t+1}, DocHarness-RSI re-evaluates it to continuously adapt to shifting failure distributions. The active mutation scope dynamically tracks the dominant bottleneck in profile $F_t = \text{Diagnose}(H_t, \mathcal{D}_{\text{dev}})$:

$$s_t = \underset{k \in \{R,C,P\}}{\operatorname{argmax}} \; F_t[k]. \tag{14}$$

Conditioned on scope s_t, candidate programs are sampled via Eq. (10), formalizing the recursive progression:

$$H_{t+1} = \text{Retain}\left(H_t, \{H_t^{(j)}\}_{j=1}^{J}\right), \tag{15}$$

where $\text{Retain}(\cdot)$ executes the dual-gated rule in Eq. (13). This closed loop directly addresses inter-operator coupling. For instance, resolving Evidence Miss often causes Context Dilution, automatically shifting subsequent search focus from Retrieve to Purify. When mutations within a scope plateau, archive $\mathcal{A}_t$ detects saturation and reallocates proposal budgets toward under-optimized operators. Thus, recursive self-improvement progresses by dynamically reallocating search effort toward shifting bottlenecks rather than compounding pipeline complexity.

4 Experiments

4.1 Experimental Setup

Datasets. Experiments are conducted on two long-document benchmarks: MMLongBench-Doc (Ma et al., 2024) and LongDocURL (Deng et al., 2025). MMLongBench-Doc serves as our primary benchmark for failure-guided harness evolution and controlled evaluation, where all development, gating, and held-out partitions are strictly split at the document level to prevent data contamination. To rigorously evaluate whether the discovered context constructor generalizes beyond its discovery domain, LongDocURL is used exclusively for zero-shot cross-dataset evaluation. Specifically, the harness policy evolved on MMLongBench-Doc is directly transferred to LongDocURL without accessing any target labels, ground-truth evidence, or performance feedback for policy adaptation.

Table 1: **Evaluation of various methods on MMLongBench-Doc.** We report the accuracy(%)↑ of seven types of document domains. Best performance in **bold**.

Method	Venue	Category	Pure-Text↑	Layout↑	Chart↑	Table↑	Figure↑	Single↑	Cross↑	Overall↑
M3DocRAG	arXiv'24	MM-RAG	73.14	70.67	74.16	71.54	61.67	81.44	57.86	71.40
MoLoRAG	EMNLP'25	MM-RAG	72.57	60.00	71.67	80.49	60.55	82.13	58.33	71.99
VisRAG 2.0	MM'25	Visual-RAG	71.43	62.67	68.33	75.61	63.33	80.07	56.02	69.82
DocAgent	EMNLP'25	Agentic	72.57	58.67	**74.16**	77.23	62.78	80.41	62.04	72.58
SLEUTH	CVPR'26	Agentic	73.14	57.33	73.34	**82.92**	62.23	82.81	59.26	72.78
TextRAG(H0)	Ours	Text-RAG	70.29	64.00	64.17	56.91	53.33	71.82	49.07	62.13
DocHarness-RSI	Ours	Harness-RSI	**74.86**	**74.67**	70.83	76.42	**71.67**	**83.51**	**63.89**	**75.15**

Evaluation Metrics. Following the official evaluation protocols of each benchmark, we evaluate downstream task performance alongside fine-grained category breakdowns. For MMLongBench-Doc, model responses are evaluated under the official semantic-equivalence protocol using an automated judge (Ma et al., 2024), reporting overall accuracy across diverse evidence modalities (Pure-Text, Layout, Chart, Table, Figure) and scopes (Single-page, Cross-page), accompanied by post-hoc evidence coverage diagnostics. For LongDocURL, we adopt the benchmark-native evaluation pipeline (Deng et al., 2025) by extracting standardized short answers and computing deterministic scores across Locating, Understanding, and Reasoning tasks.

Implementation Details. All experiments operate under a strictly unified protocol with a maximum page budget of $K \leq 8$, where selected pages are restored to their physical document order before answering. Multi-channel retrieval combines BM25 lexical search, local dense text encoding with MiniLM (Wang et al., 2020), and local visual encoding with CLIP (Radford et al., 2021), executed on an NVIDIA RTX 4090 GPU. We freeze GPT-5.6-Terra (OpenAI, 2026) as the primary vision-language model for both answer generation and purification, keeping prompts, decoding configurations, and interfaces invariant across all baselines and harness variants without any parameter fine-tuning. More details are provided in Appendix A.

Baselines. Under the unified budget and frozen VLM protocol, we compare DocHarness-RSI against six representative baselines spanning three major paradigms. These include static lexical retrieval via TextRAG, multimodal RAG methods such as M3DocRAG (Cho et al., 2024), MoLoRAG (Wu et al., 2025), and visual-centric VisRAG 2.0 (Sun et al., 2025b), as well as interactive agentic frameworks including DocAgent (Sun et al., 2025a) and SLEUTH (Liu et al., 2026). These baselines ensure a comprehensive evaluation across text, vision, and agent-based context curation strategies.

4.2 Overall Performance

Quantitative Results on MMLongBench-Doc. Tab. 1 demonstrates that DocHarness-RSI establishes a new state-of-the-art with 75.15% overall accuracy, outperforming the leading agentic baseline SLEUTH (72.78%) and surpassing the TextRAG baseline H_0 (62.13%) by 13.02 points under the identical frozen VLM protocol. Category breakdowns directly validate our targeted design against the core failure modes. In visually structured scenarios, DocHarness-RSI achieves top performance on Figure (71.67%) and Layout (74.67%), confirming that multi-channel fusion effectively eliminates Evidence Miss. More importantly, on Cross-page reasoning, our approach attains the highest accuracy of 63.89%, outperforming DocAgent (62.04%) and SLEUTH (59.26%). This demonstrates that conditional cross-page completion reliably overcomes Cross-page Incompleteness, while evidence purification maintains high context density under the strict budget $K = 8$.

