ViG-LLM: Enhancing Visual Grounding Capabilities in Closed-Box LLMs for Document Information Extraction without OCR Dependencies

Sudhanshu Bhoi
Amazon
Hyderabad, India
sudhbee@amazon.com

Abstract

Large Language Models (LLMs) have shown remarkable capabilities in document processing, but their inability to provide visual grounding without OCR dependencies poses significant challenges in business-critical applications. Current solutions either require model fine-tuning or rely on external OCR services, introducing additional costs, latency, and limitations in handling derived information. This paper presents ViG-LLM, a novel framework that enables closed-box LLMs to generate localization information through a multi-agent system combining U-Net-based layout deconstruction with viewport identification tasks. Evaluated on the FATURA and CORD dataset, our framework achieves perfect accuracy over spatial reasoning tuned LLM like Amazon Nova Pro, while demonstrating superior template-specific consistency. The framework maintains robust performance across LLM architectures while maintaining comparable operational costs and latency to enterprise OCR-based solutions with efficient foundation models. In real-world document processing applications, the framework helps retain the high reasoning capabilities of the system in document information extraction tasks while improving explainability, reliability and human interaction for information verification. Through human-in-the-loop learning and closed-box prompt alignment techniques, ViG-LLM provides a robust, adaptable solution for visual grounding tasks in document processing workflows.

1 Introduction

Large Language Models (LLMs) have demonstrated remarkable multi-modal capabilities across diverse domains, including healthcare, autonomous systems, customer experience, and creative industries. However, in business-critical applications such as medical imaging analysis, scene understanding, visual troubleshooting, and visual question answering, it is imperative that the information extracted by LLMs is visually grounded. This requirement is particularly crucial in domains like financial, legal, and medical document processing, where information accuracy and verifiability directly impact decision-making and compliance.

Multi-modal LLMs (MLLMs) have proven especially effective in handling complex document layouts, interpreting visual elements like charts and graphs, and processing low-quality or semi-structured documents. These models enable sophisticated Document Visual Question Answering (VQA) by allowing users to query document content while providing both answers and visual evidence. Such visual grounding capabilities are essential for critical tasks including human-in-the-loop validation or Personally Identifiable Information (PII) detection, content redaction.

Despite their advanced text generation, analysis, and reasoning capabilities, LLMs exhibit significant limitations in producing grounding information. Current approaches typically rely on external Optical Character Recognition (OCR) technologies, introducing additional costs, latency, and limitations in handling derived information. These limitations, as demonstrated in figure 1, become particularly apparent in several scenarios: when information must be synthesized from multiple sources, when processing diagrammatic or figurative content, when extracting specific objects like QR codes, and when analyzing long-form text. Alternative methods involving LLM fine-tuning assume model transparency, making them unsuitable for proprietary LLMs.

This paper presents a novel ViG-LLM framework that enables **Large Language Models (LLM)** to generate **Visual Grounding (ViG)** information without external dependencies, accommodating their closed-box nature. We provide a comprehensive evaluation of our framework against existing techniques, measuring localization accuracy, performance consistency, and operational-effectiveness. Furthermore, our framework incorporates black-box prompt alignment techniques with human-in-the-loop (HITL) capabilities for bounding box corrections, addressing tail-end accuracy challenges.

Our key contributions are as follows:

- We present ViG-LLM, a visual grounding framework that enables localization capabilities in closed-box LLMs without external dependencies using multi-agent viewport identification with U-Net segmentation to achieve OCR-independent bounding box generation.
- We demonstrate superior accuracy and consistency in document processing tasks, achieving perfect accuracy over spatial reasoning tuned MLLMs while maintaining comparable operational metrics to enterprise OCR-based solutions and showing robust performance across multiple LLM architectures.
- We present comprehensive ablation studies examining

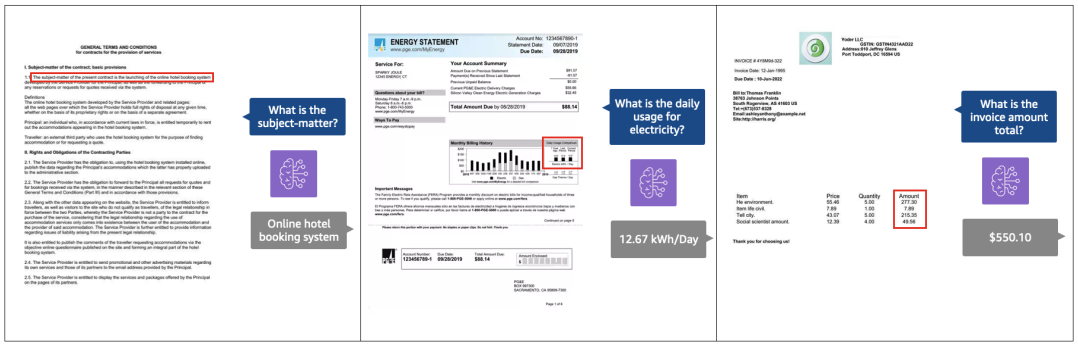

Figure 1: Illustration of complex document information extraction scenarios challenging OCR-dependent LLM approaches. Three critical cases demonstrate the need for direct visual grounding: (1) extraction from long-form text where target values span partial paragraphs, (2) interpretation of diagrammatic content requiring contextual understanding, and (3) synthesis of information from multiple table rows. Red annotations highlight source regions from which the LLM derives field values, illustrating the complexity of establishing precise visual references.

individual framework components, systematically validating their contributions to the overall system performance.

