

VinQA: Visual Elements Interleaved Long-form Answer Generation for Real-World Multimodal Document QA

Young Rok Jang* Hyesoo Kong* Kyunghwan An Jae Sub Huh
 Gyeonghun KIM Stanley Jungkyu Choi
 LG AI Research

{jyrok3357, hyesoo.kong, ghkayne.kim, stanleyjk.choi}@lgresearch.ai

Abstract

Real-world documents combine text with tables, charts, photographs, and diagrams arranged in diverse layouts, yet existing research on multimodal large language models (MLLMs) for document QA predominantly produces text-only responses, underutilizing these visual elements. We introduce VinQA, a dataset designed for long-form answer generation where cited visual elements are explicitly interleaved with their supporting text and grounded in relevant document pages. To support this task, we study two encoding methods for feeding raw document page images into an MLLM, along with their visual-element citation mechanisms: (1) Page Encoding, which directly encodes full-page images with bounding boxes of visual elements and treats these boxed regions as citable units; and (2) Modality Encoding, which parses each page to extract text and crop visual elements, encodes them separately, and uses these cropped elements as citable units. In our experiments, we propose M-GroSE, a multimodal evaluation framework extending GroUSE to assess such answers along four dimensions: completeness, answer relevancy, faithfulness, and unanswerability. We additionally report Visual Source F1 to directly measure visual citation accuracy. Although proprietary frontier models still achieve the best overall scores on the VinQA test split, fine-tuning open Qwen2.5-VL models on the VinQA training split substantially improves their performance and markedly narrows this gap. Modality Encoding is initially more robust than Page Encoding for complex documents with long text, many visual elements, and diverse visual citation requirements. After training on VinQA, however, Page Encoding reaches a comparable performance level, showing that it can compete effectively even without the explicit parsing used in Modality Encoding. Finally, Visual G-Eval, an MLLM-based judge, confirms that fine-tuned models insert visual elements at semantically ap-

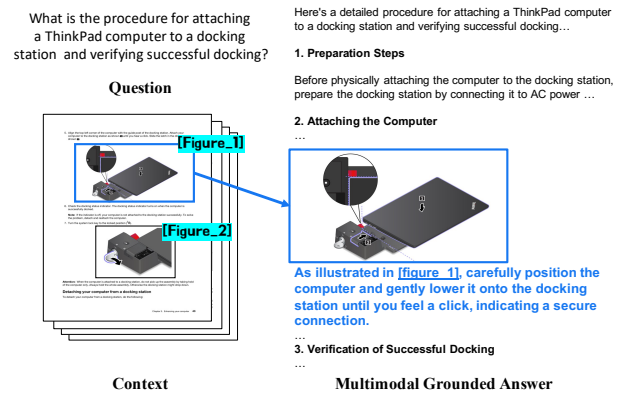


Figure 1. The example of VinQA. Each question is paired with a context, and the multimodal grounded answer cites relevant visual elements with supporting text, inserting each cited element immediately before its citing sentence.

propriate positions with faithful supporting text.

1. Introduction

Real-world documents are inherently multimodal, comprising not only text but also various visual elements (e.g., tables, charts, photographs, and diagrams) arranged in diverse layouts. With the advent of Multimodal Large Language Models (MLLMs) [1, 20, 28], various studies have explored question answering over such documents. Some approaches process entire document pages [2, 4, 5, 13, 27], while others first retrieve relevant pages using a multimodal retriever in a multimodal Retrieval-Augmented Generation (RAG) scenario, and then generate grounded answers from the retrieved context [3, 15, 23]. However, these studies only consider text-only answers, underutilizing the rich visual elements embedded in the documents.

For example, as illustrated in Figure 1, when a question pertains to a specific part of a product’s installation process,

*Equal contribution.

providing textual explanations alongside a figure that illustrates the corresponding installation step can significantly enhance understanding. Similarly, figures (e.g., diagrams) are effective for conveying complex ideas in research domains, and charts and tables are well-suited for presenting quantitative information in financial domains.

Motivated by these benefits, some recent work has begun to interleave visual elements into the answers [9, 19, 22]. However, these approaches typically either operate on narrow-domain document collections or give limited consideration to the procedure for detecting and citing visual regions from raw document page images.

In this paper, we introduce VinQA, the Visual Elements Interleaved Long-form Answer Generation in Question Answering dataset, consisting of training and test splits. VinQA is built from diverse real-world documents that combine text with various visual elements. As shown in Figure 1, each VinQA instance includes (i) a Question requiring both textual and visual understanding, (ii) a Context of the top- K relevant pages obtained through a multimodal RAG pipeline, and (iii) a Multimodal Grounded Answer—a long-form response grounded in the context, where visual elements are interleaved before the supporting text that cites them.

We propose two encoding methods for feeding raw document page images into the MLLM, each with a visual-element citation mechanism. In *Page Encoding*, we directly encode each page as visual tokens together with bounding boxes detected for visual elements, and the model cites these regions via their identifiers. In *Modality Encoding*, we parse each page with OCR to extract text and crop detected visual elements, encode them separately, and cite each cropped element via its identifier. *Page Encoding* preserves full-page layout by operating directly on raw images, but it also requires the model to read text and interpret bounding-box regions from the image itself.

