Doc-V^*: Coarse-to-Fine Interactive Visual Reasoning for Multi-Page Document VQA

Yuanlei Zheng[1]*, Pei Fu[2]*, Hang Li[2], Ziyang Wang[1], Yuyi Zhang[1], Wenyu Ruan [1], Xiaojin Zhang[3], Zhongyu Wei[4], Zhenbo Luo[2]†, Jian Luan[2], Wei Chen[1]†, Xiang Bai[1]

[1]School of Software Engineering, Huazhong University of Science and Technology,
[2]MiLM Plus, Xiaomi Inc.,
[3]School of Computer Science and Technology, Huazhong University of Science and Technology,
[4]School of Data Science, Fudan University

Abstract

Multi-page Document Visual Question Answering requires reasoning over semantics, layouts, and visual elements in long, visually dense documents. Existing OCR-free methods face a trade-off between capacity and precision: end-to-end models scale poorly with document length, while visual retrieval-based pipelines are brittle and passive. We propose **Doc-V^***, an **OCR-free agentic** framework that casts multi-page DocVQA as sequential evidence aggregation. **Doc-V^*** begins with a thumbnail overview, then actively navigates via semantic retrieval and targeted page fetching, and aggregates evidence in a structured working memory for grounded reasoning. Trained by imitation learning from expert trajectories and further optimized with Group Relative Policy Optimization, **Doc-V^*** balances answer accuracy with evidence-seeking efficiency. Across five benchmarks, **Doc-V^*** outperforms open-source baselines and approaches proprietary models, improving out-of-domain performance by up to **47.9%** over RAG baseline. Other results reveal effective evidence aggregation with selective attention, not increased input pages.

1 Introduction

Understanding multi-page, visually rich documents—such as academic papers, financial reports, and industrial manuals—remains a core challenge in *Document Visual Question Answering* (DocVQA) (Mathew et al., 2021; Tito et al., 2023). Unlike plain text, such documents convey information through a complex interplay of textual semantics, spatial layouts, and visual elements (e.g., tables and figures) (Ding et al., 2025). Conventional **OCR-based** pipelines linearize document images into text before reasoning (Memon et al., 2020; Wang et al., 2024; Appalaraju et al., 2021), but inevitably lose fine-grained layout cues and suffer from cascading OCR errors. Recent **OCR-free** or **pure-vision** approaches instead model documents directly as images using multimodal large language models (MLLMs) (Lee et al., 2023; Kim et al., 2022; Liu et al., 2024b), enabling joint visual–semantic reasoning and improved robustness.

However, existing pure-vision methods face a fundamental trade-off between *capacity* and *precision*. **End-to-end** models process entire documents as long image sequences (Zhu et al., 2025; Hu et al., 2025; Bai et al., 2025), but scale poorly to long documents due to quadratic attention cost, context length limits, and the "*lost-in-the-middle*" effect (Liu et al., 2024a). In contrast, visual retrieval-augmented generation (**RAG**) systems reduce noise by retrieving top-k pages before generation (Cho et al., 2024; Faysse et al., 2025; Song et al., 2025), yet suffer from retrieval errors, sensitivity to hyperparameters, and limited multi-hop reasoning. Critically, both paradigms remain *passive*: they process a fixed input without adapting their strategy as new evidence emerges.

We argue that this limitation arises from a mismatch with human document-reading behavior. Guided by *Active Vision Theory* (Aloimonos et al., 1988), human experts treat perception as a goal-directed process: they first obtain a global structural overview, then iteratively seek, verify, and integrate evidence while maintaining working memory. Inspired by this cognitive process, we propose **Doc-V^***, an **OCR-free agentic framework** that formulates multi-page DocVQA as a ***sequential evidence aggregation process***. **Doc-V^*** begins with a *Global Thumbnail Overview* that provides a low-cost structural prior, and then alternates between *structured visual reasoning* and *document navigation actions*, including semantic retrieval and targeted page fetching. This interactive reasoning allows the agent to **active perception** and **piece together discontinuous visual evidence** before answering. Figure 1 shows the agent workflow of

*Equal contribution
†Corresponding author

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 45901–45923
July 2-7, 2026

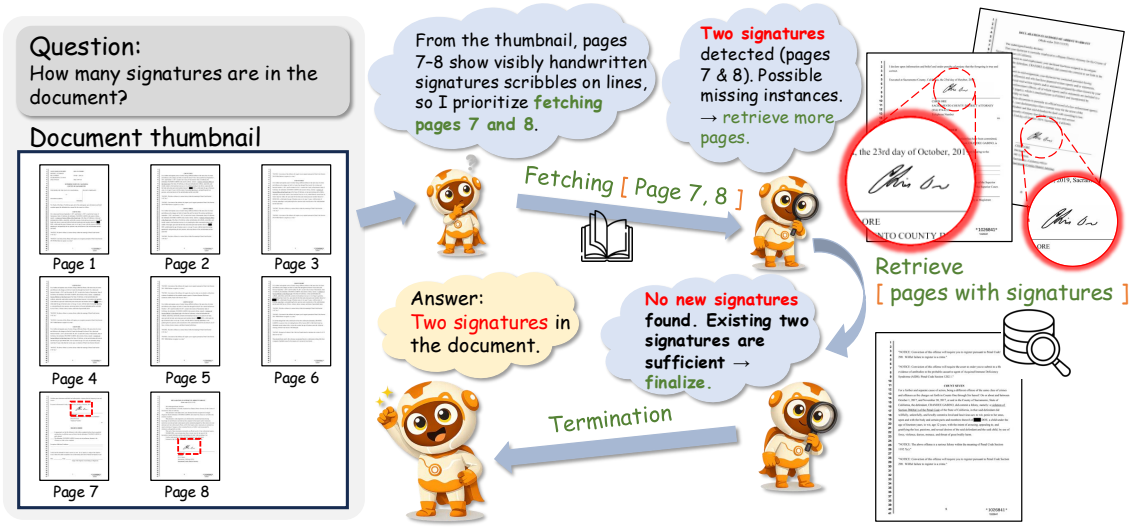


Figure 1: **The Doc-V^* agent workflow for multi-page document VQA.** It adopts an *active perception* paradigm by planning from a global thumbnail view and iteratively deciding when to fetch high-resolution pages or perform semantic searches, aggregating evidence in a structured working memory for grounded answering.

Doc-V^*.

To train **Doc-V^***, we adopt a two-stage optimization strategy. We first perform supervised fine-tuning using high-quality interaction trajectories synthesized by GPT-4o, providing a strong cold start. We then apply Group Relative Policy Optimization (GRPO) (Guo et al., 2025) to jointly optimize answer accuracy and evidence-seeking efficiency through reward signals that account for answer quality, evidence discovery, and format compliance. Extensive experiments on five benchmarks demonstrate that **Doc-V^*** consistently outperforms existing open-source baselines and rivals proprietary models like **GPT-4o**, particularly in out-of-domain settings where it achieves up to a **47.9% improvement** over static RAG baselines, as well as **robustness** under variations in retrieval tools and hyperparameters. We also demonstrate that long-document understanding hinges on **effective aggregation of evidence** with selective attention rather than **sheer input pages**, which is crucial to the success of **Doc-V^***.

2 Related Work

Visual Document Question Answering (DocVQA) has progressed from single-page inputs to long and multi-page documents, driven by the increasing demand for handling complex real-world document understanding scenarios. Existing methods mainly follow two paradigms: 1) **OCR-based DocVQA** OCR-based approaches first extract textual and layout structures via OCR and document parsing, followed by reasoning over structured representations (Tito et al., 2023; Zhang et al., 2024; Luo et al., 2024; Fujitake, 2024; Li et al., 2024; Duan et al., 2025; Nacson et al., 2025; Xie et al., 2024). While effective on clean and well-formatted documents, these pipelines inevitably suffer from cascading OCR and layout errors and generalize poorly to noisy or out-of-domain scenarios; 2) **OCR-free Pure-Vision DocVQA** Recent OCR-free methods leverage large vision–language models to reason directly over document images, preserving rich visual and spatial cues. However, scaling to long documents remains challenging. Existing approaches include: (i) *end-to-end* models that process all pages jointly (Hu et al., 2025; Zhu et al., 2025), which scale poorly with document length, as computational cost and memory consumption grow rapidly with the number of pages; (ii) *retrieval-based* methods that select top-k pages before generation (Cho et al., 2024; Yu et al., 2024; Chen et al., 2024; Tanaka et al., 2025; Wang et al., 2025; Wu et al., 2025; Shi et al., 2025), improving efficiency but remaining sensitive to retrieval errors and fixed hyperparameters; and (iii) *agent-based* systems that iteratively explore documents (Xu et al., 2025; Yang et al., 2025; Yue et al., 2025), which introduce interaction at the cost of increased complexity. In contrast, our method formulates DocVQA as a sequential evidence aggregation process, enabling a single OCR-free agent to actively and efficiently aggregate visual evidence over long documents.

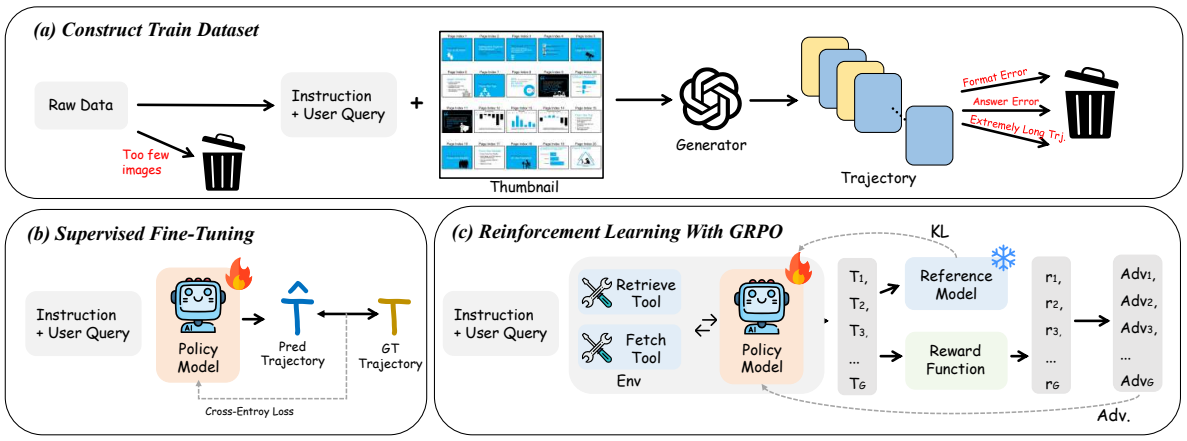


Figure 2: **Overview of the training pipeline for Doc-V^*.** *(a) Training data construction.* Documents and queries are paired to generate thumbnail-guided reasoning trajectories, followed by quality filtering. *(b) Supervised fine-tuning (SFT). (c) Reinforcement learning with GRPO.*

3 Method

3.1 Formulation and Cognitive Motivation

Faced with lengthy, unfamiliar documents, human experts exhibit pronounced ***goal-directedness*** and ***proactivity*** rather than reading cover-to-cover: they navigate using structural cues and keyword-like searches, and iteratively update their strategy as evidence is found. This behavior is consistent with ***Active Vision*** (Aloimonos et al., 1988), which views perception as goal-directed sampling to reduce uncertainty, and ***Resource-Rational Cognition*** (Lieder and Griffiths, 2020), which trades off information gain against processing costs. Motivated by these principles, we propose **Doc-V^***, formulating *Multi-page Document VQA* as a *Sequential Decision Process*: given a document $\mathcal{D} = \{p_1, \ldots, p_N\}$ and a question Q, an **OCR-free** MLLM-based agent π_θ interacts with the document environment for up to T steps. At step t, the agent receives its observation O_t, performs reasoning, and selects an action $a_t \in \mathcal{A}$; the environment then returns feedback E_{t+1}, which is incorporated into the next observation O_{t+1}. This closed-loop formulation enables **selective evidence acquisition** and the **integration of scattered visual cues into a coherent reasoning chain**.