Quantitative Results on LongDocURL. Tab. 2 evaluates the zero-shot cross-dataset transferability of the harness policy evolved on MMLongBench-Doc to LongDocURL without any target adaptation. DocHarness-RSI achieves top performance across all evaluation dimensions, reaching an overall score of 66.36% and outperforming the strongest baseline DocAgent (63.78%) by 2.58 points and the TextRAG baseline H_0 (60.13%) by 6.23 points. Across specific tasks, our approach consistently leads in Locating (62.21%), Understanding (68.58%), and Reasoning (66.67%), showing substantial gains on complex synthesis (+3.33 points on Understanding). This robust cross-benchmark performance confirms that the evolved Retrieve-Complete-Purify pipeline captures universal principles of evidence organization rather than benchmark-specific regularities, verifying that dual-gated verification effectively ensures out-of-distribution generalization.

Table 2: **Evaluation of various methods on LongDocURL.** We report the accuracy(%)↑ of three types of document domains. Best performance in **bold**.

Method	Venue	Category	Locating↑	Understanding↑	Reasoning↑	Overall↑
M3DocRAG	arXiv'24	MM-RAG	56.59	61.40	58.12	59.39
MoLoRAG	EMNLP'25	MM-RAG	58.42	63.96	63.05	62.15
VisRAG 2.0	MM'25	Visual-RAG	58.85	63.31	61.49	61.68
DocAgent	EMNLP'25	Agentic	61.01	65.25	64.08	63.78
SLEUTH	CVPR'26	Agentic	59.86	63.79	62.02	62.32
TextRAG(H0)	Ours	Text-RAG	54.77	63.32	59.52	60.13
DocHarness-RSI	Ours	Harness-RSI	**62.21**	**68.58**	**66.67**	**66.36**

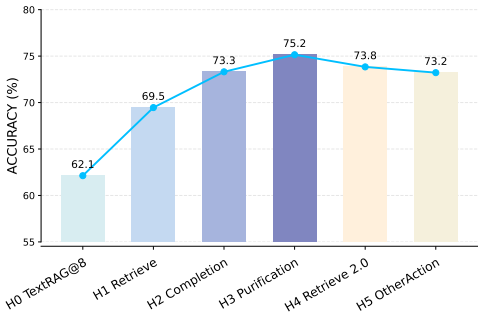

Figure 3: **Ablation on Harness Evolution.**

Table 3: **Ablation on Harness Context Mechanism Analysis.** *Hit-all* measures the percentage of instances where all ground-truth evidence pages are covered. *Rec.* is Evidence Recall, and *Acc.* represents context purity measuring the ratio of true evidence pages.

Stage	Mean Pages	Hit-all (%)↑	Evidence Rec.(%)↑	Evidence Acc.(%)↑
H0(TextRAG)	8	63.71	71.65	13.73
H1(Retrieve)	8	78.30	85.40	17.11
H2(Complete)	8	84.22	90.38	18.59
H3(Purify)	2	73.18	80.79	83.86

4.3 Ablation Studies

Ablation on Harness Evolution. Fig. 3 traces the performance progression across search generations from H_0 to H_5 on the MMLongBench-Doc dataset. Recursive optimization yields consistent gains across the first three iterations, validating the cumulative efficacy of the RCP pipeline. Transitioning from H_0 (62.1%) to multi-channel retrieval H_1 (69.5%) substantially mitigates Evidence Miss. Adding conditional cross-page completion in H_2 further lifts accuracy to 73.3% by recovering multi-page chains, while evidence purification in H_3 achieves the peak accuracy of 75.2% by eliminating Context Dilution under budget $K = 8$. Crucially, subsequent candidates H_4 (73.8%) and H_5 (73.2%) fail to surpass the incumbent frontier and are strictly rejected by the dual-gated rule in Eq. (13). This confirms that DocHarness-RSI prevents performance regression and detects algorithmic saturation without inflating pipeline complexity.

Ablation on Harness Context Mechanism Analysis. Tab. 3 reveals how each operator alters context composition to systematically resolve the three target failure modes. Transitioning from H_0 to multi-channel retrieval H_1 elevates Hit-all from 63.71% to 78.30% and Evidence Recall from 71.65% to 85.40%, successfully mitigating Evidence Miss. Adding cross-page completion in H_2 achieves peak evidence coverage (84.22% Hit-all, 90.38% Recall), resolving Cross-page Incompleteness. Finally, evidence purification in H_3 eliminates Context Dilution by compressing average context from 8 to 2 pages while surging Evidence Accuracy from 18.59% to 83.86% with 80.79% recall. This dramatic boost in evidence purity supplies the frozen VLM with concentrated, noise-free context, underpinning our downstream accuracy peak.

Ablation on Failure-Guided Harness Distribution Shift. Fig. 4 traces empirical failure distributions across generations on the LongDocURL, validating the recursive closed loop formalized in Section 3.4. In H_0, Evidence Miss dominates at 38%. Multi-channel retrieval in H_1 halves Evidence Miss to 19%, yet candidate expansion inflates Context Dilution from 27% to 49%, shifting the primary error mode. In H_2, cross-page completion reduces Evidence Miss to 7% and Incompleteness to 27%, but drives Context Dilution to a peak of 66%. Crucially, this empirical surge prompts the scope selector to target Purify ($s_2 = P$), allowing H_3 to curtail Context Dilution by 18 percentage points to 48% and achieve a balanced residual profile. This progression confirms that DocHarness-RSI adaptively diagnoses and resolves shifting bottlenecks rather than following a static optimization path.