In our experiments, we propose M-GroSE (Multimodal Grounded QA Scoring Evaluator), which extends GroUSE [14] from text-only long-form grounded answers to multimodal answers, and evaluates three dimensions: relevance, completeness, and faithfulness. We also report Visual Source F1, which directly measures visual-element citation accuracy. Although proprietary frontier models still achieve the highest scores on these metrics on the VinQA test split, fine-tuning the open-source Qwen2.5-VL [1] model on the VinQA training split substantially improves its performance and markedly narrows this gap. Modality Encoding is initially more robust than Page Encoding on complex documents, for example those with long text and many visual elements, but after training on VinQA the performance gap largely disappears, showing that Page Encoding can perform competitively even without the explicit parsing used in Modality Encoding. In addition, ab-

lations with Visual G-Eval, an MLLM-based evaluator focused on the visual aspects of cited elements, show that VinQA helps models use cited visual elements more effectively and place them in contextually appropriate positions with faithful supporting text.

Our main contributions are summarized as follows:

1. **VinQA dataset.** We present a document-grounded QA dataset in which long-form answers interleave cited visual elements with their supporting text.
2. **Two encoding strategies with visual-element citation.** *Page Encoding* encodes full-page images with bounding boxes of visual elements and treats these boxed regions as citable units. *Modality Encoding* parses pages and then encodes text and cropped visual elements separately, using these crops as citable units.
3. **Evaluation framework for multimodal grounded answers.** We propose M-GroSE to evaluate the groundedness of multimodal long-form answers. To independently assess visual-element citations, we report Visual Source F1. We also use Visual G-Eval, an MLLM-based evaluator that directly considers the visual content of cited elements.
4. **Key findings.** We empirically demonstrate that VinQA effectively improves both the groundedness of multimodal answer generation and the accuracy of visual-element citations. In particular, Page Encoding, when trained on VinQA, reaches performance comparable to Modality Encoding even without explicit parsing. Furthermore, Visual G-Eval shows that VinQA helps models insert visually informative elements at contextually appropriate positions with faithful supporting text.

2. Related Work

2.1. Multimodal Document QA

Research on multimodal document QA has progressed along two main lines: retrieval-augmented pipelines and full-document processing with MLLMs. The former retrieves relevant document pages using a retriever and then generates answers from the retrieved context [3, 10, 15]. VisRAG [23] and VDocRAG [17] further explore grounded short-form answer generation by adopting a *Page Encoding* strategy, in which each page is encoded as an image input to the MLLM. In parallel, another line of work performs QA by feeding entire documents directly into an MLLM without a retrieval step [4, 5, 13, 27]. M-LongDoc [2] constructs long-form grounded QA data over documents such as academic papers and financial reports, and adopts a *Modality Encoding* approach.

While these approaches enable multimodal document understanding via Page Encoding and Modality Encoding, the generated answers remain text-only. Therefore, we build on these two encoding methods and equip each with

a visual-element citation mechanism, enabling models to generate long-form answers that explicitly interleave cited visual elements with supporting text.

2.2. Multimodal Citation Text Generation

Citation text generation aims to produce answers that explicitly cite the given context. Early studies focus on text-only contexts [8, 11, 24], while recent work extends the task to multimodal settings [19]. VISA [12] additionally predicts bounding boxes as visual evidence for the cited source regions. However, VISA does not interleave cited visual regions into the answer with supporting text, as VinQA does.

MCiteBench [9] requires citing both textual spans and visual elements (e.g., figures, tables), but its context is limited to pre-extracted text snippets and cropped images of visual-elements from academic papers. MRAMG-Bench [21] generates multimodal answers grounded in pre-separated text blocks and images extracted from academic, web, and lifestyle documents. In contrast, we start from raw document page images and investigate how to convert them into MLLM inputs with an explicit citation mechanism. Our dataset also covers a broader range of document domains (Table 2), and we further introduce M-GroSE, an evaluation framework tailored to assessing such multimodal grounded answers.

3. Visual Elements Interleaved Long-Form Answer Generation in Question Answering (VinQA) Dataset

We introduce the VinQA dataset for question answering over multimodal contexts, with its statistics summarized in Table 2 and key characteristics compared with prior datasets in Table 1.

(i) **Question.** VinQA covers diverse question types, including cross-page (multi-page reasoning), cross-modal (requiring both textual and diverse types of visual evidence), and unanswerable questions. (ii) **Context.** The context is built using multimodal retriever over diverse document corpus spanning heterogeneous document domains—academic paper, web pages, textbooks, guidebooks, research reports, financial reports, and slides (P/W/T/G/R/F/S). It contains rich visual elements (charts, tables, figures) and complex layouts (e.g., slides). Moreover, it supports multi-document cases, where a single instance may comprise pages taken from different documents. (iii) **Answer.** Each answer is a structured, long-form response that includes a header and a body, grounded in both text and visual elements. Cited visual elements are interleaved at contextually appropriate positions with supporting text.

We first use an MLLM to textualize each multimodal document, followed by an LLM that generates corresponding questions, contexts, and answers. The resulting data are

organized into multimodal inputs using the encoding methods in Section 4, enabling MLLMs to inherit and extend LLM grounding capabilities. The overall construction process is illustrated in Figure 2.

3.1. Document Collection and Data Preprocessing

To construct the VinQA corpus, we first collect multimodal documents from real-world sources by aggregating the source documents from six public document QA datasets: MMLongBench-Doc [13], TAT-DQA [26], VisDoM [15], SlideVQA [16], MMDocIR [6], and VisRAG [23]. We only utilize the raw documents from these datasets and do not use their original question–answer annotations. We manually classify the documents into seven domains based on topic and format, as shown in Table 1. The corpus is split into training and test sets using domain-balanced stratified sampling, ensuring document-level separation.¹

To textualize document pages, we extract text using OCR and convert visual elements into structured textual representations. Concretely, we detect charts, tables, and figures with DocLayout [25] to obtain their bounding boxes on each page. For every detected element, we prompt an MLLM with two types of inputs: (i) the full page image, where the target element is highlighted with a bounding box, and (ii) the cropped image of the target element. Using these inputs, the MLLM generates two textual outputs: (a) a caption—the in-document text that directly refers to the visual element, and (b) a description—a longer explanation derived from surrounding text and the visual features of the element itself.