3.2 Environment Design

Document Visual Representation Our agent is built upon the Qwen-2.5-VL (Bai et al., 2025) architecture, which comprises a visual encoder $\mathcal{V}$ (adopting ViT (Dosovitskiy, 2020) architecture), a multi-layer perceptron projection module $\mathcal{M}$, and a large language model backbone $\mathcal{L}$. We pre-compute and cache the visual tokens $\mathbf{v}_i = \mathcal{M}(\mathcal{V}(p_i)) \in \mathbb{R}^{L_i \times d}$ for all pages $\{p_i\}_{i=1}^N$ within D at their **native high resolution** (capped at 1024×768), where L_i is the token count and d is the hidden dimension. Crucially, these visual tokens are not fed to the agent all at once but are dynamically requested by the agent based on its decisions.

Initial Observation Before interaction begins, we design a ***Global Thumbnail Overview*** $\tilde{D}$ for the document, inspired by the human behavior of first "rapidly flipping through pages" to grasp the overall structure when browsing a document. Concretely, we partition the document into groups of pages, resize each page to a thumbnail (256×256), reorganize each group into a grid image and annotate each thumbnail with its ***absolute page number***. While body text details become indiscernible at this resolution, rich structural information remains visible like *document type*, *section layout*, *chart distribution* and *larger-font titles*. This coarse-grained global perception provides considerable navigational priors for subsequent fine-grained exploration. Formally, the initial input fed to the agent is denoted as: $O_o = \{Q, \tilde{D}\}$, where $\tilde{D}$ possibly consisting of one or multiple grid images.

Please refer to Appendix A for the detail of the *Environment Design*.

3.3 Action Space

We define three types of atomic actions for the agent that capture common human document-reading behaviors.

1. Retrieval Action The retrieval action is in-

tended to approximate the "Ctrl+F search within document" behavior, but at the level of page images. To trigger this, the agent need emits a structured command: **"<retrieval_page>**q_t**"**, which signifies a decision to retrieve document images using the textual query q_t. The query can differ from the original question Q, allowing iterative refinement as evidence accumulates. The environment then calls an external multimodal retriever (e.g., ColQwen (Faysse et al., 2024)), ranking pages in $\mathcal{D} \setminus \mathcal{P}_{\text{visited}}$ and returns the top-k **unvisited pages**, where $\mathcal{P}_{\text{visited}}$ is an external variable that maintains a set of visited pages to avoid redundancy.

2. Fetch Action The fetch action requests specific pages by absolute indices via the command **"<fetch_page>**$[i_1, \ldots, i_m]$**"**. Upon receiving this, the environment parses the index list and retrieves the exact pages specified. This action facilitates several common navigation strategies: 1) direct page fetching based on visual features observed in the thumbnail view (e.g., TOC and chart positions); 2) needing to view adjacent pages before or after the current page for complete context after reading a certain page; 3) responding to page numbers explicitly mentioned in the user question (e.g., *"How many baselines are there in the table on page three?"*).

For both actions, the environment returns the **cached high-resolution visual tokens** of the requested pages. Each page's visual tokens are prefixed with a textual page number identifier (e.g., **"Page 5:"**) to ensure the agent can correctly associate the visual content with its specific page number. If a requested page has already been visited, the environment returns a **text reminder** instead of re-inputting the visual tokens. We denote E_t as the *environment feedback* at interaction step $t \geq 1$.

3. Answer Action When the agent determines that sufficient evidence has been gathered, it terminates the interaction by executing the answer action by generating **"<answer>**y**"**, where y is the final answer string.

3.4 Structured Visual Reasoning

To make the agent's decision process explicit and auditable, we enforce a fixed *think-acting* interaction protocol, a ReAct (Yao et al., 2022) reasoning style with visual feedback. At each step, the agent's output must follow the format: **"<think>**$\cdots$**</think><action>**$\cdots$**</action>"**, where **<action>** instantiates exactly one action from §3.3 with the required arguments.

We further structure **<think>** into 3 blocks, with a slight distinction between the first turn and later turns. At turn t=0, given the initial observation, i.e., document thumbnails with question, **<think>** consists of: 1) **<analysis>**: a coarse document-level inspection from thumbnails, identifying likely question-relevant regions/pages and key visual cues; 2) **<plan>**: an explicit subgoal decomposition and an interaction plan, which guides subsequent actions under a limited step budget; 3) **<summary>**: a compact summary of the initial inspection and plan. At turns t>0, given newly returned high-resolution pages, **<think>** consists of: 1) **<analysis>**: Page-by-page content analysis of newly returned pages, evaluating each page's relevance to the user question, determining whether the evidence is sufficient to answer, and deciding on the next optimal action that can reduce uncertainty; 2) **<relevant_pages>**: Explicitly outputs the list of page numbers judged to be relevant among the pages returned in the current turn. This component forces the agent to make binary relevance judgments, facilitating subsequent reward signal computation and model evaluation; 3) **<summary>** An incremental information summary for the current turn, which together with historical summaries constitutes the agent's ***Working Memory***.

As interaction proceeds, image-text interleaved tokens accumulate and pages may arrive out of order, which can cause the agent to forget and drift (e.g., forgetting resolved sub-questions or repeatedly fetching a certain page). To mitigate this, we feed the agent an **augmented observation** $O_t = E_t \cup \{W_t\}, t \geq 1$, where the ***Working Memory*** $W_t = \text{Concat}(S_0, \ldots, S_{t-1})$ concatenates all previous **<summary>** within **<think>**. Please refer to Appendix B for the detail of the *Agent Environment Interaction Protocol*.

3.5 Training

We adopt a standard two-stage training pipeline to obtain an agent that is both tool-competent and exploration-efficient under a bounded interaction budget. First, we perform supervised fine-tuning with a cross-entropy objective on distilled closed-loop interaction trajectories, where a strong teacher interacts with the real environment and we compute loss only on agent-generated tokens; we further filter trajectories by format validity, answer correctness, and evidence-page sanity, yielding **9,019** high-quality trajectories constructed from MP-DocVQA and DUDE. Second, we apply GRPO reinforce-

ment learning using only outcome supervision: we filter **2,048** non-overlapping training examples, stratify them into easy/medium/hard buckets estimated by the SFT policy via multiple rollouts, and train the agent by sampling groups of trajectories in the same closed-loop environment and optimizing a weighted reward that combines answer correctness, evidence retrieval quality, and format validity. Full training details are provided in Appendix C.

4 Experiments

4.1 Experimental Setup

Datasets Our raw training data is sourced from **MP-DocVQA** (Tito et al., 2023) and **DUDE** (Van Landeghem et al., 2023). Evaluation is conducted under two settings. (1) *In-Domain* evaluation is performed on the test splits of MP-DocVQA and DUDE. (2) *Out-of-Domain (OOD)* evaluation is carried out on three challenging benchmarks: **SlideVQA** (Tanaka et al., 2023), **LongDocURL** (Deng et al., 2025), and **MMLongBench-Doc** (Ma et al., 2024). These benchmarks cover diverse document types and reasoning challenges, enabling a comprehensive evaluation of generalization beyond the training domain. Detailed statistics and dataset characteristics are provided in Appendix E.

Evaluation Metrics All methods are evaluated using the *official metrics and evaluation protocols* of each benchmark. Specifically, we report ANLS for DUDE and MPDocVQA, F1 score for SlideVQA, and Accuracy for MMLongBench-Doc and LongDocURL.

Agent and Environment Setup Our agent is initialized from **Qwen-2.5-VL-7B-Instruct** (Bai et al., 2025). For the `retrieval_page`, we employ **ColQwen** (Faysse et al., 2025) as the external retriever. Retrieval budget is dynamically set to $k = \min(\lceil N/10 \rceil, 4)$ to balance information coverage and context efficiency, and the maximum interaction horizon is fixed to $T = 8$ steps during both training and inference. The optimization objective incorporates a composite reward function balancing answer correctness ($\omega_{\text{ans}} = 0.6$), evidence recall ($\omega_{\text{evi}} = 0.3$), and structural validity ($\omega_{\text{struct}} = 0.1$). Specific training hyperparameters and further implementation details are provided in Appendix D.

4.2 Main Results

We compare **Doc-V^*** with a broad suite of baselines spanning three paradigms: (i) **End-to-End (E2E)** models including HiVT5 (Tito et al., 2023), mPLUG-DocOwl2 (Hu et al., 2025), Docopilot (Duan et al., 2025), DocVLM (Nacson et al., 2025), and InternVL3 (Zhu et al., 2025); (ii) **Retrieval-Augmented Generation (RAG)** methods including CREAM (Zhang et al., 2024), M3DocRAG (Cho et al., 2024), VisRAG (Yu et al., 2024), SV-RAG (Chen et al., 2024), VDocRAG (Tanaka et al., 2025), MoLoRAG (Wu et al., 2025), and URaG (Shi et al., 2025); and (iii) **Agent-based** approaches including VRAG-RL (Wang et al., 2025) and CogDoc (Xu et al., 2025). We additionally report closed-source systems (Gemini-1.5-Pro (Team et al., 2024), GPT-4o mini, GPT-4o (Hurst et al., 2024), GPT-4.1, and Claude-3.7-Sonnet) as reference points, and include Qwen2.5-VL (Bai et al., 2025) along with its RAG Top-5 variant as direct backbone baselines. Detailed descriptions of these baseline methods are provided in Appendix F. As shown in Table 1, our GRPO-enhanced model achieves the best overall performance among open-source methods on four of five benchmarks, while remaining competitive on the remaining benchmark.

On the In-domain benchmarks (DUDE and MP-DocVQA), **Doc-V^*** achieves strong accuracy. On DUDE, it reaches **64.5** ANLS, **outperforming all open-source baselines** and also **surpassing some closed-source models reported**, including GPT-4o (54.1) and Claude-3.7-Sonnet (58.1). On MP-DocVQA, our method attains **86.2** ANLS, remaining highly competitive with URaG (88.2).

On the Out-of-Domain benchmarks, **Doc-V^*** shows clear generalization advantages. On SlideVQA, our model achieves 77.2 F1, outperforming SlideVQA-trained baselines CogDoc (67.9). It also sets new open-source highs on long-context benchmarks, scoring 42.1 accuracy on MMLongBench-Doc and 56.3 accuracy on LongDocURL. These results indicate that **Doc-V^*** maintains robust long-context evidence localization and aggregation ability when transferring to diverse document domains and substantially longer inputs.

To isolate the effect of the agentic policy and GRPO training, we compare **Doc-V^*** against Qwen2.5-VL and Qwen2.5-VL (RAG Top-5) under the same 7B scale. Static retrieval is beneficial—Qwen2.5-VL (RAG Top-5) improves

Table 1: **Comparison of different methods on five long-context and multi-page document understanding benchmarks.** The results are reported on **DUDE** (ANLS), **MPDocVQA** (ANLS), **SlideVQA** (F1), **MMLongBench-Doc** (Acc), and **LongDocURL** (Acc). "Param." denotes the parameter scale (referring specifically to the **Generator** for RAG methods). "Backbone" indicates the underlying LLM or LVLM used. "Paradigm" categorizes methods into End-to-End (**E2E**), Retrieval-Augmented Generation (**RAG**), or **Agent**. The best and second-best results **among Open Source methods** are highlighted in **bold** and underlined, respectively. Scores marked with an asterisk (*) indicate that the method's backbone was supervised fine-tuned on that specific benchmark's training set. Red subscripts in parentheses indicate the absolute performance gain over the baseline (Qwen2.5-VL).