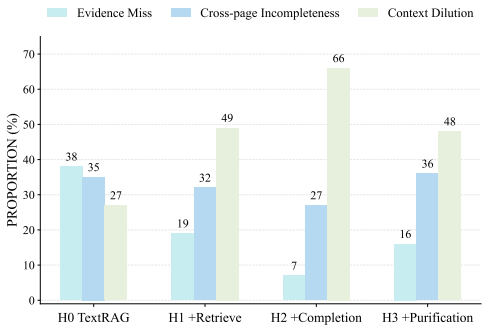

Figure 4: **Ablation on Failure-Guided Harness Distribution Shift.**

Table 4: **Ablation on Failure-Guided Harness Evolution Outer Loop Components.** Scope Path traces the generational sequence of active operators across iterations, where *R*, *C*, and *P* denote Retrieve, Complete, and Purify, respectively. The *w/o* denotes without the corresponding module.

Method	Fail-Guid	Dual-Gate	Scope Path	Acc. (%)↑
Fail-Guid(w/o)		✓	R→C→R	67.74
Dual-Gate(w/o)	✓		C→R→C	70.23
Full RSI	✓	✓	R→C→P	75.15

Ablation on Failure-Guided Harness Evolution Outer Loop. Tab. 4 isolates the contributions of failure-guided scoping and dual-gated verification in the outer loop. Removing failure guidance causes a steep accuracy drop of 7.41 percentage points from 75.15% to 67.74%, as the LLM proposer follows an unguided trajectory ($R \rightarrow C \rightarrow R$) that squanders proposal budgets and completely overlooks the Purify operator. Omitting dual-gated verification similarly degrades performance to 70.23% (-4.92 points), where the absence of a document-disjoint safeguard leads to development-set overfitting along a suboptimal path ($C \rightarrow R \rightarrow C$). In contrast, Full RSI successfully navigates the structured $R \rightarrow C \rightarrow P$ progression to attain the peak accuracy of 75.15%. This confirms that coupling diagnostic attribution with conservative dual-gated validation is essential for stable and generalizable programmatic self-improvement.

Ablation on Cross-Model Generalization. Tab. 5 evaluates the transferability of our proposed DocHarness-RSI across diverse downstream vision-language models, pairing the optimal constructor with four distinct answerers without re-running harness search. Across all models, DocHarness-RSI maintains consistently strong performance on both MMLongBench-Doc (68.65% to 75.15%) and LongDocURL (62.34% to 66.36%). Remarkably, deploying our harness with the lightweight Qwen-3.8Flash achieves 72.78% on MMLongBench-Doc and 63.88% on LongDocURL, outperforming several full-pipeline baselines in Tab. 1 and Tab. 2. This confirms that DocHarness-RSI captures universal evidence curation principles, establishing executable context construction as a robust, model-agnostic optimization layer.

Table 5: **Ablation on Downstream Frozen VLMs.**

Method	Dataset Accuracy Avg.↑	
	MMLongBench.	LongDocURL
Ours(Qwen3.8-Flash)	72.78%	63.88%
Ours(Gemini 3.6Flash)	70.19%	64.47%
Ours(Claude Sonnet 5)	68.65%	62.34%
Ours(GPT-5.6-Terra)	75.15%	66.36%

5 Conclusion

In this paper, we show that context construction is the primary bottleneck in long multimodal document understanding, and propose DocHarness-RSI to recursively optimize executable context constructors around a frozen VLM. By uniting an inner Retrieve-Complete-Purify pipeline with a failure-guided, dual-gated outer loop, our framework systematically resolves evidence miss, cross-page incompleteness, and context dilution under strict page budgets. Extensive evaluations confirm that DocHarness-RSI unlocks substantial performance headroom without parameter updates, establishing executable harness evolution as an effective optimization layer for frozen VLM systems.

AI use statement

During the preparation of this work, we utilized artificial intelligence tools to assist with manuscript language polishing, implementation of select experimental scripts, interpretation of empirical results, and design guidance for figures, tables, and typography to enhance readability. All AI-assisted contributions were rigorously audited and verified by the authors. Specifically, all experimental protocols, quantitative results, and bibliographic references were thoroughly inspected and validated by the human research team. The authors assume full responsibility for all content, findings, and conclusions presented in this paper.

REFERENCES

Srikar Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie, and R Manmatha. Docformer: End-to-end transformer for document understanding. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 973–983. IEEE, 2021.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avi Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *International conference on learning representations*, volume 2024, pp. 9112–9141, 2024.

Tsachi Blau, Sharon Fogel, Roi Ronen, Alona Golts, Roy Ganz, Elad Ben Avraham, Aviad Aberdam, Shahar Tsiper, and Ron Litman. Gram: Global reasoning for multi-page vqa. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15598–15607, 2024.

Jaemin Cho, Debanjan Mahata, Ozan Irsoy, Yujie He, and Mohit Bansal. M3docrag: Multimodal retrieval is what you need for multi-page multi-document understanding. *arXiv preprint arXiv:2411.04952*, 2024.

Chao Deng, Jiale Yuan, Pi Bu, Peijie Wang, Zhong-Zhi Li, Jian Xu, Xiao-Hui Li, Yuan Gao, Jun Song, Bo Zheng, et al. Longdocurl: a comprehensive multimodal long document benchmark integrating understanding, reasoning, and locating. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1135–1159, 2025.

Yihao Ding, Kaixuan Ren, Jiabin Huang, Siwen Luo, and Soyeon Caren Han. Mmvqa: A comprehensive dataset for investigating multipage multimodal information retrieval in pdf-based visual question answering. In *IJCAI*, pp. 6243–6251, 2024a.