3.2. Question and Answer Generation

We generate questions that are naturally asked about our collected document corpus. Specifically, we randomly sample a document page, extract its image embedding using ColPali [7], and cluster pages by embedding similarity. An LLM then generates questions for each cluster, including cases that require reasoning across multiple pages and modalities, thereby encouraging multi-page and cross-modal grounding in subsequent QA tasks. Following VisRAG [23], we discard questions that rely on local context but omit explicit entity names, as such questions are ambiguous for retrieval.

We then use ColPali to retrieve the top- K most relevant document pages for each question; these pages serve as the context. During answer generation, we prompt the LLM with this context to produce a long-form grounded answer that cites visual elements via their identifiers, along with faithful supporting text at contextually appropriate positions. In a post-processing step, each cited visual element

¹We exclude textbook documents from the test split because its pages, unlike other categories, lack clear document boundaries, making it impossible to avoid overlap between training and test corpus.

Dataset	Question Type			Contexts			Answer	
	Cross-page	Cross-modal	Unans.	Multi Doc	Doc Domains	Visual Elements	Type	Modalities
SlideVQA [16]	✓	✓	✗	✗	S	Ch/Fg/Tb	Short	T
MMLongBench-Doc [13]	✓	✓	✓	✗	P/G/R/F/S	Ch/Fg/Tb	Short	T
LongDocURL [4]	✓	✓	✓	✗	P/T/G/R/S	Ch/Fg/Tb	Short	T
VisDoM [15]	✓	✓	✗	✓	P/W/S	Ch/Fg/Tb	Short	T
Document Haystacks & V-RAG [10]	✗	✓	✗	✓	Etc	Fg/Tb	Short	T
VDocRAG [17]	✓	✓	✓	✓	W/S/Etc	Ch/Fg/Tb	Short	T
ViDoRAG [18]	✗	✗	✗	✓	P/W/S	Ch/Fg/Tb	Short	T
VisRAG [23]	✓	✓	✗	✓	P/W/T/G/S/Etc	Ch/Fg/Tb	Short	T
M-LongDoc [2]	✓	✗	✗	✗	P/G/F	Ch/Fg/Tb	Long	T
VISA [12]	✗	✗	✗	✓	P/W	Fg/Tb	Short	T+V
MCiteBench [9]	✗	✓	✗	✗	P	Fg/Tb	Long	T+V
MDocSum [19]	✓	✓	✗	✗	P	Fg/Tb	Long	T+V
MRAMG-Bench [22]	✓	✓	✗	✓	P/W/G/Etc	Fg/Tb	Long	T+V
VinQA (Ours)	✓	✓	✓	✓	P/W/T/G/R/F/S	Ch/Fg/Tb	Long	T+V

Table 1. Comparison of multimodal document QA datasets in terms of question type, context, and answer. **Question Types** include **Cross-page**, **Cross-modal**, and **Unanswerable** questions. **Contexts** comprise **Multi Doc**, **Doc Domain**, and **Visual Elements**. **Doc Domains**: **P** (Academic Paper), **W** (Web), **T** (Textbook), **G** (Guidebook), **R** (Research Report), **F** (Financial Report), **S** (Slides), **Etc** (Others). **Visual Elements**: **Ch** (Chart), **Fg** (Figure), **Tb** (Table). **Answer** consists of **Type** (Short vs. Long) and **Modalities** (**T** vs. **T+V**), where **T** indicates text-only answers, and **T+V** denotes answers grounded on both textual and visual elements.

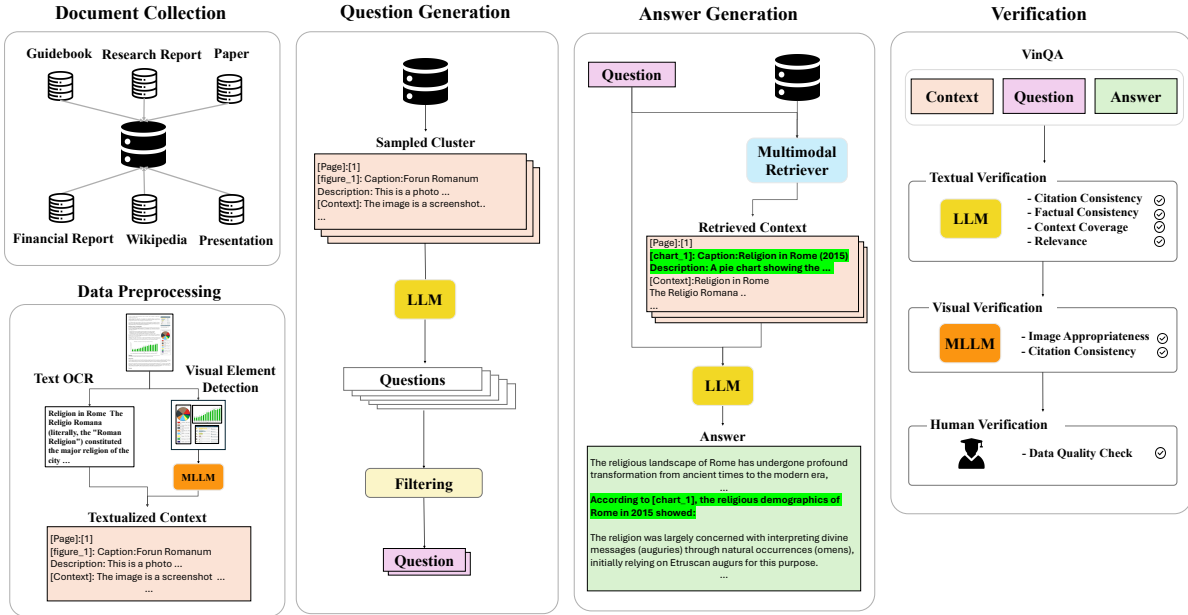


Figure 2. Overview of VinQA construction process. We design the data construction process with the consideration of simulating a Multimodal RAG pipeline over real-world documents.

is inserted immediately above the paragraph where its identifier appears.