Method	Backbone	Param	Paradigm	DUDE (ANLS)	MPDocVQA (ANLS)	SlideVQA (F1)	MMLong. (Acc)	LongDoc. (Acc)
Closed Source								
Gemini-1.5-Pro	-	-	E2E	46.0	-	-	28.2	50.9
GPT-4o mini	-	-	E2E	46.5	-	60.7	28.6	-
GPT-4o	-	-	E2E	54.1	67.4	65.8	42.8	64.5
GPT-4.1	-	-	E2E	50.2	-	74.7	45.6	-
Claude-3.7-Sonnet	-	-	E2E	58.1	-	76.3	33.9	-
Open Source								
HiVT5 *(PR)*	DiT / T5	0.3B	E2E	23.1	62.0*	-	-	-
CREAM *(ACM MM'24)*	Pix2Struct / LLaMa2	7B	RAG	52.5*	74.3*	-	-	-
mPLUG-DocOwl2 *(ACL'25)*	ViT / LLaMa	8B	E2E	46.8*	69.4*	27.8	13.4	5.3
M3DocRAG *(arXiv'24)*	Qwen2-VL	7B	RAG	39.5	84.4	55.7	21.0	35.1
VisRAG *(ICLR'25)*	MiniCPM-V 2.6	8B	RAG	43.1	-	52.4	18.8	41.9
SV-RAG *(ICLR'25)*	InternVL2	4B	RAG	45.0	71.0	34.3*	23.0	-
VDocRAG *(CVPR'25)*	Phi3-Vision	4B	RAG	44.0*	62.6	42.0	18.4	39.8
Docopilot *(CVPR'25)*	InternVL2	8B	E2E	40.7*	81.3*	43.1	28.8	-
DocVLM *(CVPR'25)*	Qwen2-VL	7B	E2E	47.4	84.5	-	-	-
InternVL3 *(arXiv'25)*	InternViT / Qwen2.5	8B	E2E	47.4	80.8	64.4	24.1	38.7
VRAG-RL *(NeurIPS'25)*	Qwen2.5-VL	7B	Agent	-	-	-	26.6	44.9
MoLoRAG *(EMNLP'25)*	Qwen2.5-VL	7B	RAG	-	-	-	41.0	51.9
CogDoc *(arXiv'25)*	Qwen2.5-VL	7B	Agent	46.2*	75.0	67.9*	33.0	-
URaG *(AAAI'26)*	Qwen2.5-VL	7B	RAG	57.6*	**88.2***	-	33.8	52.2
Ours								
Qwen2.5-VL (Baseline)	Qwen2.5-VL	7B	E2E	51.9	75.2	55.2	28.0	32.9
Qwen2.5-VL (RAG Top-5)	Qwen2.5-VL	7B	RAG	52.2 (+0.3)	77.4 (+2.2)	62.9 (+7.7)	36.1 (+8.1)	37.8 (+4.9)
Doc-V^* (SFT)	Qwen2.5-VL	7B	Agent	58.1 (+6.2)	81.3 (+6.1)	73.8 (+18.6)	39.8 (+11.8)	53.0 (+20.1)
Doc-V^* (GRPO)	Qwen2.5-VL	7B	Agent	**64.5** (+12.6)	86.2 (+11.0)	**77.2** (+22.0)	**42.1** (+14.1)	**56.3** (+23.4)

over the vanilla backbone, e.g., 28.0 → 36.1 on MMLongBench-Doc and 32.9 → 37.8 on LongDocURL. Nevertheless, our proposed method yields substantially larger gains at the same parameter scale, improving over RAG Top-5 by +12.3 on DUDE (52.2 → 64.5) and +18.5 on LongDocURL (37.8 → 56.3). These results demonstrate that optimizing a multi-step evidence-seeking policy via GRPO offers superior robustness compared to fixed top-k retrieval, allowing small open-source models to rival powerful closed-source models in complex document understanding.

4.3 Analysis of Page-Level Retrieval

Figure 3 analyzes the trade-off between the average number of input pages and both page-level evidence quality and downstream task performance. This analysis directly probes how different methods handle evidence under constrained budgets.

For **multimodal RAG**, increasing the number of retrieved pages exhibits a characteristic non-monotonic trend: performance initially improves as more relevant pages are included, but degrades once additional pages introduce noise. This behavior highlights two structural limitations of RAG-style pipelines. First, performance is highly sensitive to the choice of **Top-K**. Second, evidence selection and reasoning are loosely coupled—the generator must attend over a fixed, noisy context without explicit mechanisms for evidence validation or revision.

In contrast, **Doc-V^*** frames long-document understanding as a **progressive evidence aggregation process**. Instead of consuming all pages at once, the agent incrementally explores the document, extracts candidate evidence, and explicitly decides which pages are relevant at each step. This difference is reflected in the **Page-F1 metric**, which measures the alignment between the pages ultimately selected by the model and the ground-truth evidence pages.

Under comparable average input budgets, **Doc-V^*** consistently achieves substantially higher Page-

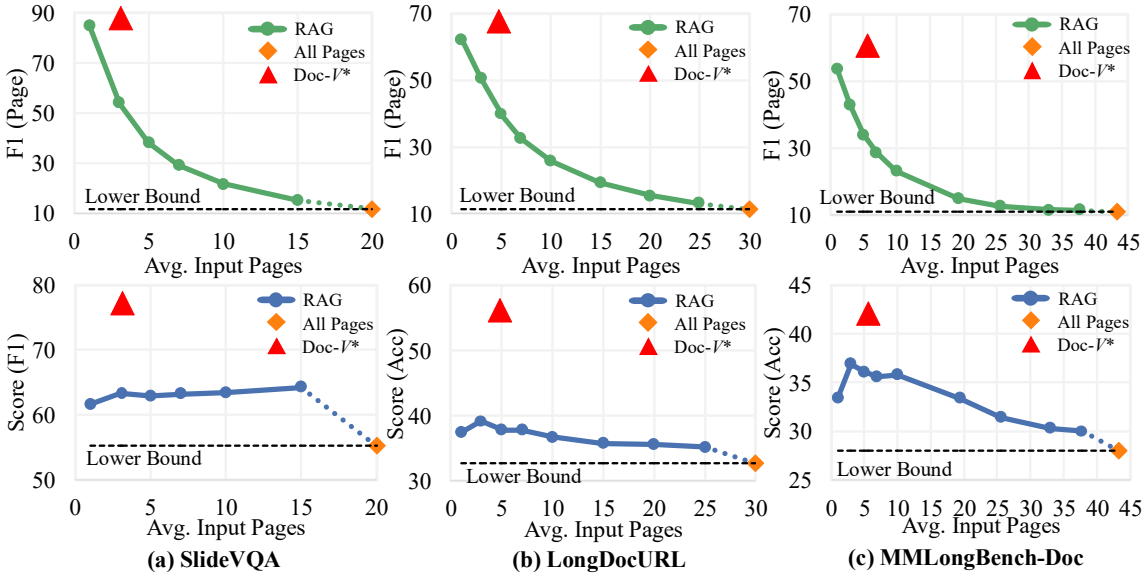


Figure 3: Efficiency–effectiveness trade-off across **SlideVQA**, **LongDocURL**, and **MMLongBench-Doc**.The top row reports Page-F1, measuring the quality of page selection under different input budgets, while the bottom row shows downstream task performance. **For Doc-V^***, Page-F1 is computed based on the pages that the model explicitly predicts as relevant, i.e., the model outputs a set of `relevant_pages`, which are then compared against the ground-truth evidence pages to compute F1.

F1 than RAG across SlideVQA, LongDocURL, and MMLongBench-Doc. Importantly, this improvement does not arise from retrieving more pages, but from **selectively consolidating evidence across multiple interaction steps**. Early observations guide hypothesis formation, while later page accesses serve to verify, refine, or reject these hypotheses.

These results suggest that **long-document understanding is not limited by insufficient context, but by the model's ability to organize and integrate evidence**. Revisiting the behavior of multimodal RAG, increasing the number of input pages primarily amplifies irrelevant or weakly related signals, while lacking explicit mechanisms for evidence consolidation. As a result, evidence becomes diluted rather than reinforced, leading to degraded reasoning performance.

4.4 Robustness Analysis

In this section, we analyze the robustness of our framework regarding the number of reasoning steps and the efficiency trade-off compared to traditional retrieval methods. More analysis see Appendix G

Impact of Document Length Figure 4 shows performance across different document length ranges. Both *All Pages* and *RAG* exhibit a clear performance degradation as document length increases, whereas **Doc-V^*** maintains consistently strong results across all ranges. Both *All Pages* and *RAG* suffer from substantial performance degradation as document length increases, while **Doc-V^*** remains consistently strong. In the longest-document regime ($>$ 80 pages), **Doc-V^*** outperforms RAG by **31.7%** (40.7 vs. 30.9) and exceeds the *All Pages* setting by a large margin of **85.8%** (40.7 vs. 21.9), demonstrating its **effectiveness** and **robustness** for long-document understanding.

Table 2: **Comparison of different retrievers on MMLongBench-Doc.**

Retriever	Model	Avg. Pages	Page-F1	Overall	SIN	MUL	UNA
ColQwen	Qwen2.5-VL	6.0	30.9	35.5	37.0	13.4	**70.4**
	Doc-V^*	5.6	**49.7**	**42.1**	**54.6**	**23.5**	45.7
BGE-Large	Qwen2.5-VL	9.0	17.6	33.0	31.2	9.8	**77.1**
	Doc-V^*	8.4	**34.0**	**36.3**	**45.7**	**18.5**	45.7
BM25	Qwen2.5-VL	10.0	20.5	32.9	32.7	11.3	**71.3**
	Doc-V^*	9.2	**36.8**	**37.5**	**48.4**	**20.4**	43.0

Efficiency and Cost To evaluate the efficiency and computational cost of different document processing strategies, a subset of samples with long documents is randomly selected for analysis. These samples are characterized by a large number of pages, with an average document length of **107.3** pages, which provides a representative setting for assessing scalability under realistic long-document scenarios. Figure 5 presents a comparative analysis of inference latency and GPU memory consumption across different methods. The results indicate

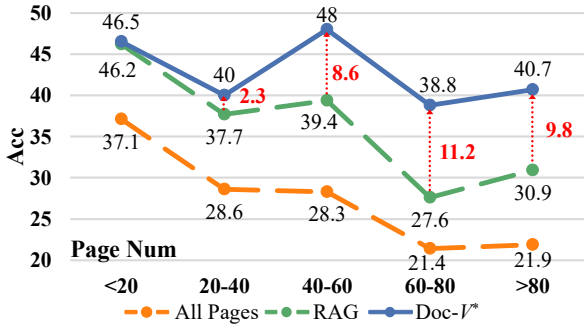


Figure 4: Accuracy vs. document length under different methods (RAG uses top-k = 5 retrieval).