Yiran Ding, Li Lyna Zhang, Chengruidong Zhang, Yuanyuan Xu, Ning Shang, Jiahang Xu, Fan Yang, and Mao Yang. Longrope: Extending llm context window beyond 2 million tokens. *arXiv preprint arXiv:2402.13753*, 2024b.

Qi Dong, Lei Kang, and Dimosthenis Karatzas. Multi-page document vqa with recurrent memory transformer. In *International Workshop on Document Analysis Systems*, pp. 57–70. Springer, 2024.

Shengran Hu, Cong Lu, and Jeff Clune. Automated design of agentic systems. In *International Conference on Learning Representations*, volume 2025, pp. 21344–21377, 2025.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pp. 6769–6781, 2020.

Kenton Lee, Mandar Joshi, Iulia Raluca Turc, Hexiang Hu, Fangyu Liu, Julian Martin Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. Pix2struct: Screenshot parsing as pretraining for visual language understanding. In *International Conference on Machine Learning*, pp. 18893–18912. PMLR, 2023.

Yoonho Lee, Roshen Nair, Qizheng Zhang, Kangwook Lee, Omar Khattab, and Chelsea Finn. Meta-harness: End-to-end optimization of model harnesses. *arXiv preprint arXiv:2603.28052*, 2026.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474, 2020.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

Keliang Liu, Zizhi Chen, Mingcheng Li, Jingqun Tang, Dingkang Yang, and Lihua Zhang. Resolving evidence sparsity: Agentic context engineering for long-document understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19452–19462, 2026.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the association for computational linguistics*, 12:157–173, 2024.

Yubo Ma, Yuhang Zang, Liangyu Chen, Meiqi Chen, Yizhu Jiao, Xinze Li, Xinyuan Lu, Ziyu Liu, Yan Ma, Xiaoyi Dong, et al. Mmlongbench-doc: Benchmarking long-context document understanding with visualizations. *Advances in Neural Information Processing Systems*, 37:95963–96010, 2024.

OpenAI. GPT-5.6 System Card. `https://deploymentsafety.openai.com/gpt-5-6`, July 2026. Accessed: September 21, 2026.

Chenhao Qiu, Yechao Zhang, Xin Luo, Shien Song, and Xusheng Liu. Videoseal: Mitigating evidence misalignment in agentic long video understanding by decoupling answer authority. *arXiv preprint arXiv:2605.12571*, 2026.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balog, M Pawan Kumar, Emilien Dupont, Francisco JR Ruiz, Jordan S Ellenberg, Pengming Wang, Omar Fawzi, et al. Mathematical discoveries from program search with large language models. *Nature*, 625(7995):468–475, 2024.

Jon Saad-Falcon, Joe Barrow, Alexa Siu, Ani Nenkova, Seunghyun Yoon, Ryan A Rossi, and Franck Dernoncourt. Pdftriage: Question answering over long, structured documents. In *Proceedings of the 2024 conference on empirical methods in natural language processing: Industry track*, pp. 153–169, 2024.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in neural information processing systems*, 36:68539–68551, 2023.

Shreya Shankar, Tristan Chambers, Tarak Shah, Aditya G Parameswaran, and Eugene Wu. Docetl: Agentic query rewriting and evaluation for complex document processing. *arXiv preprint arXiv:2410.12189*, 2024.

Li Sun, Liu He, Shuyue Jia, Yangfan He, and Chenyu You. Docagent: An agentic framework for multi-modal long-context document understanding. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 17712–17727, 2025a.

Yubo Sun, Chunyi Peng, Yukun Yan, Shi Yu, Zhenghao Liu, Chi Chen, Zhiyuan Liu, and Maosong Sun. Visrag 2.0: Evidence-guided multi-image reasoning in visual retrieval-augmented generation. *arXiv preprint arXiv:2510.09733*, 2025b.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in neural information processing systems*, 33:5776–5788, 2020.

Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. Videoagent: Long-form video understanding with large language model as agent. In *European Conference on Computer Vision*, pp. 58–76. Springer, 2024.

Xixi Wu, Yanchao Tan, Nan Hou, Ruiyang Zhang, and Hong Cheng. Molorag: Bootstrapping document understanding via multi-modal logic-aware retrieval. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 14035–14056, 2025.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh J Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *Advances in Neural Information Processing Systems*, 37:52040–52094, 2024.

Guoyang Xu and Hao Chen. Videoharness-rsi: Recursive harness self-improvement for long-video understanding with frozen vision-language models. *arXiv preprint arXiv:2608.24302*, 2026.

Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. Layoutlm: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 1192–1200, 2020.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022.

Woongyeong Yeo, Kangsan Kim, Jaehong Yoon, and Sung Ju Hwang. Worldmm: Dynamic multimodal memory agent for long video reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 25599–25609, 2026.

Jiayi Zhang, Jinyu Xiang, Zhaoyang Yu, Fengwei Teng, Xionghui Chen, Jiaqi Chen, Mingchen Zhuge, Xin Cheng, Sirui Hong, Jinlin Wang, et al. Aflow: Automating agentic workflow generation. In *International Conference on Learning Representations*, volume 2025, pp. 34040–34077, 2025.

Jinxu Zhang, Yongqi Yu, and Yu Zhang. Cream: coarse-to-fine retrieval and multi-modal efficient tuning for document vqa. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 925–934, 2024.

Xiaoyi Zhang, Zhaoyang Jia, Zongyu Guo, Jiahao Li, Bin Li, Houqiang Li, and Yan Lu. Deep video discovery: Agentic search with tool use for long-form video understanding. *Advances in Neural Information Processing Systems*, 38:89863–89895, 2026.