To construct unanswerable QA examples, we first form hard negatives by retrieving lower-ranked pages from ColPali that have little or no relevance to the given question. We then retain only those cases where the LLM correctly identifies the question as unanswerable. This procedure mimics realistic retrieval errors that may occur in practice and makes the task more challenging by requiring the model to distinguish between answerable and unanswerable questions based on the retrieved content.

3.3. Data Verification

To ensure the quality of the generated QA data, we perform a three-step verification process consisting of textual, visual, and human verification. For the textual verification step, each question–context–answer triple is evaluated by a state-of-the-art LLM along four key aspects: citation accuracy, factual consistency with the context, completeness of relevant information, and absence of irrelevant content. For the visual verification step, a state-of-the-art MLLM checks whether visual elements (e.g., charts, tables, figures)

Category	Train	Test
Documents		
Total (# Pages)	131,906	9,373
Guidebook	19,616	1,278
Financial report	1,554	1,681
Academic Paper	35,331	2,081
Presentation	48,703	1,953
Research report	398	565
Web	16,508	1,815
Textbook	9,796	–
QA		
Total	42,700	1,605
Answerable	39,700	1,206
Unanswerable	3,000	399
Answerable Q Types		
Single-page	4,030	176
Multi-page	35,670	1,030
Text-only	12,949	369
Text + Visual Elements	26,751	837
- Chart	4,170	204
- Figure	9,984	226
- Table	7,010	293
- Mixed	5,587	114

Table 2. Statistics of VinQA. Answerable question types are categorized as follows: *Single-page* and *Multi-page* indicate how many pages each question must be grounded in, while *Text-only* and *Text + Visual Elements* indicate which modalities are used for grounding that question.

are properly cited and semantically aligned with the corresponding statements.

For the test set, we further reduce noise by running an additional textual verification pass with a different LLM and by performing human verification focused on visual aspects. A team of four human annotators manually inspects each QA pair following a structured guideline that checks (i) whether the bounding boxes of visual elements are correctly specified, (ii) whether visual citations are accurate and necessary to support the answer, and (iii) whether the supporting text is free of hallucinations. After human verification, 106 answerable and one unanswerable QA pairs were filtered out, resulting in a highly reliable and clean test set.

4. Methods

Given a user query q and a context composed of n document page images $\mathcal{P} = \{p_1, p_2, \dots, p_n\}$, the MLLM is tasked with producing a grounded long-form answer $A = \{x_1, x_2, \dots, x_k\}$, where each x_i is a text span. Some spans include a specific visual-element identifier token for citation, and the corresponding visual element is interleaved immediately preceding the span. This section describes two encoding methods for converting $\mathcal{P}$ into MLLM inputs and

enabling citation of visual elements: *Page Encoding* and *Modality Encoding*. Figure 3 illustrates the overall workflow with these two encoding methods.

4.1. Page Encoding

Inspired by VisRAG [23], we directly use document page images $\mathcal{P}$ as model input. Additionally, we use DocLayout [25] to detect visual elements on each page. The detected elements are annotated with bounding boxes, which are provided to the MLLM alongside the page image as auxiliary input.

The processed input is represented as:

$$\{(p_i, \text{BBoxList}_i)\}_{i=1}^n$$

where p_i denotes the page image of the i -th document page, which is encoded into visual tokens, and $\text{BBoxList}_i = \{b_i^{(1)}, b_i^{(2)}, \dots\}$ is the set of bounding box coordinates for visual elements in p_i , each encoded into text tokens. Each bounding box is assigned a unique visual element identifier, allowing the MLLM to cite the corresponding visual element during answer generation.

This input preserves all visual information present in the page image (e.g., layout). However, it imposes additional challenges for the MLLM: it must (i) interpret textual content directly from pixels (without explicit OCR) and (ii) align each visual-element identifier with its bounding box region on the page for accurate understanding and citation.

4.2. Modality Encoding

Unlike *Page Encoding*, we first extract text from document images via OCR and detect visual element regions and crop them according to their bounding boxes. The extracted text is then encoded into text tokens, and the cropped visual element images are encoded into visual tokens.

The input is represented as:

$$\{(t_i, \mathbf{V}_i)\}_{i=1}^n$$

where t_i denotes the extracted text from the i -th page, and $\mathbf{V}_i = \{v_i^{(1)}, v_i^{(2)}, \dots\}$ is the set of visual element crops derived from p_i . Each crop is assigned a unique visual-element identifier, which is used to cite the corresponding visual element during answer generation.

While this approach loses some visual information (e.g., layout) during processing, it enables fine-grained understanding by processing text and visual elements independently with modality-specific representations.

5. Experiments

5.1. Main Results

Implementations We adopt Qwen2.5-VL-7B [1] and train for 3 epochs with both encoding methods on the VinQA training split, conducted on 16xA100 GPUs.