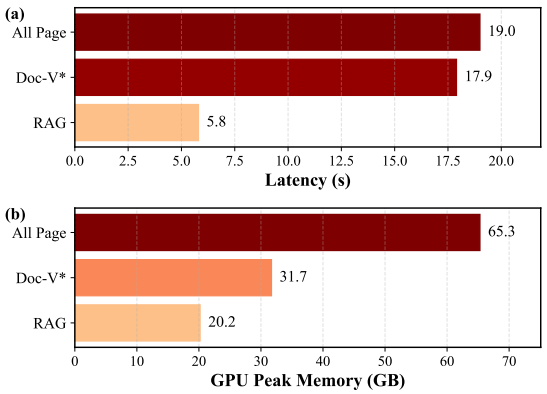


Figure 5: **(a)**: Comparison of average inference latency per sample across different methods. **(b)**: Comparison of average peak GPU memory consumption per sample under different methods.

that processing the entire document at once leads to substantially higher inference latency and GPU memory consumption, as all pages must be loaded and processed simultaneously. By contrast, the standard RAG baseline significantly reduces both latency and memory footprint by restricting computation to a small subset of retrieved pages. **Doc-V^*** occupies a middle ground between these two extremes: while incurring higher cost than RAG due to iterative page access and multi-step reasoning, it avoids the prohibitive overhead of full-document processing and achieves a more favorable balance between efficiency and document coverage.

Impact of Different Retrievers Table 2 shows that **Doc-V^*** maintains strong overall performance across retrievers with substantially different capabilities. Even when coupled with weak text-based retrievers (BM25 (Robertson et al., 2009), BGE-Large (Xiao et al., 2023)), which suffer from low Page-F1 and increased noise due to OCR and layout loss, Doc-V^* incurs only moderate performance degradation, indicating limited dependence on high-quality retrieval. Unlike conventional RAG pipelines where downstream performance is tightly coupled with retrieval recall, this robustness stems from **Doc-V^***'s active compensation mechanism: when initial retrieval misses critical evidence, the model detects contextual insufficiency and proactively recovers missing pages via browsing actions (e.g., `fetch_page`), effectively acting as an intelligent correction layer rather than a passive consumer.

4.5 Ablation Study

To validate the design choices of the proposed agentic framework, we conduct ablation experiments on **MMLongBench-Doc**, focusing on both the cognitive modules that govern the agent's reasoning process and the navigation actions that support evidence acquisition.

Importance of Multi-granularity Page Understanding Removing either the global thumbnail overview or the page-by-page analysis module causes significant performance drops of 4.9 and 4.7 accuracy points, respectively (Table 3), indicating that effective long-document reasoning relies on multi-granularity page understanding. The global overview provides structural cues for efficient navigation, while fine-grained analysis enables precise evidence extraction; using only one level of perception leads to either inefficient exploration or insufficient evidence recovery.

Complementary Roles of Retrieval and Fetch Actions We analyze the agent's navigation behavior using both page-level metrics (Table 5) and action ablation on MMLongBench-Doc (Table 4). The `retrieval_page` action achieves higher recall but lower precision, serving as a coarse semantic filter, while `fetch_page` provides higher precision for fine-grained evidence grounding. Ablation results further confirm their complementarity: removing retrieval leads to inefficient exploration (more pages), while removing fetch degrades accuracy. Combining both yields the best accuracy–efficiency trade-off, forming a coarse-to-fine evidence aggregation strategy.

Conclusion

This paper introduces **Doc-V***, an OCR-free agentic framework for multi-page document VQA via active evidence aggregation. Experiments on five benchmarks show gains over strong open-source baselines and competitive results against proprietary models, particularly on long and OOD docu-

Table 3: Ablation study on the cognitive modules of the **Doc-V^*** agent. **T**: Global Thumbnail Overview; **A**: Page-by-page content analyis; **M**: Memory.

Cognitive Modules T	A	M	MMLong. (Acc)	LongDoc. (Acc)	SlideVQA (F1)
✓	✓	✓	**39.8**	**53.0**	**73.8**
✗	✓	✓	34.9 (-4.9)	46.3 (-6.7)	68.3 (-5.5)
✓	✗	✓	35.9 (-4.7)	49.5 (-3.5)	71.8 (-2.0)
✓	✓	✗	36.4 (-3.4)	47.1 (-5.9)	69.8 (-4.0)

Table 4: Action ablation study on MMLongBench-Doc. Removing either retrieval or fetch leads to clear performance degradation.

Setting	Acc ↑	Avg. Pages ↓
Doc-V^*	**39.8**	6.4
w/o Retrieval	34.9	14.2
w/o Fetch	35.2	**5.9**

ments. These findings position selective evidence aggregation as a robust alternative to fixed-context and retrieval-augmented methods.

Limitations

This work is subject to several limitations. First, all experiments are conducted with a single backbone (Qwen2.5-VL), and the effectiveness of the proposed agentic framework across different vision–language backbones is not systematically evaluated. Although the method is conceptually backbone-agnostic, architectural differences may affect evidence aggregation and tool usage behaviors. Second, Doc-V^* is evaluated only in the single-document setting; its performance on multi-document scenarios, where evidence must be aggregated across multiple heterogeneous documents, remains unexplored and requires further study.

Ethical Considerations

Most datasets used in this work are publicly available benchmarks for document visual question answering and are utilized in accordance with their respective licenses. The proposed framework does not introduce new data collection or annotation processes involving human subjects. Similar to existing vision–language models, Doc-V^* may produce incorrect or incomplete answers due to hallucination or imperfect evidence aggregation, particularly on complex or ambiguous documents. As with prior work, its outputs are intended to support document understanding and analysis, rather than to serve as authoritative or final interpretations.

Table 5: **Page-level analysis of agent tool usage and retrieval quality across three benchmarks. RP** denotes pages retrieved by the `retrieval_page`, while **FP** denotes pages obtained via the `fetch_page`. **Ratio** indicates the proportion of samples in which the corresponding tool is invoked. **Recall**, **Precision**, and **F1** are computed at the page level.

Metric	SlideVQA		LongDoc.		MMLong.	
	RP	FP	RP	FP	RP	FP
Ratio	97.6	4.1	99.8	3.6	94.0	14.7
Recall	95.7	70.9	83.4	37.3	75.4	55.9
Precision	39.0	81.2	32.7	36.6	33.1	49.9
F1	54.1	72.9	44.4	31.9	42.1	49.6

Acknowledgments

This work is supported by the NSFC (62225603).

References

John Aloimonos, Isaac Weiss, and Amit Bandyopadhyay. 1988. Active vision. *International journal of computer vision*, 1(4):333–356.

Srikar Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie, and R Manmatha. 2021. Docformer: End-to-end transformer for document understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 993–1003.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Jian Chen, Ruiyi Zhang, Yufan Zhou, Tong Yu, Franck Dernoncourt, Jiuxiang Gu, Ryan A Rossi, Changyou Chen, and Tong Sun. 2024. Sv-rag: Lora-contextualizing adaptation of mllms for long document understanding. *arXiv preprint arXiv:2411.01106*.

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

Chao Deng, Jiale Yuan, Pi Bu, Peijie Wang, Zhong-Zhi Li, Jian Xu, Xiao-Hui Li, Yuan Gao, Jun Song, Bo Zheng, and 1 others. 2025. 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)*, pages 1135–1159.

Yihao Ding, Soyeon Caren Han, Jean Lee, and Eduard Hovy. 2025. Deep learning based visually rich document content understanding: A survey. *Preprint*, arXiv:2408.01287.

Alexey Dosovitskiy. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Yuchen Duan, Zhe Chen, Yusong Hu, Weiyun Wang, Shenglong Ye, Botian Shi, Lewei Lu, Qibin Hou, Tong Lu, Hongsheng Li, and 1 others. 2025. Docopilot: Improving multimodal models for document-level understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 4026–4037.

Manuel Faysse, Hugues Sibille, Tony Wu, Bilel Omrani, Gautier Viaud, Céline Hudelot, and Pierre Colombo. 2024. Colpali: Efficient document retrieval with vision language models. *Preprint*, arXiv:2407.01449.

Manuel Faysse, Hugues Sibille, Tony Wu, Bilel Omrani, Gautier Viaud, Céline Hudelot, and Pierre Colombo. 2025. Colpali: Efficient document retrieval with vision language models. *Preprint*, arXiv:2407.01449.

Masato Fujitake. 2024. Layoutllm: Large language model instruction tuning for visually rich document understanding. *arXiv preprint arXiv:2403.14252*.

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

Anwen Hu, Haiyang Xu, Liang Zhang, Jiabo Ye, Ming Yan, Ji Zhang, Qin Jin, Fei Huang, and Jingren Zhou. 2025. mplug-docowl2: High-resolution compressing for ocr-free multi-page document understanding. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5817–5834.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. 2022. Ocr-free document understanding transformer. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVIII*, page 498–517, Berlin, Heidelberg. Springer-Verlag.

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

Zhang Li, Biao Yang, Qiang Liu, Zhiyin Ma, Shuo Zhang, Jingxu Yang, Yabo Sun, Yuliang Liu, and Xiang Bai. 2024. Monkey: Image resolution and text label are important things for large multi-modal models. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26763–26773.

Falk Lieder and Thomas L Griffiths. 2020. Resource-rational analysis: Understanding human cognition as the optimal use of limited computational resources. *Behavioral and brain sciences*, 43:e1.

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

Yuliang Liu, Biao Yang, Qiang Liu, Zhang Li, Zhiyin Ma, Shuo Zhang, and Xiang Bai. 2024b. Textmonkey: An ocr-free large multimodal model for understanding document. *arXiv preprint arXiv:2403.04473*.

Chuwei Luo, Yufan Shen, Zhaoqing Zhu, Qi Zheng, Zhi Yu, and Cong Yao. 2024. Layoutllm: Layout instruction tuning with large language models for document understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15630–15640.

Yubo Ma, Yuhang Zang, Liangyu Chen, Meiqi Chen, Yizhu Jiao, Xinze Li, Xinyuan Lu, Ziyu Liu, Yan Ma, Xiaoyi Dong, and 1 others. 2024. Mmlongbench-doc: Benchmarking long-context document understanding with visualizations. *Advances in Neural Information Processing Systems*, 37:95963–96010.

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. 2021. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2200–2209.

Jamshed Memon, Maira Sami, Rizwan Ahmed Khan, and Mueen Uddin. 2020. Handwritten optical character recognition (ocr): A comprehensive systematic literature review (slr). *IEEE Access*, 8:142642–142668.

Mor Shpigel Nacson, Aviad Aberdam, Roy Ganz, Elad Ben Avraham, Alona Golts, Yair Kittenplon, Shai Mazor, and Ron Litman. 2025. Docvlm: Make your vlm an efficient reader. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29005–29015.

Stephen Robertson, Hugo Zaragoza, and 1 others. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Yongxin Shi, Jiapeng Wang, Zeyu Shan, Dezhi Peng, Zening Lin, and Lianwen Jin. 2025. Urag: Unified retrieval and generation in multimodal llms for efficient long document understanding. *arXiv preprint arXiv:2511.10552*.

Yulun Song, Long Yan, Lina Qin, Gongju Wang, Xingru Huang, Luzhe Hu, and Weixin Liu. 2025. Urag: Unified retrieval-augmented generation. In *Proceedings of the 2024 10th International Conference on Communication and Information Processing*, ICCIP '24, page 660–667, New York, NY, USA. Association for Computing Machinery.

Ryota Tanaka, Taichi Iki, Taku Hasegawa, Kyosuke Nishida, Kuniko Saito, and Jun Suzuki. 2025. Vdocrag: Retrieval-augmented generation over visually-rich documents. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24827–24837.