2 Related Work

Recent approaches to document information extraction using Large Language Models (LLMs) can be categorized into open-box and closed-box techniques, each with distinct advantages and limitations. Open-box techniques, exemplified by UIPath (Blog), TextMonkey (Liu et al. 2024), and DocLLM (Wang et al. 2023), require direct access to model internals and fine-tuning capabilities. While these approaches offer greater control and customization, they often demand substantial computational resources and expertise, making them impractical for applications utilizing proprietary LLM APIs. In contrast, closed-box techniques treat LLMs as closed systems and have demonstrated success across various domains, including legal contract analysis (Zhao and Gao 2024) and research paper information extraction (Polak and Morgan 2024), primarily through semantic parsing and prompt engineering. The widespread adoption of closed-box approaches in business-critical applications can be attributed to their superior performance, rapid deployment capabilities, reduced infrastructure requirements, and enhanced reliability and security guarantees.

The ability to produce localization information presents a significant differentiator between these approaches. Open-box techniques can be specifically trained or fine-tuned to generate such information, as demonstrated by model families like Google Gemini (goo) and Amazon Nova (ama). However, prominent reasoning models such as Anthropic Claude have shown limitations in producing localization information directly (cla). To address this limitation in closed-box approaches, researchers have integrated OCR techniques, either by encoding text information during LLM input (Lu et al. 2025) or implementing post-inference matching (Sinha and S 2025). These hybrid approaches, however, introduce several critical challenges: potential information loss and OCR noise affecting LLM inference quality, complex matching logic requirements - particularly challenging for multi-lingual documents, additional computational overhead, and limited control over localization granularity (Kanerva et al. 2025),(Boros et al. 2022),(vel),(llm),(Loukil et al. 2024),(Zhang et al. 2025). This work presents a novel framework for enabling localization capabilities in closed-box LLMs without OCR dependencies.

Document segmentation, a well-established preprocessing technique in OCR applications (Ariki and Motegi 1995), decomposes a document image into discrete elements. Segmentation approaches can be broadly classified into geometric and layout/semantic-based methods, with configurable granularity based on specific task requirements (Eskenazi, Gomez-Krämer, and Ogier 2016). While transformer-based models like LayoutLM have shown promising results in layout-based segmentation (Xu et al. 2020), they have been superseded in document information extraction tasks by LLMs due to their superior interpretability (Bhattacharyya et al. 2025). In the domain of geometric segmentation, U-Net architectures have emerged as the state-of-the-art solution for high-precision boundary segmentation, outperforming traditional computer vision techniques such as X-Y Cut and Run-Length Smearing Algorithm (RLSA) (Plaksyvyi, Skublewska-Paszkowska, and Powroźnik 2023),(Soujanya et al. 2013). Originally developed for medical image segmentation (Ronneberger, Fischer, and Brox 2015), U-Nets have been successfully adapted for various document analysis tasks, including geometric layout segmentation (Mechi et al. 2019) and logical layout analysis for key-value pair extraction (Mohammadshirazi et al. 2024). The architecture's demonstrated flexibility across different segmentation granularities (Sivasubramanian, Mohan, and Sowmya 2024) makes it particularly suitable for block segmentation tasks, as implemented in this work.

Human-in-the-Loop (HITL) methods in LLM systems encompass techniques for active elicitation (Settles 2010), feedback-driven adaptation (Christiano et al. 2023), and post-hoc correction (Ribeiro, Singh, and Guestrin 2018) to improve model performance and reliability. These tech-

niques are broadly categorized into pre-generation (Zhang et al. 2024), in-generation (Xiao et al. 2025), and post-generation (Gutowska 2025) interventions based on the stage of human involvement in the interaction pipeline. In closed-box LLM settings, where direct access to model parameters is restricted, specialized HITL approaches become necessary as traditional gradient-based fine-tuning and internal uncertainty metrics are inaccessible. These closed-box techniques primarily focus on prompt-level and output-level interventions, including prompt engineering loops (Shah 2024), output ranking schemes (Lee and Shin 2024), and external reward modeling (Sahoo et al. 2025). Within this, closed-box prompt alignment techniques like Black-Box Prompt Optimization (BPO) (Cheng et al. 2024) emerges as a critical pre-generation approach, offering automated and scalable methods for leveraging human feedback to improve model behavior without direct intervention in every interaction. BPO techniques span across evolutionary algorithms (e.g., EvoPrompt (Guo et al. 2025)), language model feedback-based optimization (e.g., ProTeGi (Pryzant et al. 2023)), sampling and resampling approaches (e.g., APE (Zhou et al. 2023)), and trajectory-informed optimization (e.g., OPRO (Yang et al. 2024)). Each approach offers distinct advantages: evolutionary algorithms for exploration, feedback-based methods for human preference alignment, sampling approaches for computational efficiency, and trajectory-informed methods for optimization stability. In this work, we propose to leverage these techniques for prompt alignment in business-specific settings, aiming to optimize prompt effectiveness while maintaining the closed-box nature of the underlying LLM.