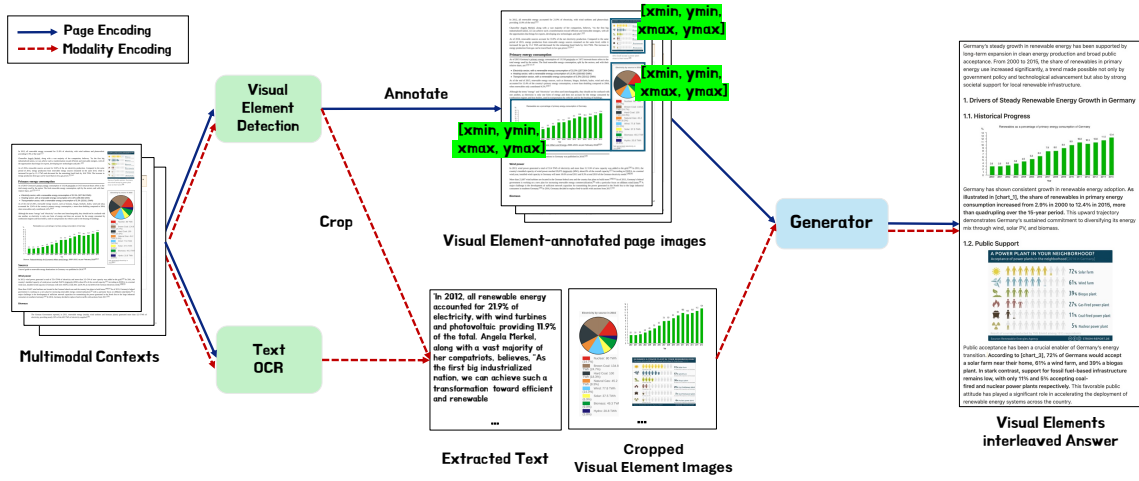


Figure 3. Visual Elements Interleaved Answer Generation based on encoding methods. The blue solid arrows represent the *Page Encoding* method, and the red dotted arrows represent the *Modality Encoding* method.

Model	M-GroSE					Visual Source
	Relevance	Completeness	Faithfulness	Unans. (F1)	Avg	F1
<i>Page Encoding</i>						
Proprietary Models						
GPT-4.1	4.88	3.68	4.27	0.78	3.46	0.62
GPT-4.1-mini	4.57	3.58	4.00	0.47	3.29	0.49
Gemini 2.0 Flash	4.19	2.60	3.62	0.86	2.86	0.46
Claude 3.5 Sonnet	4.83	3.81	4.46	0.70	3.53	0.65
Open-source Models						
InternVL3-8B	3.26	1.66	2.24	0.78	2.05	0.23
Qwen2.5-VL-7B	3.01	1.61	2.32	0.71	1.99	0.31
Qwen2.5-VL-7B (VinQA)	4.60	3.65	4.09	0.91	3.34	0.55
<i>Modality Encoding</i>						
Proprietary Models						
GPT-4.1	4.89	3.98	4.60	0.70	3.63 (+0.16)	0.72 (+0.10)
GPT-4.1-mini	4.63	3.94	4.22	0.76	3.45 (+0.16)	0.60 (+0.11)
Gemini 2.0 Flash	4.65	3.51	4.52	0.86	3.43 (+0.57)	0.61 (+0.14)
Claude 3.5 Sonnet	4.83	3.94	4.68	0.69	3.62 (+0.09)	0.70 (+0.06)
Open-source Models						
InternVL3-8B	3.48	1.89	2.57	0.77	2.24 (+0.19)	0.32 (+0.09)
Qwen2.5-VL-7B	2.96	1.81	2.53	0.75	2.08 (+0.09)	0.40 (+0.09)
Qwen2.5-VL-7B (VinQA)	4.59	3.59	4.09	0.90	3.33 (-0.02)	0.58 (+0.02)

Table 3. Overall performance on the VinQA test set. The values in parentheses show the difference between *Modality Encoding* and *Page Encoding*. The M-GroSE Avg score is computed as the mean of the three 1–5 criteria (relevance, completeness, and faithfulness) and the unanswerability F1 score, where the latter is linearly rescaled from [0,1] to [1,5] before averaging.

Evaluation We propose Multimodal Grounded QA Scoring Evaluator (M-GroSE), a modified version of GroUSE [14] originally designed for text-only long-form grounded answers, extended to assess multimodal grounding by incorporating visual citations. For answerable questions, we use GPT-5-mini as an LLM-judge to evaluate answers along three criteria: relevance (whether the answer addresses the

question), completeness (whether it includes all necessary information from the retrieved context), and faithfulness (whether it is grounded in the sources without hallucination). For unanswerable questions, we report Unanswerability F1, which measures whether the model correctly abstains when no valid answer exists in the provided context. To extend GroUSE to multimodal QA, we textualize both

textual and visual context as done in Section 3.1 and modify the LLM-judge prompt to evaluate both textual reasoning and visual citations with their supporting text.

In addition, we evaluate visual citation accuracy using Visual Source F1, following MCiteBench [9], by comparing predicted visual-element references with VinQA’s gold citations.

Overall Performance As shown in Table 3², on the VinQA test split, the best Page Encoding performance is achieved by Claude 3.5 Sonnet, with M-GroSE Avg of 3.53 and Visual Source F1 of 0.65, while the best Modality Encoding result comes from GPT-4.1 (M-GroSE Avg of 3.63, Visual Source F1 of 0.72), showing a clear gap between proprietary and open-source models. However, fine-tuning Qwen2.5-VL-7B on VinQA significantly improves both metrics; fine-tuned model reaches 3.34 (M-GroSE Avg), 0.55 (Visual Source F1) with Page Encoding and 3.33 (M-GroSE Avg), 0.58 (Visual Source F1) with Modality Encoding, markedly narrowing the gap to the SOTA. These results suggest that VinQA provides highly effective supervision for multimodal grounding and visual-element citation.