A Implementation Details, Hyperparameters, and System Costs

This appendix provides a comprehensive record of the experimental environment, algorithmic hyperparameters, and runtime accounting for DocHarness-RSI. Page indices used throughout context construction correspond to zero-based physical PDF pages, and all final contexts are deduplicated and restored to natural reading order before querying the downstream model. The final context budget is strictly capped at $K \leq 8$ pages.

A.1 Implementation Details and Environment Setup

Hardware and Software Infrastructure All local feature extraction, rendering routines, and offline embedding computations are executed on a dedicated workstation. The detailed hardware specifications and software dependencies are summarized in Table 6. Local compute is restricted to an NVIDIA GeForce RTX 4090 GPU, while the frozen downstream answerer, the purification verifier, and benchmark evaluators are accessed via OpenAI-compatible API interfaces.

Table 6: **System architecture and software dependencies used in DocHarness-RSI.**

Component	Configuration
Operating System	Ubuntu 22.04.1 LTS (x86_64)
Host Processor	2 × Intel Xeon Gold 6342
System Memory	629 GiB RAM
Local GPU	NVIDIA GeForce RTX 4090
GPU Driver & CUDA	Driver 570.133.07, CUDA Toolkit 12.3
Python Environment	Python 3.11.16, PyTorch 2.9.1+cu128
Core Libraries	Transformers 4.57.6, PyMuPDF 1.28.2, Pillow 12.3.0
Frozen Remote VLM	GPT-5.6-Terra (high reasoning effort, high image detail)

Deterministic Page Rendering and Text Extraction PDF pages are rasterized into images using PyMuPDF under a deterministic configuration. Each page is rendered at 144 DPI with an initial scale factor of 2.0 without alpha transparency, converted to RGB, and resized using Lanczos resampling such that its maximum long edge does not exceed 2,000 pixels. Rendered pages are stored as baseline JPEG images at quality 95 with chroma subsampling disabled. For page text, native PDF text layers are extracted with topological reading-order sorting enabled, followed by Unicode NFKC normalization. When a page lacks a usable native text layer, the system automatically falls back to RapidOCR (version 3.9.2 with PP-OCRv6-small) executed over the 144 DPI rendering. Page text is exclusively utilized for sparse retrieval, outline extraction, and candidate purification, whereas the downstream answer model primarily consumes high-resolution page images.

Context Construction Firewall To ensure rigorous evaluation and prevent information leakage, the context harness operates behind an isolated firewall. Ground-truth evidence page annotations, reference answers, dataset answer formats, and modality classification labels are strictly inaccessible during the retrieval, completion, and purification stages. These fields are loaded solely during post-hoc diagnostic auditing and official benchmark judging.

Prompt Specifications The answer generation prompt is strictly frozen across all baseline comparisons and evolutionary iterations. The system prompt instructs the model to answer concisely using only the supplied document pages without providing auxiliary citations or reasoning, and to return exactly "Not answerable" when evidence is insufficient. Each page image is prepended with its zero-based physical page index to preserve structural referencing. For evidence purification, the verifier prompt directs the model to inspect candidate pages and select the smallest sufficient subset of at most 8 pages based on relevance, sufficiency, cross-page continuity, contradiction, and redundancy, returning structured JSON indices.

A.2 Component Hyperparameters and Algorithmic Specifications

DocHarness-RSI structures context curation into three modular operators, governed by frozen algorithmic parameters established during development.

Table 7: **Algorithmic hyperparameters across the Retrieve, Complete, and Purify operators.**

Operator	Parameter	Value / Configuration
Retrieve	BM25 Parameters	$k_1 = 1.5$, $b = 0.75$, $\epsilon = 0.25$, regex alphanumeric tokenization
	Dense Text Encoder	all-MiniLM-L6-v2 (384-dim, 512 max tokens, L2 normalized)
	Visual Page Encoder	CLIP ViT-B/32 (512-dim joint projection, 224×224 input)
	Reciprocal Rank Fusion	$k_{\text{rrf}} = 60$, equal channel weights ($w_{\text{lex}} = w_{\text{dense}} = w_{\text{vis}} = 1.0$)
	Candidate Pool Budget	24 candidate pages
	Structure Routing	BM25 weight 0.65, Dense 0.85, Visual 0.60, Outline title 0.45
Complete	High-Confidence Seeds	Top 6 pages from candidate pool
	Section Completion	3 semantic sections, 4 pages retained per section
	Dilation Radius $\delta(q)$	Radius $\delta = 1, 2, 3$ adjacent sections, boundary pages included
	Continuity Cues	across, compare, respectively, steps, process, figure, table, following
Purify	Verifier Model	GPT-5.6-Terra (temperature 0.0, max output 2,048 tokens)
	Candidate Text Excerpt	First 1,400 characters per candidate page
	Selection Bounds	Minimum 1 page, maximum 8 pages
	Fallback Coefficients	Relevance $\lambda_r = 1.00$, Coverage $\lambda_c = 0.35$, Diversity $\lambda_d = 0.15$

Multi-Channel Retrieval and Structure Routing As outlined in Table 7, the retrieval stage combines sparse lexical matching, dense semantic text embedding, and visual page representation. Candidate ranks from the three channels are first fused via Reciprocal Rank Fusion (RRF) with smoothing constant $k_{\text{rrf}} = 60$ to form an initial pool of 24 pages. In documents with structured outlines, section routing aggregates page evidence and title overlap to select the top 4 sections, capping page retention at 8 pages per section. Documents without an outline are deterministically segmented into uniform five-page blocks.

Semantic Section Completion Starting from the top 6 retrieval seed pages, completion identifies up to 3 highly correlated document sections. When linguistic cues indicate multi-page procedures or tabular continuity, the operator dynamically expands the candidate window to adjacent sections with radius $\delta(q)$, ensuring that fragmented evidence chains are bridged while adhering to the 24-page ceiling.