Ryota Tanaka, Kyosuke Nishida, Kosuke Nishida, Taku Hasegawa, Itsumi Saito, and Kuniko Saito. 2023. Slidevqa: A dataset for document visual question answering on multiple images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 13636–13645.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, and 1 others. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Rubèn Tito, Dimosthenis Karatzas, and Ernest Valveny. 2023. Hierarchical multimodal transformers for multipage docvqa. *Pattern Recognition*, 144:109834.

Jordy Van Landeghem, Rubèn Tito, Łukasz Borchmann, Michał Pietruszka, Pawel Joziak, Rafal Powalski, Dawid Jurkiewicz, Mickaël Coustaty, Bertrand Anckaert, Ernest Valveny, and 1 others. 2023. Document understanding dataset and evaluation (dude). In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19528–19540.

Bin Wang, Chao Xu, Xiaomeng Zhao, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Rui Xu, Kaiwen Liu, Yuan Qu, Fukai Shang, Bo Zhang, Liqun Wei, Zhihao Sui, Wei Li, Botian Shi, Yu Qiao, Dahua Lin, and Conghui He. 2024. Mineru: An open-source solution for precise document content extraction. *Preprint*, arXiv:2409.18839.

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

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

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding. *Preprint*, arXiv:2309.07597.

Xudong Xie, Hao Yan, Liang Yin, Yang Liu, Jing Ding, Minghui Liao, Yuliang Liu, Wei Chen, and Xiang Bai. 2024. Wukong: A large multimodal model for efficient long pdf reading with end-to-end sparse sampling. *arXiv preprint arXiv:2410.05970*.

Qixin Xu, Haozhe Wang, Che Liu, Fangzhen Lin, and Wenhu Chen. 2025. Cogdoc: Towards unified thinking in documents. *arXiv preprint arXiv:2512.12658*.

Dayu Yang, Antoine Simoulin, Xin Qian, Xiaoyi Liu, Yuwei Cao, Zhaopu Teng, and Grey Yang. 2025. Docagent: A multi-agent system for automated code documentation generation. *arXiv preprint arXiv:2504.08725*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*.

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

Shengbin Yue, Siyuan Wang, Wei Chen, Xuanjing Huang, and Zhongyu Wei. 2025. Synergistic multi-agent framework with trajectory learning for knowledge-intensive tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25796–25804.

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

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

A Detail of Environment Design

This subsection provides the detailed construction of the ***Global Thumbnail Overview*** $\tilde{D}$ referenced in our *Environment Initialization*. Given a document with N pages $D = \{I_1, \ldots, I_N\}$, we build $\tilde{D}$ as a small set of tiled overview images that together cover all pages while maintaining a very low

initial visual budget compared to all image with high-resolution. We set $G = 36$ to be the maximum number of pages allowed per overview image. We first partition the page indices into consecutive groups in sequential order

$$\mathcal{G}_k = \{(k-1)G + 1, \ldots, \min(kG, N)\},$$

where $k = 1, \ldots, K$, and the number of overview images K is

$$K = \left\lceil \frac{N}{G} \right\rceil, \quad n_k = |\mathcal{G}_k| \leq G.$$

Each page I_i is resized to a fixed thumbnail $T_i \in \mathbb{R}^{256\times256}$ (aspect-ratio handling follows standard padding/letterboxing so that all thumbnails share identical canvas size). For each group $\mathcal{G}_k$, we pack its n_k thumbnails into a single composite image $\tilde{I}^{(k)}$ using an adaptive near-square grid. Concretely, we choose grid dimensions (R_k, C_k) such that $R_k C_k \geq n_k$ and the grid is as close to square as possible; in practice, we set

$$R_k = \lceil \sqrt{n_k} \rceil, \quad C_k = \left\lceil \frac{n_k}{R_k} \right\rceil,$$

which guarantees $R_k C_k \geq n_k$ and yields a compact layout. If $R_k C_k > n_k$, the remaining cells are left empty (blank padding) to preserve a regular grid geometry.

To ensure unambiguous visual indexing, each grid cell includes a thin blank header band of height h pixels above the thumbnail region; we render the absolute page index i (for the corresponding thumbnail T_i) inside this header band. Thus, a cell is a $(h + 256) \times 256$ block consisting of a header strip for the index and a 256×256 thumbnail area below it. The resulting overview image $\tilde{I}^{(k)}$ is obtained by tiling these blocks into an $R_k \times C_k$ array, with empty cells rendered as blank blocks.

This construction yields the global overview set

$$\tilde{D} = \{\tilde{I}^{(1)}, \ldots, \tilde{I}^{(K)}\}, \quad K = \left\lceil \frac{N}{G} \right\rceil,$$

which is then used in the initial observation $O_1 = \{Q, \tilde{D}\}$ as described in the main paper.

For intuition, consider several document lengths. When $N = 40$, we obtain $K = \lceil 40/36 \rceil = 2$ overview images: the first group has $n_1 = 36$ pages and forms a 6×6 grid, while the second group has $n_2 = 4$ pages and forms a 2×2 grid. When $N = 50$, we again have $K = 2$: the first overview remains 6×6 (36 pages), and the second overview contains $n_2 = 14$ pages, which under the near-square rule becomes a 4×4 grid with two empty cells. In the appendix H, we visualizes these overviews images, it illustrates that these low-cost overviews provide strong initial navigational signals, especially for counting-style user questions.

In summary, a critical advantage of the proposed ***Global Thumbnail Overview*** is the substantial reduction in visual token consumption compared to full-resolution ingestion. Empirical analysis using the Qwen-2.5-VL (Bai et al., 2025) vision encoder demonstrates that our method achieves a compression ratio of approximately $10\times$ to $12\times$. For instance, a 100-page document processed at a standard high resolution of 1024×768 typically generates over $100,000$ visual tokens. In contrast, representing the same document via our tiled overview construction (resulting in $K = 3$ composite images) yields only $\approx 8,000$ visual tokens. While further downscaling of individual page thumbnails T_i is theoretically possible, our chosen resolution strikes a balance between **legibility and efficiency**. Consequently, this approach functions as a strategic compromise between full-document input—which preserves global context but incurs prohibitive computational costs—and Visual Retrieval-Augmented Generation (RAG), which optimizes for cost but often fragments global coherence. By retaining a macro-level visual representation, we preserve structural and semantic continuity while leveraging external tools for fine-grained details.

B Agent–Environment Interaction Protocol

This section provides a complete, implementation oriented description of how the **Doc-V^*** agent interacts with a multi-page document environment. Our goal is to make the interaction loop explicit and reproducible: what the agent *receives* at each turn, what it *must output*, how the environment *responds*, and how state (e.g., visited pages and working memory) is maintained. Please refer to Algorithm 1 for the complete pseudocode.

Given a document $\mathcal{D} = \{p_1, \ldots, p_N\}$ (each p_i is a page image) and a question Q, we cast multi-page Document VQA as a sequential decision process with a maximum budget of T interaction turns. At each turn t: (i) the agent receives an observation O_t, (ii) it performs reasoning and emits exactly one atomic action $a_t \in \mathcal{A}$, (iii) the environment exe-

cutes the action and returns feedback E_{t+1}, (iv) the feedback is incorporated into the next observation.

Crucially, the agent is **not** given the full document at high resolution upfront. Instead, the environment pre-computes and caches high-resolution visual tokens for each page and only reveals the requested pages on demand, enabling selective evidence acquisition under limited context/computation budgets.

Algorithm 1 Doc-V^* Agent–Environment Interaction (Inference-Time Loop)

Require: Document pages $\mathcal{D} = \{p_1, \ldots, p_N\}$, question Q, turn limit T, retrieval top-k, Global Thumbnail Overview $\tilde{D}$, high-res visual tokens $\mathbf{v}_i \leftarrow \mathcal{M}(\mathcal{V}(p_i))$

Initialization:

$\mathcal{P}_{\text{visited}} \leftarrow \emptyset$ ▷ tracks pages already revealed to the agent

$W \leftarrow \emptyset$ ▷ working memory: concatenated per-turn summaries

$O \leftarrow \{Q, \tilde{D}\}$ ▷ initial observation O_0

for $t \leftarrow 0$ to $T-1$ **do**

  $u_t \leftarrow \pi_\theta(O)$ ▷ must follow `<think>...</think><action>...</action>`

  Parse u_t to obtain (i) one atomic action a_t and (ii) summary S_t

  $W \leftarrow W \oplus S_t$ ▷ append summary to working memory

  if a_t is `<answer>` with string y **then**

    return y ▷ terminate interaction

  else if a_t is `<retrieval_page>` with query q_t **then**

    $\mathcal{I} \leftarrow \textsc{Retriever}(q_t, \mathcal{D} \setminus \mathcal{P}_{\text{visited}}, k)$ ▷ rank *unvisited* pages using an external multimodal retriever

  else if a_t is `<fetch_page>` with indices $[i_1, \ldots, i_m]$ **then**

    $\mathcal{I} \leftarrow [i_1, \ldots, i_m]$ ▷ direct request by absolute page indices

  else

    $\mathcal{I} \leftarrow \emptyset$ ▷ invalid action; environment may return a format reminder

  end if

  Environment feedback construction:

  $E \leftarrow \emptyset$

  for all $i \in \mathcal{I}$ **do**

    if $i \in \mathcal{P}_{\text{visited}}$ **then**

      $E \leftarrow E \cup \{$"Page i already visited."$\}$ ▷ avoid redundant visual tokens

    else

      $E \leftarrow E \cup \{$"Page i:", $\mathbf{v}_i\}$ ▷ prefix page id + cached high-res tokens

      $\mathcal{P}_{\text{visited}} \leftarrow \mathcal{P}_{\text{visited}} \cup \{i\}$

    end if

  end for

  $O \leftarrow E \cup \{W\}$ ▷ augmented observation for next turn: O_{t+1}

end for

return `NoAnswer` ▷ optional fallback when turn budget is exhausted

Cached high-resolution page tokens For each page p_i, the environment caches its high-resolution visual tokens $\mathbf{v}_i = \mathcal{M}(\mathcal{V}(p_i)) \in \mathbb{R}^{L_i \times d}$, computed at the page's native resolution (capped at 1024×768).

Initial Observation (t=0). Before any interaction, the environment constructs a *Global Thumbnail Overview* $\tilde{D}$ by resizing pages to thumbnails (e.g., 256×256), arranging them into one or more grid images, and annotating each thumbnail with its *absolute page number*. While fine text is typically unreadable at this scale, it preserves strong structural cues (document type, section layout, chart distribution, large-font titles). The initial observation is

$$O_0 = \{Q, \tilde{D}\}.$$

Visited Page Set The environment maintains an external set $\mathcal{P}_{\text{visited}}$ to prevent redundant page inputs. If the agent requests an already visited page, the environment returns a short *text reminder* rather than re-sending visual tokens.

Working Memory To reduce forgetting and repetitive behaviors during multi-turn interaction, we maintain a *Working Memory* W_t formed by concatenating the agent's per-turn summaries:

$$W_t = \text{Concat}(S_0, \ldots, S_{t-1}),$$

where S_t is the content of the agent's `<summary>` block at turn t.

Augmented Observation ($t \geq 1$). At turn $t \geq 1$, the agent receives an augmented observation:

$$O_t = E_t \cup \{W_t\},$$

where E_t is the environment feedback produced by executing the previous action.

At each turn, the agent must output *exactly one* atomic action from the following set:

- **Retrieval action:** `<retrieval_page>` q_t. This action mimics a "Ctrl+F"-like search but over page images. The query q_t may differ from the original question Q and can be iteratively refined.
- **Fetch action:** `<fetch_page>`$[i_1, \ldots, i_m]$. This action requests pages by absolute indices (e.g., based on thumbnail cues, adjacency exploration, or explicit page references in the question).
- **Answer action:** `<answer>` y. This action terminates the interaction and outputs the final answer string y.