Notably, *Modality Encoding* generally outperforms *Page Encoding* on both M-GroSE Avg and Visual Source F1—as indicated by the positive deltas in parentheses in Table 3—except for models fine-tuned on VinQA. This indicates that while Modality Encoding is more effective prior to fine-tuning, Page Encoding can achieve comparable performance once fine-tuned on the dataset. We analyze this result in detail in Section 5.2.

5.2. Analysis

Does VinQA Improve Model Performance Across Context Token Length and Visual Element Diversity?

First, we analyze how VinQA fine-tuning affects M-GroSE scores of Qwen2.5-VL-7B with respect to context token length. Figure 4 (a) and (b) show the results for Modality Encoding and Page Encoding, respectively. We measure context token length under Modality Encoding, which scales with both textual volume and the number or resolution of visual elements on a page. It serves as a proxy for document complexity. Across all context-length bins (0–2.5k to ≥ 10 k), VinQA fine-tuning consistently improves multimodal grounded answer quality. However, performance decreases for very long contexts (≥ 7.5 k), suggesting that increased document complexity limits complete and faithful information integration.

Second, we analyze the gains in Visual Source F1 with respect to visual element types in context before and after VinQA fine-tuning. Figure 5 (a) and (b) show the results for Modality Encoding and Page Encoding, respectively. For

this analysis, we use the VinQA’s visual-element annotations—Chart, Table, and Figure³—to evaluate performance. The “Mixed” type covers answers required to include citations to two or more types of visual elements. Before fine-tuning, answers citing Figures perform worst among all types under Modality Encoding, while under Page Encoding both Figures and Tables lag behind the other types. After fine-tuning on VinQA, these categories improve substantially, narrowing their performance gap with the rest. Gains are also observed in the Mixed category, indicating that VinQA enhances citation accuracy even for answers that integrate multiple types of visual modalities.

Which is Better: Page Encoding or Modality Encoding?

We first compare the M-GroSE Avg score difference between *Modality Encoding* and *Page Encoding* across the context token length as shown in Figure 4 (c). Before VinQA fine-tuning, Page Encoding performs slightly better for short contexts (≤ 2.5 k), but Modality Encoding yields higher scores as context token length increases. This trend arises because Modality Encoding processes extracted text and cropped visual elements directly, whereas Page Encoding requires the model to interpret textual content from rendered page images and to reason over bounding-box-annotated visual regions. As the context token length increases, this burden on the model grows accordingly. After VinQA fine-tuning, however, this gap becomes negligible across all context lengths. This suggests that Page Encoding enables the model to generate multimodal grounded answers for complex contexts at a level comparable to Modality Encoding, despite not relying on explicit parsing.

We next compare Visual Source F1 difference between *Modality Encoding* and *Page Encoding* across visual element types, as shown in Figure 5 (c). Before VinQA fine-tuning, Modality Encoding achieves higher citation performance for Table and Mixed category. This is likely because tables contain more text than charts or figures, and Mixed requires referencing heterogeneous visual elements; representing cropped regions as separate visual tokens is therefore advantageous. After VinQA fine-tuning, these type-wise gaps narrow considerably, indicating that VinQA helps Page Encoding handle diverse visual elements more effectively, even when operating on page images with bounding-box regions.

Are the visual elements appropriately interleaved in the answer with faithful supporting text?

We have presented evaluation results using M-GroSE and Visual Source F1, but because these metrics operate on textualized context and answers, they are limited in assessing visual aspects. To directly evaluate visual factors, we introduce Visual G-Eval, which uses GPT-5 to score the question

²We evaluate recent state-of-the-art MLLMs, but exclude models that rely on test-time scaling for a fair comparison

³The Figure category includes photos, diagrams, and other non-chart/table visual elements.

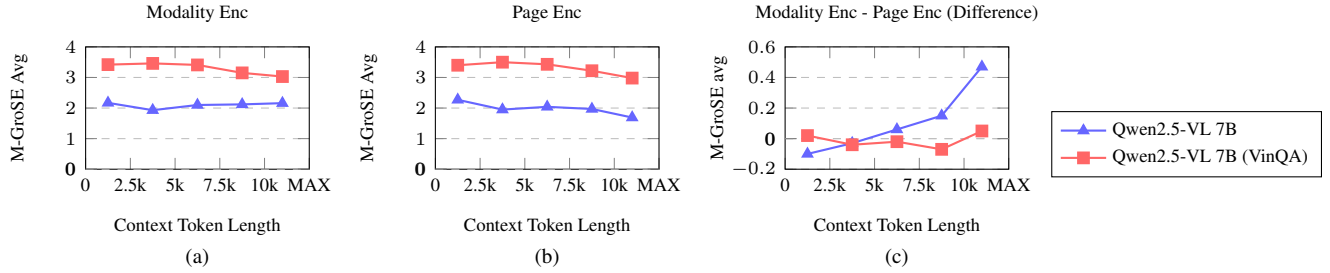


Figure 4. M-GroSE Avg performance across context token length.