Verification-Aware Purification The purification operator evaluates at most 24 candidate pages using GPT-5.6-Terra. Each page is presented with its visual image and the first 1,400 characters of extracted text. The verifier filters out weakly related or duplicate pages, selecting between 1 and 8 unique pages. In cases where the verifier call encounters an unrecoverable service error, the system relies on the deterministic fallback utility score parameterized by relevance $\lambda_r = 1.00$, question coverage $\lambda_c = 0.35$, and diversity $\lambda_d = 0.15$.

A.3 System Costs, Runtime Accounting, and Latency Analysis

End-to-End Token Consumption We track token consumption and API invocation durations across all canonical evaluation runs. Table 8 reports the complete accounting for MMLongBench-Doc and LongDocURL under the optimal H3 configuration. Across the 507 benchmark questions of MMLongBench-Doc, the core pipeline (purification, answering, and judging) consumes 31.9M total tokens, averaging approximately 62,966 tokens per question. On LongDocURL, the entire evaluation across 2,322 context-bearing examples consumes 176.6M tokens, corresponding to an average footprint of 76,045 tokens per question.

Latency Distributions Table 9 details the latency profile per question across the primary processing stages. While multi-image purification requires a mean duration of approximately 40 to 41 seconds per instance due to multi-page visual inspection, downstream answer generation remains highly efficient, averaging 5.13 seconds on MMLongBench-Doc and 23.61 seconds on LongDocURL.

Monetary Cost Modeling and Accounting Transparency To facilitate reproducible cost estimation across arbitrary API providers, overall monetary expenditure can be formalized as

$$C = \frac{T_{\text{in}} - T_{\text{cache}}}{10^6} P_{\text{in}} + \frac{T_{\text{cache}}}{10^6} P_{\text{cache}} + \frac{T_{\text{out}}}{10^6} P_{\text{out}}, \tag{16}$$

Table 8: **End-to-end token consumption and cumulative API latency for DocHarness-RSI.**

Evaluation Suite / Stage	Calls	Input Tokens	Output Tokens	Total Tokens	Cumulative Latency
MMLongBench-Doc					
VLM Purification	507	27,044,183	94,772	27,183,415	20,562.24 s (5.71 h)
Frozen Answer Inference	507	1,808,753	47,752	1,856,505	2,600.51 s (0.72 h)
Official V2 Judge	507	2,863,804	20,225	2,884,029	1,544.01 s (0.43 h)
Core Pipeline Total	1,521	31,716,740	162,749	31,923,949	24,706.76 s (6.86 h)
LongDocURL					
VLM Purification	2,322	154,379,211	580,945	154,960,156	95,194.76 s (26.44 h)
Frozen Answer Inference	2,322	8,577,494	196,039	8,773,533	54,816.32 s (15.23 h)
Answer Extraction	2,322	12,646,222	196,779	12,843,001	18,667.74 s (5.19 h)
Full Suite Total	6,966	175,602,927	973,763	176,576,690	168,678.82 s (46.86 h)

Table 9: **Empirical latency distributions per question across processing stages.**

Dataset and Processing Stage	Mean (s)	Median (s)	95th Percentile (s)
MMLongBench-Doc Purification	40.56	25.00	114.25
MMLongBench-Doc Answer Generation	5.13	4.19	11.11
MMLongBench-Doc Official Judge	3.05	2.79	4.56
LongDocURL Purification	41.00	34.97	74.25
LongDocURL Answer Generation	23.61	16.99	62.26
LongDocURL Short-Answer Extraction	8.04	5.13	23.05

where T_{in}, T_{cache}, and T_{out} denote raw prompt tokens, cached prompt tokens, and generated completion tokens, respectively, and P represents the corresponding price per million tokens. Because our experimental evaluation spanned multiple concurrent relays, API keys, and tiered volume discounts, exact financial billing records vary by provider contract. Therefore, the auditable token counts reported in Table 8 serve as the invariant, hardware-independent metric of system cost.

B Qualitative Case Studies and Failure Analysis

To mechanistically understand how each operator in the Retrieve-Complete-Purify pipeline alters context quality and downstream reasoning, we examine three representative examples from MMLongBench-Doc. In each case, the target operator converts a failure into a correct prediction. Ground-truth evidence page annotations are utilized solely for post-hoc verification and were never exposed to the harness during retrieval, completion, or purification. Table 10 provides an overview of the three qualitative cases.

Table 10: Summary of qualitative case studies across the three evolutionary stages.

Case	Failure Mode	Target Operator	Evidence Recall	Initial Answer	Evolved Answer
1	Evidence Miss	Multimodal Retrieval (H_1)	$0\% \rightarrow 100\%$	Not answerable	76%
2	Incompleteness	Semantic Completion (H_2)	$50\% \rightarrow 100\%$	7.256%	7.273% (0.073)
3	Context Dilution	Page Purification (H_3)	$100\% \rightarrow 100\%$	5	2

B.1 Case 1 – Repairing Evidence Miss via Multimodal Retrieval

As shown in Figure 5, the question asks for the battery percentage displayed inside Figure 107. The baseline text retriever H_0 selects eight candidate pages based purely on lexical matching, but completely misses physical page 137 where the figure is located. Because the target information is encapsulated within a graphical UI screenshot rather than machine-readable OCR text, the frozen VLM strictly outputs "Not answerable" under H_0.

By integrating visual CLIP embeddings and document outline routing, H_1 successfully identifies page 137 as a top-ranked candidate. With full evidence recall restored from 0% to 100%, the frozen answerer directly reads the battery icon and outputs the correct reference value, 76%.