To make decision-making auditable, we enforce a fixed ReAct-style output schema:

`<think>...</think><action>...</action>`.

At t=0, the <think> section should include (i) <analysis> based on thumbnails, (ii) <plan> for a turn-budgeted strategy, and (iii) <summary> to be appended to working memory. At t>0, the <think> section should include (i) <analysis> of newly returned pages, (ii) <relevant_pages> listing the page numbers judged relevant among the newly returned pages, and (iii) <summary>.

Environment Response Semantics For retrieval or fetch actions, the environment returns the cached high-resolution visual tokens of the requested pages. Each page's tokens are preceded by a textual page identifier (e.g., "Page 5:") to maintain an unambiguous mapping between content and absolute page index, especially when pages arrive out of order. For already-visited pages, the environment returns a short reminder string instead of re-injecting tokens.

C Detail of Training

Existing *Multi-page Document Visual Question Answering (VQA)* benchmarks usually annotate only the final supervision tuple (D, Q, y, P_{gt}), i.e., the document, question, final answer, and (optionally) evidence pages, but they do not provide the multi-step interaction traces required by our agent. To train the behavior model described in the main text, we adopt a **two-stage recipe**: first supervised fine-tuning (SFT) on distilled closed-loop interaction trajectories, and then GRPO-based (Guo et al., 2025) reinforcement learning to further optimize answer correctness and evidence discovery under a bounded interaction budget. In both stages, all environment feedback (returned page images and working memorys) is used only as conditioning context; training losses are applied only to tokens generated by the agent itself.

SFT: Closed-loop Interaction Trajectory Distillation We distill interaction trajectories from a strong teacher model (GPT-4o (Hurst et al., 2024)) by running it in a closed-loop environment that executes real actions and returns real page images. Each teacher turn must follow our protocol: one **<think>** block plus exactly one **<action>** among **<retrieval_page>**, **<fetch_page>**, and **<answer>**. The environment executes the action and returns the corresponding visual observation (thumbnail overview at the beginning; high-resolution pages thereafter) and working memory as feedback for the next turn. This closed-loop distillation is essential because retrieval and fetching change subsequent observations, so the distilled traces reflect realistic exploration dynamics rather than offline labels.

SFT: Trajectory Filtering We keep only reliable trajectories for imitation: 1) **Format validity:** the full trace must be parseable; every turn contains exactly one valid action with valid arguments and required fields in **<think>**; 2) **Answer correctness:** we compare the teacher final answer $\hat{y}$ with the ground-truth y. For free-form textual answers, we compute ANLS and require $\text{ANLS}(\hat{y}, y) \geq \tau_{\text{anls}}$ (we use $\tau_{\text{anls}} = 0.7$). For identifier-like answers (dates, counts, phone numbers, emails), we require exact match $\mathbb{I}[\hat{y} = y] = 1$. When ANLS is low (may be due to benign formatting differences), we additionally use a judge model (GPT-4o) to verify semantic equivalence; 3) **Evidence sanity:** the teacher outputs **<relevant_pages>** inside **<think>**. Let P_{rel} be the union of all pages listed in **<relevant_pages>** across turns. We require $P_{\text{rel}} \cap P_{\text{gt}} \neq \emptyset$; if not, we keep the trajectory only if another judge model (GPT-4o) verifies that the selected pages support the answer (to mitigate incomplete evidence annotations).

We build long-document training samples by selecting examples with more than 10 pages from MP-DocVQA (Tito et al., 2023) and DUDE (Van Landeghem et al., 2023) dataset. We keep DUDE not-answerable cases to improve abstention when evidence is insufficient. After distillation and filtering, our SFT set contains 9,019 trajectories in total (5,969 from MP-DocVQA and 3,050 from DUDE). Each distilled trajectory is serialized into a single sequence that interleaves environment observations and agent outputs across multiple turns. Observations include the current visual feedback (thumbnail overview or returned page images) and the accumulated working-memory summaries from previous turns. Agent outputs include structured **<think>** content (*analysis/plan/summary* in the first turn; *analysis/relevant_pages/summary* in later turns) followed by exactly one action tag. During training, the model is conditioned on the entire serialized prefix, but only the agent-generated tokens contribute to the loss.

SFT: Objective Let a serialized trajectory be the token sequence $x_{1:L}$. We define a mask $m_\ell \in \{0, 1\}$ indicating whether token x_ℓ belongs to the agent-generated part (<think> and <action>) or

to the environment observation. The SFT objective is the masked negative log-likelihood:

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = -\sum_{\ell=1}^{L-1} m_{\ell+1} \log \pi_\theta(x_{\ell+1} \mid x_{1:\ell}). \quad (1)$$

GRPO: Training Data While SFT enables effective imitation, it inherits teacher biases and does not explicitly optimize exploration efficiency under the interaction budget. We therefore further train the agent with GRPO (Guo et al., 2025), which optimizes expected trajectory-level reward using group-wise sampled rollouts. GRPO training uses only raw dataset-level supervision $(D, Q, y, P_{\mathrm{gt}})$ without intermediate traces. We select 2,048 training examples from MP-DocVQA and DUDE that do not overlap with the SFT training set. To ensure a balanced difficulty distribution, we estimate per-example difficulty using the SFT model: for each (D, Q) we run 4 independent rollouts and count the number of successes (ANLS ≥ 0.7). We then stratify samples into easy/medium/hard buckets and randomly draw them with proportions 10%/70%/20%, respectively.

For each training sample (D, Q), we run the current policy π_θ in the same closed-loop environment to sample a group of G complete trajectories $\{T_1, \ldots, T_G\}$ (stochastic decoding). Each trajectory terminates when the agent outputs `<answer>` or reaches the interaction budget. Each sampled trajectory can be represented as a pair (c_i, a_i), where c_i denotes all conditioning context tokens (all observations, including page images and working memory) and a_i denotes the concatenated agent-generated tokens (all **<think>** and **<action>** tokens) in that trajectory.

GRPO: Reward For a trajectory T, we compute

$$R(T) = w_{\mathrm{a}} R_{\mathrm{a}}(T) + w_{\mathrm{e}} R_{\mathrm{e}}(T) + w_{\mathrm{f}} R_{\mathrm{f}}(T).$$

R_{a} measures answer correctness. For free-form textual answers, we use thresholded ***Average Normalized Levenshtein Similarity (ANLS)***:

$$R_{\mathrm{a}}(T) = \mathbb{I}[\mathrm{ANLS}(\hat{y}, y) \geq \tau]\, \mathrm{ANLS}(\hat{y}, y),$$

where $\tau = 0.5$ and for identifier-like answers we use ***Exact Match (EM)***:

$$R_{\mathrm{a}}(T) = \mathbb{I}[\hat{y} = y]$$

For evidence, let $P_{\mathrm{rel}}(T)$ be the union of all pages listed in **<relevant_pages>** across turns. We compute a recall-weighted F-score:

$$R_{\mathrm{e}}(T) = \frac{(1+\beta^2)\, pr}{\beta^2 p + r}, \quad \beta^2 = 2,$$

where

$$p = \frac{|P_{\mathrm{rel}} \cap P_{\mathrm{gt}}|}{|P_{\mathrm{rel}}| + \epsilon}, \quad r = \frac{|P_{\mathrm{rel}} \cap P_{\mathrm{gt}}|}{|P_{\mathrm{gt}}| + \epsilon},$$

where ϵ is a small constant. Finally R_{f} penalizes binary invalid outputs (unparseable format, invalid action arguments, or budget violation), with a reward of 1 for valid output and 0 for invalid output.

GRPO: Objective GRPO optimizes relative performance within a sampled group. For each group $\{T_i\}_{i=1}^{G}$, let $R_i = R(T_i)$. We compute the group-normalized advantage:

$$A_i = \frac{R_i - \mu}{\sigma + \epsilon}$$

$$\mu = \frac{1}{G}\sum_{j=1}^{G} R_j, \quad \sigma = \sqrt{\frac{1}{G}\sum_{j=1}^{G}(R_j - \mu)^2}$$

We then update the policy by maximizing the log-likelihood of sampled actions weighted by A_i, using a PPO-style clipped objective at the token level. Let $\pi_{\theta_{\mathrm{old}}}$ denote the policy used to sample the group. For each trajectory i and each agent token position t, define the ratio

$$\rho_{i,t}(\theta) = \frac{\pi_\theta(a_{i,t} \mid c_i, a_{i,<t})}{\pi_{\theta_{\mathrm{old}}}(a_{i,t} \mid c_i, a_{i,<t})}. \quad (2)$$

The GRPO loss is defined as:

$$\mathcal{L}_{\mathrm{GRPO}}(\theta) = -\frac{1}{G}\sum_{i=1}^{G}\sum_{t=1}^{|a_i|} \min\Big(\rho_{i,t}(\theta)A_i, \mathrm{clip}\big(\rho_{i,t}(\theta), 1-\epsilon_c, 1+\epsilon_c\big)A_i\Big), \quad (3)$$

where ϵ_c is the clip range. Importantly, the loss is applied only on agent-generated tokens a_i; all environment observation tokens are used only as conditioning context.

Inference: Coarse-to-Fine Evidence Acquisition At test time, we use greedy decoding (temperature $= 0$) and enforce a maximum of T interaction steps. Starting from the global thumbnail overview, the agent follows a coarse-to-fine strategy: it uses structural cues in $\tilde{D}$ to propose candidate pages, employs `retrieval_page` with refined queries to localize evidence, uses `fetch_page` for targeted reading and cross-page completion when needed, updates its working memory via summaries, and terminates with `answer` once evidence suffices. The essential idea is not to increase context indiscriminately, but to keep the input in a high signal-to-noise regime by actively selecting what to read.

D Training and Inference Configuration

In this section, we provide the comprehensive hyperparameter settings and configuration details for the training and inference of **Doc-V^***. All experiments were conducted on a computational node equipped with 8 NVIDIA A100 (80GB) GPUs, implemented in PyTorch using BF16 mixed precision to optimize memory efficiency.

Stage I: Supervised Fine-Tuning (SFT) The primary goal of the SFT stage is to initialize the agent with stable tool usage capabilities and reasoning behaviors.

- **Data:** We utilize a filtered dataset comprising 9,019 high-quality interaction trajectories.
- **Optimization:** The model is trained for 3 epochs using the AdamW optimizer with a cosine learning rate scheduler. The initial learning rate is set to 3×10^{-6}.
- **Loss Masking:** To focus the model's adaptation on reasoning and planning, the loss is computed exclusively on agent-generated tokens (specifically the contents within `<think>` and `<action>` blocks), masking out the user instructions and environment observations.

Stage II: Group Relative Policy Optimization (GRPO) Following SFT, the agent undergoes reinforcement learning alignment to further refine its decision-making logic.

- **Hyperparameters:** We employ a group size of $G = 8$ with a sampling temperature of 1.0 to encourage exploration during the generation phase. The training proceeds for 3 epochs with a reduced learning rate of 2×10^{-6}.
- **Reward Configuration:** As outlined in the main text, the composite reward function is defined as $R = \omega_{\text{ans}} R_{\text{ans}} + \omega_{\text{evi}} R_{\text{evi}} + \omega_{\text{struct}} R_{\text{struct}}$. The specific coefficients are set to $\omega_{\text{ans}} = 0.6$ (Correctness), $\omega_{\text{evi}} = 0.3$ (Evidence Recall), and $\omega_{\text{struct}} = 0.1$ (Format Validity).