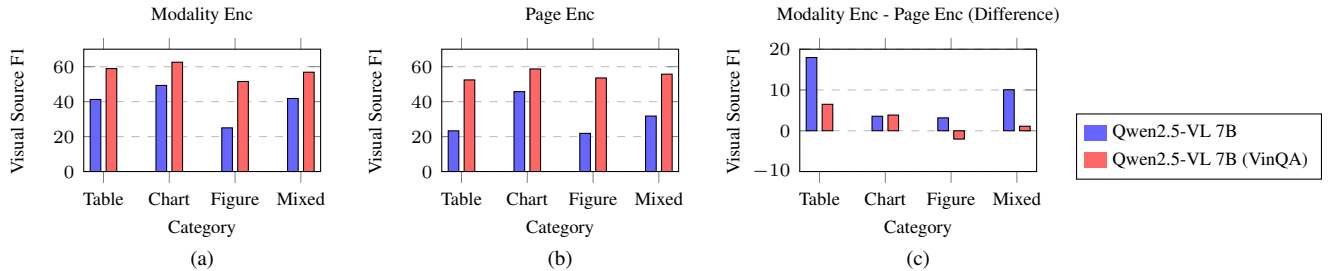


Figure 5. Visual source F1 across various visual element types

and generated answer while consuming the cited visual elements as image inputs. We score each citation on three 1–5 metrics: (1) Effectiveness – how well the cited image, together with its supporting text, visually supports the answer; (2) Position – how appropriately the image is placed within the answer; and (3) Faithfulness – how accurately the supporting text reflects the visual content.

Table 4 reports average scores over all visual-element citations across 200 instances sampled from the VinQA test split. Training with VinQA yields consistent gains on all three criteria for both encoding methods, indicating that VinQA provides an effective training signal for learning to better leverage visual elements. Interestingly, Modality Encoding attains relatively high Effectiveness and Faithfulness scores both before and after training. While VinQA training allows Page Encoding to match Modality Encoding in overall multimodal grounded answer performance as measured by M-GroSE, Modality Encoding still retains an advantage in how effectively it uses and explains visual elements to support the answer.

6. Conclusion

We introduce the VinQA dataset for generating visual-element-interleaved long-form answers over real-world multimodal documents, together with Page/Modality Encoding and multimodal evaluation methods (M-GroSE, Visual Source F1, Visual G-Eval). Fine-tuning open Qwen2.5-VL models on VinQA training split substantially improves multimodal grounding and makes both encoding

Model	Visual G-Eval		
	Effectiveness	Position	Faithfulness
<i>Page Encoding</i>			
Qwen2.5-VL-7B	3.44	4.24	3.44
Qwen2.5-VL-7B (VinQA)	3.94	4.77	3.85
<i>Modality Encoding</i>			
Qwen2.5-VL-7B	4.00	4.57	4.01
Qwen2.5-VL-7B (VinQA)	4.17	4.74	4.23

Table 4. Visual G-Eval results of Qwen2.5-VL-7B before and after VinQA fine-tuning under Page Encoding and Modality Encoding.

strategies competitive, with Page Encoding in particular achieving strong gains even without explicit parsing. As future work, we plan to apply VinQA to models with test-time reasoning (e.g., long think-token generation) and to mitigate performance degradation on more complex, longer-context documents.