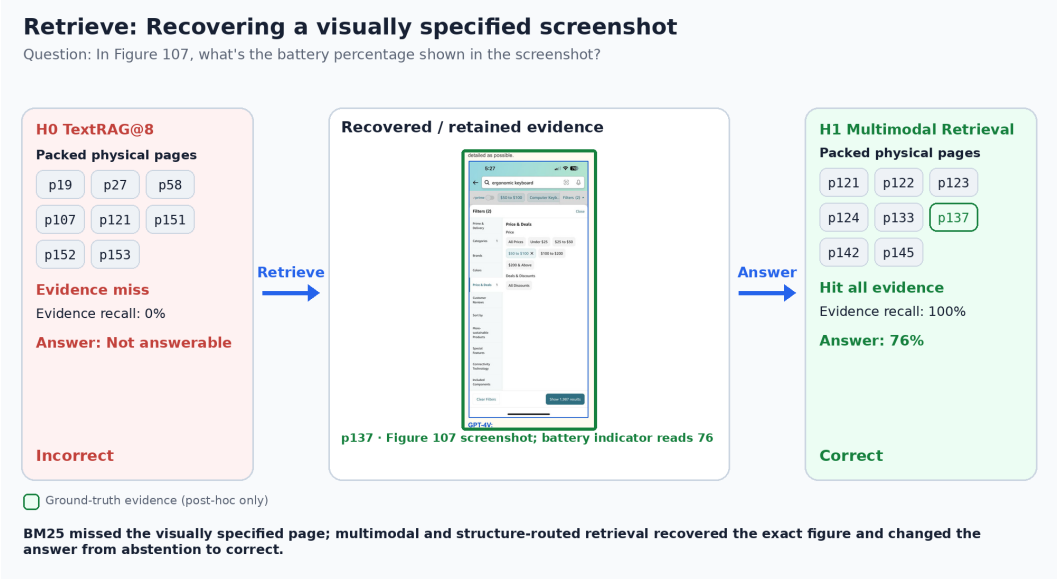

Figure 5: **Case 1 – Mitigating Evidence Miss.** Text-only BM25 (H_0) misses the visually specified screenshot on physical page 137, resulting in an abstention. Multimodal retrieval (H_1) captures the target figure and produces the correct answer.

B.2 Case 2 – Repairing Cross-page Incompleteness via Semantic Completion

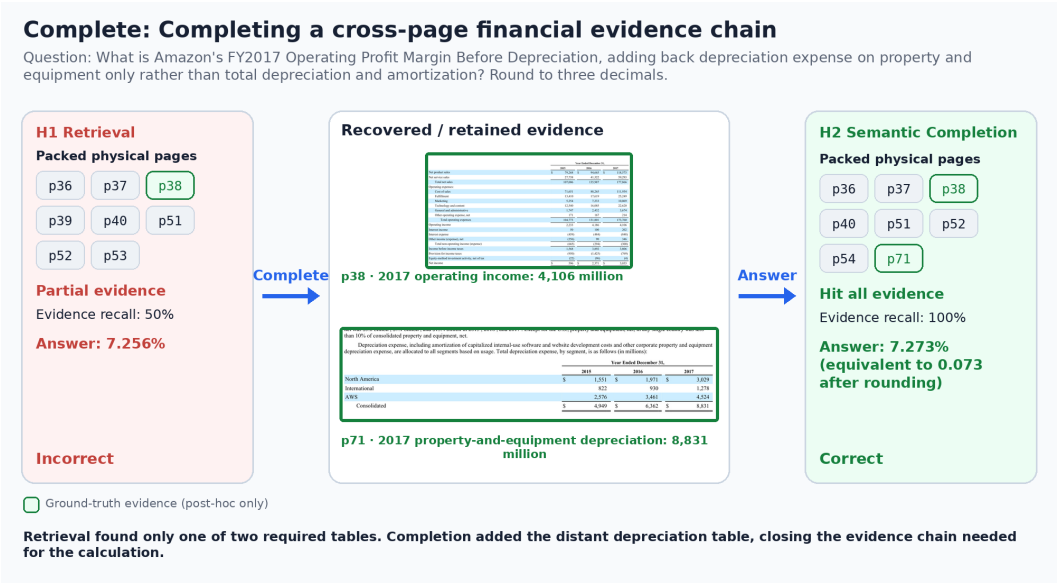

Figure 6: **Case 2 – Resolving Cross-page Incompleteness.** Single-page retrieval (H_1) captures only operating income on page 38, leading to an arithmetic calculation error. Semantic Section Completion (H_2) bridges the 33-page gap to recover depreciation on page 71.

Figure 6 illustrates a complex multi-hop financial query requiring the computation of Amazon's FY2017 operating profit margin before property-and-equipment depreciation. The calculation requires two separate values distributed across non-adjacent document sections, namely operating income located on physical page 38 and property-and-equipment depreciation located 33 pages later on page 71.

The multi-channel retrieval stage H_1 successfully identifies the primary operating income table on page 38, but fails to surface page 71 due to lower standalone relevance, achieving only 50% evidence recall. Consequently, the answerer attempts calculation using incomplete context and outputs an erroneous ratio of 7.256%. Semantic Section Completion in H_2 detects the cross-page continuity cue "property and equipment" and dynamically expands candidate coverage to the adjacent accounting disclosures. This successfully recovers page 71, closing the multi-page evidence chain and enabling the model to calculate the correct value of 7.273% (rounding to the ground-truth answer 0.073).