Inference Configuration During the evaluation phase, to ensure deterministic and reproducible results, we employ greedy decoding (temperature $= 0$). The maximum interaction horizon is fixed at $T = 8$ steps, consistent with the constraints applied during the training phase.

E Details of Datasets

MP-DocVQA (Tito et al., 2023) a multi-page document visual question answering benchmark that focuses on fine-grained information extraction from scanned documents. Questions often require precise localization of textual or visual elements within a document and explicit reasoning over page indices. The dataset emphasizes accurate page navigation and localized evidence grounding.

DUDE (Van Landeghem et al., 2023) consists of document images paired with questions that demand detailed visual-textual understanding. Compared to MP-DocVQA, DUDE places stronger emphasis on structured layouts such as forms and tables, and requires robust cross-page navigation to retrieve relevant evidence scattered across multiple pages.

SlideVQA (Tanaka et al., 2023) a document visual question answering dataset focused on understanding presentation slides. It contains slide documents with diverse visual layouts, including figures, charts, bullet lists, and sparsely distributed text. Documents typically span around 20 pages, and the associated questions require complex reasoning over non-linear reading orders and spatial arrangements, rather than relying solely on sequential textual flow.

LongDocURL (Deng et al., 2025) composed of web-based multi-modal documents with rich structural diversity, such as headings, hyperlinks, images, and embedded tables. With an average document length of approximately 30 pages, the dataset evaluates long-range retrieval and the ability to locate and synthesize information across distant document sections.

MMLongBench-Doc (Ma et al., 2024) designed for long-context multi-modal document understanding. Documents in this benchmark are substantially longer, extending up to 468 pages. The dataset poses significant challenges for scalable page selection, efficient navigation, and multi-hop reasoning over large multi-modal contexts.

F Details of Baseline

This section provides detailed specifications for the open-source baselines compared in our study. Table 6 summarizes their key configurations and training settings, followed by comprehensive descriptions of each method's architecture and paradigm.

Table 6: **Detailed configurations of Open Source baselines.** "Retriever" denotes the model used for page retrieval. "Param" refers to the parameter size of the LLM backbone. "Paradigm" categorizes methods into End-to-End (**E2E**), Retrieval-Augmented Generation (**RAG**), or **Agent**. The columns under "Trained on Dataset?" indicate whether the **backbone** was supervised fine-tuned (✓) on the corresponding benchmark's training set or evaluated in a zero-shot setting (×).

Method	Retriever	Backbone	Param	OCR-Free	Paradigm	Trained on Dataset?		
						DUDE	MPDocVQA	SlideVQA
HiVT5 *(PR)*	-	DiT / T5	0.3B	×	E2E	×	✓	×
CREAM *(ACM MM'24)*	bge-large	Pix2Struct / LLaMa2	7B	×	RAG	✓	✓	×
mPLUG-DocOwl2 *(ACL'25)*	-	ViT / LLaMa	8B	✓	E2E	✓	✓	×
M3DocRAG *(arXiv'24)*	Colpali	Qwen2-VL	7B	✓	RAG	×	×	×
VisRAG *(ICLR'25)*	VisRAG-Ret	MiniCPM-V 2.6	8B	✓	RAG	×	×	×
SV-RAG *(ICLR'25)*	SV-RAG-InternVL2	InternVL2	4B	✓	RAG	×	×	✓
VDocRAG *(CVPR'25)*	VDocRetriever	Phi3-Vision	4B	✓	RAG	✓	×	×
Docopilot *(CVPR'25)*	-	InternVL2	8B	×	E2E	✓	✓	×
DocVLM *(CVPR'25)*	-	Qwen2-VL	7B	×	E2E	×	×	×
InternVL3 *(arXiv'25)*	-	InternViT / Qwen2.5	8B	✓	E2E	×	×	×
VRAG-RL *(NeurIPS'25)*	ColQwen2	Qwen2.5-VL	7B	✓	Agent	×	×	✓
MoLoRAG *(EMNLP'25)*	Colpali+Qwen2.5-VL	Qwen2.5-VL	7B	✓	RAG	×	×	×
CogDoc *(arXiv'25)*	-	Qwen2.5-VL	7B	✓	Agent	✓	×	✓
URaG *(AAAI'26)*	Qwen2.5-VL (Early Layers)	Qwen2.5-VL	7B	✓	RAG	✓	✓	✓
Ours	Colqwen2.5	Qwen2.5-VL	7B	✓	Agent	✓	✓	×

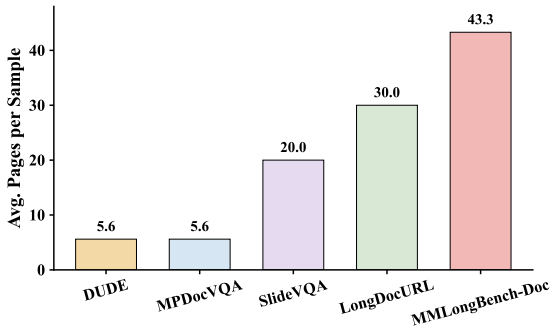


Figure 6: **Average document length across datasets.** The figure reports the average number of pages per document for MP-DocVQA, DUDE, SlideVQA, LongDocURL, and MMLongBench-Doc, illustrating the increasing document length and context complexity from standard document QA benchmarks to long-context multi-modal settings.

HiVT5 HiVT5 (Tito et al., 2023) proposes a hierarchical multimodal transformer to extend Document VQA to multi-page scenarios, addressing the quadratic complexity of standard attention mechanisms. Relying on an off-the-shelf OCR engine for text and bounding box extraction, it employs a T5-based encoder to process each page independently. The model fuses OCR tokens, layout embeddings, and visual features into learned [PAGE] tokens, which summarize page content conditioned on the query. These summaries are concatenated for the decoder to generate the answer, supported by a module predicting evidence page indices. Training involves a hierarchical layout-aware pre-training task followed by fine-tuning on MP-DocVQA.

CREAM CREAM (Zhang et al., 2024) presents a framework integrating coarse-to-fine retrieval with multimodal efficient tuning to handle token limitations in multi-page documents. It first utilizes an OCR engine to extract and chunk text, followed by a two-stage retrieval process: a coarse ranking via text embedding similarity and a fine-grained re-ranking where an LLM recursively groups chunks to select the top-k candidates. To incorporate visual context, a multi-page vision encoder employs attention pooling to merge features into a unified representation. Based on LLaMA-Adapter V2, the model undergoes multimodal instruction tuning (using LoRA and prefix tuning) to jointly optimize the LLM with retrieved chunks and visual embeddings.

mPLUG-DocOwl2 mPLUG-DocOwl2 (Hu et al., 2025) introduces a modularized Multimodal Large Language Model (MLLM) specialized for OCR-free document understanding. Improving upon the mPLUG-Owl architecture, it employs a visual abstractor to bridge the pre-trained visual encoder and the LLM, directly aligning visual features with textual semantics to eliminate external OCR dependency. The model is optimized via a unified instruction tuning strategy on a diverse document instruction dataset (covering tables, charts, and webpages), enhancing its capability to comprehend fine-grained visual text and complex structures.

M3DocRAG M3DocRAG (Cho et al., 2024) proposes a multimodal Retrieval-Augmented Generation (RAG) framework to overcome the limita-

tions of text-based pipelines in visually rich, open-domain tasks. Diverging from OCR-dependent methods, it adopts an all-multimodal paradigm using a vision-language retriever (e.g., ColPali) to encode page images into visual embeddings. This enables precise retrieval via late interaction mechanisms that preserve layout semantics. The retrieved top-k raw page images are then fed into an MLLM (e.g., Qwen2-VL) for end-to-end question answering. The authors also introduce M3DocVQA, a benchmark requiring cross-document retrieval and multi-hop reasoning.

VisRAG VisRAG (Yu et al., 2024) presents a vision-based RAG framework that treats document pages purely as images, mitigating information loss from OCR extraction. It employs a dual-encoder architecture (VisRAG-Ret) where queries and document images are encoded into a shared embedding space using position-weighted mean pooling. Generation (VisRAG-Gen) is handled by a generative VLM that synthesizes answers directly from the retrieved visual context. The retriever is fine-tuned via contrastive learning on a mixture of public VQA datasets and synthetic query-document pairs to ensure robust generalization.

SV-RAG SV-RAG (Chen et al., 2024) leverages a single MLLM backbone equipped with two distinct Low-Rank Adaptation (LoRA) adapters to handle both retrieval and generation without external parsers. It employs a retrieval adapter using contextualized late interaction to identify evidence pages, and a QA adapter for answer generation. The adapters are optimized via contrastive learning for retrieval and autoregressive generation for QA, enabling efficient, unified visual retrieval and reasoning within a single model architecture.

VDocRAG VDocRAG (Tanaka et al., 2025) introduces a visual RAG framework designed to process visually rich documents by leveraging visual features directly. It employs a dual-component architecture: VDocRetriever, which retrieves relevant page images using dense token representations, and VDocGenerator, which synthesizes answers from these inputs. To align visual and textual information, the authors utilize self-supervised pre-training tasks that adapt Large Vision-Language Models (LVLMs) for retrieval by compressing visual representations into dense tokens, facilitating open-domain document reasoning.

Docopilot Docopilot (Duan et al., 2025) proposes a native multimodal framework that eschews external retrieval in favor of scaling the model's intrinsic context processing. Centered on a "retrieval-free" paradigm, the model ingests entire documents as concatenated high-resolution image sequences. It leverages engineering optimizations like Ring Attention and Liger Kernel to manage long contexts (up to 32k tokens). The capability is supported by "Doc-750K," a large-scale dataset with diverse proxy tasks. Training involves Supervised Fine-Tuning (SFT) with multimodal data-packing, allowing the model to process full document contexts in a single forward pass to resolve long-distance dependencies.

DocVLM DocVLM (Nacson et al., 2025) presents a model-agnostic framework to enhance VLMs by efficiently integrating OCR-derived text and layout information. It utilizes an OCR encoder to capture textual and spatial details, compressing them into a compact set of learned queries (typically 64) which are projected into the LLM alongside visual features. This approach preserves the original VLM weights. Training follows a two-stage process: aligning the OCR encoder with the frozen VLM via captioning, followed by fine-tuning on DocVQA datasets, achieving high performance with reduced visual token usage.

InternVL3 InternVL3 (Zhu et al., 2025) is a state-of-the-art multimodal large language model (MLLM) developed by OpenGVLab that advances the field through a native multimodal pre-training paradigm, jointly acquiring visual and linguistic capabilities rather than adapting a text-only backbone. By incorporating variable visual position encoding (V2PE) for extended contexts and advanced post-training techniques like mixed preference optimization, the model achieves superior performance on diverse benchmarks, including MMMU and OCR-related tasks. In this study, InternVL3 is utilized as a strong baseline due to its robust optical character recognition (OCR) and document understanding capabilities, serving as a high-standard reference for evaluating the efficacy of the proposed method in visually rich environments.

VRAG-RL VRAG-RL (Wang et al., 2025) introduces an agentic framework empowering VLMs with iterative reasoning. It defines a unified action space integrating search queries with fine-grained visual perception actions, specifically pre-

dicting coordinates for cropping and zooming into information-dense regions to handle resolution bottlenecks. Operating in a "Thought-Action-Observation" loop, the model generates reasoning chains, executes actions to update observations, and iterates until evidence is gathered. The policy is optimized via Group Relative Policy Optimization (GRPO) with a reward function incentivizing both retrieval precision and answer accuracy.