References

- [1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhao-hai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025. 1, 2, 5
- [2] Yew Ken Chia, Liying Cheng, Hou Pong Chan, Chaoqun Liu, Maojia Song, Sharifah Mahani Aljunied, Soujanya Poria, and Lidong Bing. M-longdoc: A benchmark for multimodal super-long document understanding and a retrieval-aware tuning framework, 2024. 1, 2, 4
- [3] Jaemin Cho, Debanjan Mahata, Ozan Irsoy, Yujie He, and Mohit Bansal. M3docrag: Multi-modal retrieval is what you need for multi-page multi-document understanding, 2024. 1, 2
- [4] Chao Deng, Jiale Yuan, Pi Bu, Peijie Wang, Zhong-Zhi Li, Jian Xu, Xiao-Hui Li, Yuan Gao, Jun Song, Bo Zheng, et al. Longdocurl: a comprehensive multimodal long document benchmark integrating understanding, reasoning, and locating. *arXiv preprint arXiv:2412.18424*, 2024. 1, 2, 4
- [5] Yihao Ding, Kaixuan Ren, Jiabin Huang, Siwen Luo, and Soyeon Caren Han. Mvqa: A dataset for multimodal information retrieval in pdf-based visual question answering. *arXiv preprint arXiv:2404.12720*, 2024. 1, 2
- [6] Kuicai Dong, Yujing Chang, Xin Deik Goh, Dexun Li, Ruiming Tang, and Yong Liu. Mmdocir: Benchmarking multi-modal retrieval for long documents. *arXiv preprint arXiv:2501.08828*, 2025. 3
- [7] Manuel Faysse, Hugues Sibille, Tony Wu, Bilel Omrani, Gautier Viaud, Céline Hudelot, and Pierre Colombo. Colpali: Efficient document retrieval with vision language models, 2025. 3
- [8] Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6465–6488, Singapore, 2023. Association for Computational Linguistics. 3
- [9] Caiyu Hu, Yikai Zhang, Tinghui Zhu, Yiwei Ye, and Yanghua Xiao. Mcitebench: A benchmark for multimodal citation text generation in mllms, 2025. 2, 3, 4, 7
- [10] Goeric Huybrechts, Srikanth Ronanki, Sai Muralidhar Jayanthi, Jack Fitzgerald, and Srinivasan Veeravanallur. Document haystack: A long context multimodal image/document understanding vision llm benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4062–4070, 2025. 2, 4
- [11] Nelson Liu, Tianyi Zhang, and Percy Liang. Evaluating verifiability in generative search engines. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7001–7025, Singapore, 2023. Association for Computational Linguistics. 3
- [12] Xueguang Ma, Shengyao Zhuang, Bevan Koopman, Guido Zuccon, Wenhui Chen, and Jimmy Lin. VISA: Retrieval augmented generation with visual source attribution. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 30154–30169, Vienna, Austria, 2025. Association for Computational Linguistics. 3, 4
- [13] Yubo Ma, Yuhang Zang, Liangyu Chen, Meiqi Chen, Yizhu Jiao, Xinze Li, Xinyuan Lu, Ziyu Liu, Yan Ma, Xiaoyi Dong, et al. Mmlongbench-doc: Benchmarking long-context document understanding with visualizations. *Advances in Neural Information Processing Systems*, 37:95963–96010, 2025. 1, 2, 3, 4
- [14] Sacha Muller, Antonio Loison, Bilel Omrani, and Gautier Viaud. GroUSE: A benchmark to evaluate evaluators in grounded question answering. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 4510–4534, Abu Dhabi, UAE, 2025. Association for Computational Linguistics. 2, 6
- [15] Manan Suri, Puneet Mathur, Franck Dernoncourt, Kanika Goswami, Ryan A. Rossi, and Dinesh Manocha. Visdom: Multi-document qa with visually rich elements using multimodal retrieval-augmented generation, 2025. 1, 2, 3, 4
- [16] Ryota Tanaka, Kyosuke Nishida, Kosuke Nishida, Taku Hasegawa, Itsumi Saito, and Kuniko Saito. Slidevqa: A dataset for document visual question answering on multiple images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 13636–13645, 2023. 3, 4
- [17] Ryota Tanaka, Taichi Iki, Taku Hasegawa, Kyosuke Nishida, Kuniko Saito, and Jun Suzuki. Vdocrag: Retrieval-augmented generation over visually-rich documents. In *CVPR*, 2025. 2, 4
- [18] Qiuchen Wang, Ruixue Ding, Zehui Chen, Weiqi Wu, Shihang Wang, Pengjun Xie, and Feng Zhao. Vidorag: Visual document retrieval-augmented generation via dynamic iterative reasoning agents, 2025. 4
- [19] Haolong Yan, Kaijun Tan, Yeqing Shen, Xin Huang, Zheng Ge, Xiangyu Zhang, Si Li, and Daxin Jiang. M-docsum: Do llms genuinely comprehend interleaved image-text in document summarization? *arXiv preprint arXiv:2503.21839*, 2025. 2, 3, 4
- [20] Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. A survey on multimodal large language models. *National Science Review*, 11(12): nwa403, 2024. 1
- [21] Qinhan Yu, Zhiyou Xiao, Binghui Li, Zhengren Wang, Chong Chen, and Wentao Zhang. Mramg-bench: A comprehensive benchmark for advancing multimodal retrieval-augmented multimodal generation, 2025. 3
- [22] Qinhan Yu, Zhiyou Xiao, Binghui Li, Zhengren Wang, Chong Chen, and Wentao Zhang. Mramg-bench: A beyond-text benchmark for multimodal retrieval-augmented multimodal generation. *arXiv preprint arXiv:2502.04176*, 2025. 2, 4
- [23] Shi Yu, Chaoyue Tang, Bokai Xu, Junbo Cui, Junhao Ran, Yukun Yan, Zhenghao Liu, Shuo Wang, Xu Han, Zhiyuan Liu, and Maosong Sun. Visrag: Vision-based retrieval-augmented generation on multi-modality documents, 2025. 1, 2, 3, 4, 5
- [24] Jiajie Zhang, Yushi Bai, Xin Lv, Wanjuan Gu, Danqing Liu, Minhao Zou, Shulin Cao, Lei Hou, Yuxiao Dong, Ling Feng,

- and Juanzi Li. Longcite: Enabling llms to generate fine-grained citations in long-context qa, 2024. [3](#)
- [25] Zhiyuan Zhao, Hengrui Kang, Bin Wang, and Conghui He. Doclayout-yolo: Enhancing document layout analysis through diverse synthetic data and global-to-local adaptive perception, 2024. [3](#), [5](#)
- [26] Fengbin Zhu, Wenqiang Lei, Fuli Feng, Chao Wang, Haozhou Zhang, and Tat-Seng Chua. Towards complex document understanding by discrete reasoning. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4857–4866, 2022. [3](#)
- [27] Fengbin Zhu, Ziyang Liu, Xiang Yao Ng, Haohui Wu, Wenjie Wang, Fuli Feng, Chao Wang, Huanbo Luan, and Tat Seng Chua. Mmdocbench: Benchmarking large vision-language models for fine-grained visual document understanding. *arXiv preprint arXiv:2410.21311*, 2024. [1](#), [2](#)
- [28] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, Zhangwei Gao, Erfei Cui, Xuehui Wang, Yue Cao, Yangzhou Liu, Xingguang Wei, Hongjie Zhang, Haomin Wang, Weiye Xu, Hao Li, Jiahao Wang, Nianchen Deng, Songze Li, Yinan He, Tan Jiang, Jiapeng Luo, Yi Wang, Conghui He, Botian Shi, Xingcheng Zhang, Wenqi Shao, Junjun He, Yingtong Xiong, Wenwen Qu, Peng Sun, Penglong Jiao, Han Lv, Lijun Wu, Kaipeng Zhang, Huipeng Deng, Jiaye Ge, Kai Chen, Limin Wang, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. Internv13: Exploring advanced training and test-time recipes for open-source multimodal models, 2025. [1](#)