B.3 Case 3 – Eliminating Context Dilution via Evidence Purification

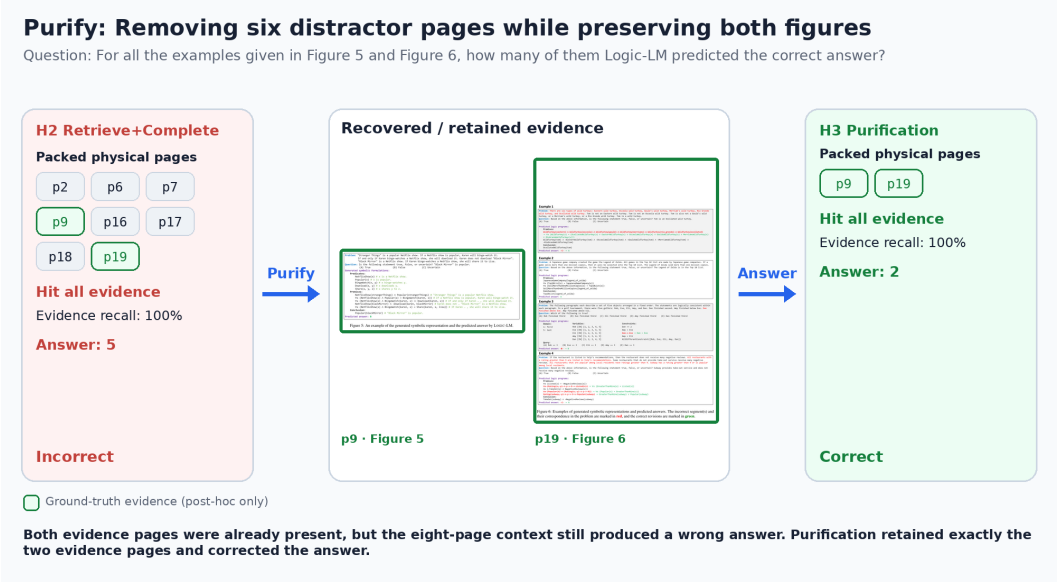

Figure 7: **Case 3 – Eliminating Context Dilution.** Both ground-truth evidence pages (p9 and p19) are present under H_2, but six uninformative distractor pages induce an erroneous answer. Verification-aware purification (H_3) strips all distractors, retaining only the two essential pages and correcting the answer.

Figure 7 presents a counting question querying how many examples across Figure 5 and Figure 6 were correctly predicted by Logic-LM. Under stage H_2, both necessary pages (page 9 containing Figure 5 and page 19 containing Figure 6) are already included within the candidate pool, achieving 100% evidence recall. However, these two vital pages are packed alongside six irrelevant distractor pages (pages 2, 6, 7, 16, 17, and 18). Despite having full evidence in view, the frozen VLM is distracted by the crowded multi-page context and produces an incorrect count of 5.

In stage H_3, the verification-aware purification operator inspects the candidate pool, filters out all six non-essential pages, and compresses the final context from eight pages down to exactly the two ground-truth pages. With context dilution eliminated and evidence purity reaching 100%, the frozen model focuses precisely on the two target figures and outputs the correct prediction of 2. This case provides direct qualitative confirmation of the findings in Table 3, proving that increasing context purity is just as critical as maximizing evidence recall.

B.4 Diagnostic Interpretation and Methodology Boundaries

These qualitative traces serve to illustrate the specific functional mechanisms of the RCP operators rather than aggregate causality. In particular, Context Dilution represents an operational failure proxy where the model errs despite complete evidence visibility. While individual errors may occasionally stem from intrinsic VLM reasoning limitations or resolution thresholds, the consistent qualitative progression across all three cases mirrors the macro-level improvements observed across our full experimental benchmark.

C LIMITATIONS

While DocHarness-RSI demonstrates consistent improvements by optimizing executable context construction around frozen vision-language models, we identify two primary operational limitations.

Semantic Relevance without Decisive Evidence A persistent challenge in document context construction arises when candidate pages exhibit strong topical alignment with the question, yet fail to contain the decisive factual statement. Multi-channel retrieval and section routing rely on lexical, dense semantic, and visual similarity. In lengthy technical reports or multi-page financial filings, extensive sections frequently discuss the queried topic or system entities at length, accumulating high similarity scores. However, the specific page containing the critical summary sentence, definitive number, or key table row may have lower overall text overlap and get omitted.

For instance, in a query asking how many modules use an LLM backbone, the retrieval stage selected multiple pages that elaborated extensively on module implementations and prompt workflows. Although these pages were topically relevant, the explicit declaration of the total module count resided on a different page that was overlooked. This indicates that general semantic proximity does not guarantee the inclusion of the single decisive evidence token, highlighting the need for more fine-grained, claim-level evidence routing.

Downstream Reasoning and Visual Errors Under Complete Evidence DocHarness-RSI optimizes which multimodal pages are exposed to the answerer within a bounded page budget, but it intentionally leaves the underlying model parameters and decoding mechanisms frozen. Consequently, providing complete, high-purity evidence does not guarantee a correct prediction. In practice, a substantial portion of remaining errors occurs on instances where all ground-truth evidence pages are successfully retrieved and purified, yet the model fails to produce the correct answer.

These failures stem from the intrinsic perceptual and cognitive boundaries of the frozen downstream model. Complex long-document tasks often require fine-grained visual reading over dense charts, parsing small font annotations, or performing multi-step numerical calculations. For example, in a demographic analysis query requiring the percentage-point difference between age groups across two countries, our purification stage correctly isolated the single ground-truth page with full evidence recall and zero context dilution. Nonetheless, the frozen answerer miscalculated the arithmetic difference from the chart values, outputting an erroneous numerical answer. This confirms that context optimization cannot overcome the foundational perception and reasoning limits of the underlying model, marking the explicit boundary between context engineering and model capacity.