MoLoRAG MoLoRAG (Wu et al., 2025) proposes a logic-aware retrieval framework capturing both semantic and logical dependencies. It constructs a document-level "page graph" where edges represent semantic similarities. A lightweight VLM acts as a retrieval engine, traversing this graph by evaluating "logical relevance"—the inferential necessity of a page—alongside semantic alignment. This allows the model to uncover logically connected but semantically distant evidence. The framework supports both a training-free mode and a fine-tuned mode where the engine is optimized on synthesized "question-image-relevance" triplets.

CogDoc CogDoc (Xu et al., 2025) proposes a unified, two-stage cognitive framework mimicking human reading patterns to balance scalability and fidelity. It decomposes reasoning into two phases executed by a single VLM: a "Fast Reading" phase (Localization Mode), scanning the document at low resolution to predict page indices based on structural cues; and a "Focused Thinking" phase (Reasoning Mode), processing localized pages at high resolution for grounded reasoning. To avoid policy conflicts in supervised training, it employs Direct Reinforcement Learning (RL from scratch), enabling the model to autonomously learn to alternate between global scanning and local reasoning.

URaG URaG (Shi et al., 2025) introduces a unified framework integrating retrieval and generation within a single MLLM to handle long documents efficiently. Based on the observation that MLLMs exhibit a "coarse-to-fine" attention pattern, the method inserts a lightweight cross-modal retrieval module into the model's early layers (e.g., layer 6). This module acts as an internal evidence selector, computing relevance via late interaction and retaining only the top-k pages while discarding irrelevant tokens from subsequent layers. This "early-exit" mechanism reduces computational overhead for deeper reasoning layers. Training involves pre-training the retrieval module followed by joint fine-tuning of both components.

Table 7: **Impact of K on MMLongBench-Doc.** "Adaptive" denotes the document-adaptive setting $K = \min(\lceil N/10 \rceil, 4)$, where N is the total number of pages.

K	Avg. Pages	Overall	Breakdown		
			SIN	MUL	UNA
Adaptive	5.6	42.1	54.6	23.5	45.7
1	3.0	40.5	51.6	17.0	56.1
2	4.3	39.7	54.4	20.7	39.9
3	5.4	40.1	53.3	23.0	40.8
4	6.5	41.1	53.3	24.0	43.5
5	8.1	41.7	52.9	23.5	48.4

Table 8: **Impact of maximum interaction steps on MMLongBench-Doc.**

Iteration	Avg. Pages	Overall	Breakdown		
			SIN	MUL	UNA
3	4.2	41.1	54.0	22.5	44.4
4	4.6	41.2	54.3	23.0	43.5
5	4.9	41.4	54.3	23.4	43.5
6	5.2	41.5	54.4	24.0	44.0
7	5.6	42.1	54.6	23.5	45.7
8	5.8	41.4	54.3	24.0	44.0
9	6.0	41.5	54.8	24.0	44.0
10	6.2	41.5	54.8	24.0	44.0

G Robustness of Iteration & K

We investigate how the number of interaction turns (iterations) affects the agent's performance on the **MMLongBench-Doc** dataset. As the agent operates in a recursive "Observe-Think-Act" loop, the number of steps determines the depth of exploration. As shown in Table 7, performance improves consistently as the maximum iteration limit increases from 3 to 7. The model achieves peak performance at **7 iterations** with an overall accuracy of **42.1%**. This suggests that for complex long-document tasks, the agent requires approximately 5–7 steps to effectively locate evidence and synthesize answers. Beyond 7 iterations, the performance plateaus and slightly fluctuates, indicating that the agent has converged and further exploration yields diminishing returns.

We further analyze the effect of the page selection budget K on **MMLongBench-Doc**, as reported in Table 8. Overall performance exhibits a clear non-monotonic trend with respect to K. When K is small (e.g., $K = 1$ or 2), the agent is restricted to a limited number of pages, leading to

insufficient evidence coverage and degraded overall accuracy. As K increases, performance improves steadily and reaches its peak under the **Adaptive** setting, where $K = \min(\lceil N/10 \rceil, 4)$. This adaptive strategy achieves the best overall accuracy of **42.1%** while maintaining a moderate average page count of 5.6.

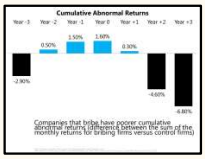
Further increasing K beyond the adaptive range does not result in consistent gains. Although larger K values introduce more pages, the additional context also brings redundant or irrelevant information, which weakens evidence aggregation and slightly hurts performance. This effect is particularly evident in the SIN and MUL subsets, where accuracy saturates or fluctuates as K grows. These results indicate that effective long-document reasoning depends on selecting a well-calibrated number of pages rather than aggressively expanding the context. The adaptive strategy strikes a favorable balance between evidence sufficiency and noise control, highlighting the importance of dynamic, document-aware page budgeting.

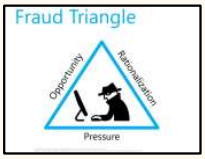
H Case Study

See Fiugre 7, 8, 9

Question: What is the difference in percentage from Year 0 to Year 3 regarding cumulative abnormal returns?

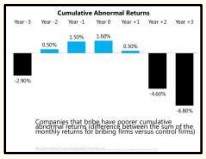
Input All Pages



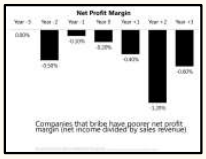


<answer>
4% increase
</answer>

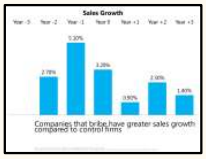
RAG-Top 5



Page 14



Page 12



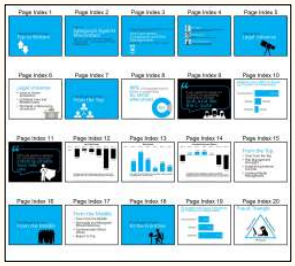
Page 13

Page 11

Page 9

<answer> -4.3</answer>

Doc-V* (SFT & GRPO)



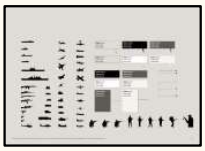
<think> **\\boxed{thumbnail}** {The thumbnail shows a slide deck with 20 slides. Several slides contain charts: Page Index 12 looks like a bar chart titled "Net Profit Margin,"..... **Page Index 14 a bar chart titled "Cumulative Abnormal Returns,"** and Page Index 20 a diagram titled "Fraud Triangle."}
\\boxed{plan} {
1) Locate the slide titled "Cumulative Abnormal Returns."
2) Read the values for Year 0 and Year 3 on that chart.
3) Compute the difference between Year 3 and Year 0 percentages.}
</think> <fetch_page> [14] </fetch_page>

<think> **\\boxed{analysis}** { Page Index 14: **Relevant**. It shows the chart titled "Cumulative Abnormal Returns" with bars labeled Year -3 through Year +3. The value at Year 0 is 1.60% and at Year +3 is -6.80%. The difference between Year 0 and Year +3 is **1.60% - (-6.80%) = 8.40%.** }
\\boxed{relevant_pages} { 14 }
\\boxed{summary}{ The difference is 8.40%. }
</think> <answer> difference is **8.40%** </answer>

Figure 7: Case 1 in SlideVQA between different methods

Question: What is the difference in percentage from Year 0 to Year 3 regarding cumulative abnormal returns?

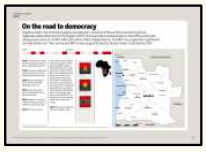
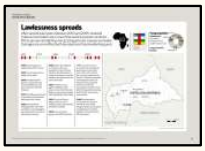
Input All Pages




<answer>
The World's Conflicts Focus 2009
</answer>

RAG-Top 5

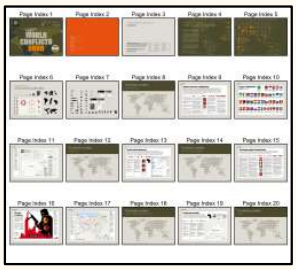


Page 1 Page 3 Page 6 Page 13 Page 18

<answer> The World's Conflicts Focus 2009</answer>

Doc-V* (SFT & GRPO)



<think> **\\boxed{thumbnail}** {The thumbnail shows a 20-page booklet titled "World Conflicts 2009" by Nordstrom Frank. **The first page appears to be a cover with pricing information**; subsequent pages include maps, symbols, country profiles, and conflict data. The price "2,500" likely appears prominently on the cover.}
\\boxed{plan} {
1) Locate where the price "2,500 euro" is mentioned.
2) Identify what item is available for that price.
3) Provide the concise answer. }
</think> <fetch_page> [1] </fetch_page>

Doc-V* (SFT)

<think> **\\boxed{analysis}** { Page Index 1 is the cover. It clearly states "Purchase now! Send a mail to: info@nordstromfrank.se €2,500 Contains facts, maps, flags and templates for use in InDesign." This directly answers what can be purchased for 2,500 euro: the World Conflicts 2009 Editorial Toolbox Focus. } **\\boxed{relevant_pages}** { 1 } **\\boxed{summary}** { From Page Index 1 (cover), we learn that for 2,500 euro, one can purchase the World Conflicts 2009 Editorial Toolbox Focus. }
</think> <answer> The World's conflicts 2009 Editorial Toolbox Focus </answer>

Doc-V* (GRPO)

<think> **\\boxed{analysis}** { ..... This directly answers what can be purchased for 2,500 euro: a supplement containing facts, maps, flags, and templates for InDesign. } \\boxed{relevant_pages} { 1 } \\boxed{summary} {From Page Index 1, we learned that for 2,500 euro, **you can purchase a supplement containing facts, maps, flags, and templates for use in InDesign**. }
</think> <answer> a supplement containing facts, maps, flags, and templates for InDesign </answer>

Figure 8: Case 2 in SlideVQA between different methods

Question: What specific word is the red arrow pointing toward?

Input All Pages



<answer>
Problem
</answer>

RAG-Top 5


Page 2


Page 3


Page 5


Page 6


Page 15

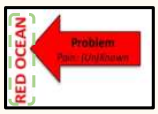
<answer> Problem</answer>

Doc-V* (SFT & GRPO)

<think>
\\boxed{thumbnail} {The document appears to be a slide deck. Each slide has a red arrow pointing to a specific field on the template. The question likely refers to the target of one of these red arrows on a template slide.}
\\boxed{plan} {
1) Identify which page contains a red arrow pointing to a specific labeled field.
2) Read the label/word the red arrow points toward.}
</think>
<page_retrieval>Find the page(s) where a red arrow points to a specific labeled field; identify the word the red arrow targets.</page_retrieval>


Page 2


Page 6




Page 11

<think>
\\boxed{analysis}{
Page Index 6: Shows a "DRAMATIC STORY CANVAS: Storyboard Template." There is a red arrow pointing to a red box labeled "Problem" with the subtitle "Pain: (Un)Known." This directly answers the question about what the red arrow points toward.
Page Index 11: Another "DRAMATIC STORY CANVAS FOR DEVELOPING 'GOOD STRATEGY'" with a red arrow pointing to the same red box labeled "Problem," confirming the label.
Page Index 2: "DRAMATIC STORY CANVAS: Space-Timeline Template" also has a red arrow pointing to the red box labeled "Problem."All three pages provide consistent evidence; the specific word is "Problem."}
\\boxed{relevant_pages} {6, 11, 2}
\\boxed{summary} {Across Page Index 6, 11, and 2, the red arrow clearly points to a red box labeled "Problem." This resolves the question.}
</think> <answer>Problem</answer>

Figure 9: Case 3 in SlideVQA between different methods