3 Approach

The ViG-LLM framework decomposes localization into separate horizontal and vertical viewport identification tasks, employing targeted prompting and visual aids through image processing. Additionally, it incorporates human feedback for continuous improvement. Figure 2 provides an overview of the end-to-end framework, while Algorithm 1 details the control flow.

3.1 Visual Layout Deconstruction

The initial stage employs a U-Net model trained for block segmentation generation. For detailed U-Net architecture, refer to Supplmentary Section 6.1. This trainable segmentation component enables controlled granularity in localization tasks. The Binary Cross Entropy (BCE) loss function $\mathcal{L}_{BCE}$ used to train the model is defined in Equation below.

$$L_{i,j} = y_{i,j} \log(p_{i,j}) + (1 - y_{i,j}) \log(1 - p_{i,j})$$

$$\mathcal{L}_{BCE} = -\frac{1}{H \cdot W} \sum_{i=1}^{H} \sum_{j=1}^{W} L_{i,j}$$

Where H and W are the height and width of the input image and the corresponding output segmentation mask; $y_{i,j}$ is the true binary label for the pixel at position (i, j), which can be either 0 (background) or 1 (foreground); $p_{i,j}$ is the predicted probability for the pixel at position (i, j) belonging to the foreground class, as output by the final sigmoid

Algorithm 1: ViG-LLM Framework - Document Value Visual Grounding Algorithm

Notation:
$\mathcal{I}$: Input document image
r: Reference text string
ψ_v, ψ_h: Vertical and horizontal prompts
$\mathcal{B}$: Set of bounding boxes, $\mathcal{B} = \{b_i\}_{i=1}^{n}$
$\mathcal{G}$: Grid structure, $\mathcal{G} = (X, Y)$
δ: Coordinate merge threshold
$\mathcal{M}$: Segmentation mask
$\mathcal{C}$: Set of contours
Ω_h, Ω_v: Horizontal and vertical viewport spans
$\mathcal{V}$: Viewport region
b^*: Output bounding box

function GENERATE_GRID($\mathcal{B}, \delta$)
  $(X, Y) \leftarrow$ GET_UNIQUE_COORDINATES($\mathcal{B}$)
  $(X', Y') \leftarrow$ MERGE_CLOSE_COORDINATES(X, Y, δ)
  return $\mathcal{G}(X', Y')$
end function
procedure DOCUMENTVALUELOCALIZATION
  Input: $\mathcal{I}, r, \psi_v, \psi_h$
  Output: b^*
  ▷ Visual Layout Deconstruction
  $\mathcal{M} \leftarrow$ SEGMENT_IMAGE($\mathcal{I}$)
  $\mathcal{C} \leftarrow$ FIND_CONTOURS($\mathcal{M}$)
  $\mathcal{B} \leftarrow$ FIND_BOUNDING_BOXES($\mathcal{C}$)
  ▷ Horizontal Viewport Identification
  $\mathcal{G} \leftarrow$ GENERATE_GRID($\mathcal{B}$)
  $L_h \leftarrow \mathcal{G}.Y$
  $P_h \leftarrow$ PROJECT_HORIZONTAL_LINES($L_h, \mathcal{I}$)
  $\Omega_h \leftarrow$ INVOKE_HVIA(P_h, r, ψ_h)
  repeat
    $\mathcal{V} \leftarrow$ CLIP_VIEWPORT_PROJECTION($\Omega_h, \mathcal{I}$)
    $\gamma \leftarrow$ INVOKE_VGVA($\mathcal{V}, r$)
  until $\gamma = true$
  ▷ Vertical Viewport Identification
  $\mathcal{B}' \leftarrow$ FILTER_BOXES_IN_BOUNDS($\mathcal{B}, \Omega_h$)
  $\mathcal{G}' \leftarrow$ GENERATE_GRID($\mathcal{B}'$)
  $L_v \leftarrow \mathcal{G}'.X$
  $P_v \leftarrow$ PROJECT_VERTICAL_LINES($L_v, \mathcal{I}$)
  $\Omega_v \leftarrow$ INVOKE_VVIA(P_v, r, ψ_v)
  repeat
    $\mathcal{V} \leftarrow$ CLIP_VIEWPORT_PROJECTION($\Omega_v, \mathcal{V}$)
    $\gamma \leftarrow$ INVOKE_VGVA($\mathcal{V}, r$)
  until $\gamma = true$
  $b^* \leftarrow$ SELECT_BOUNDING_BOX_IN_BOUNDS($\mathcal{B}', \Omega_v$)
  ▷ Learning
  if requires_validation **then**
    $b_g \leftarrow$ GET_HUMAN_VALIDATION($\mathcal{I}, r, b^*$)
    $\psi_v', \psi_h' \leftarrow$ ALIGN_PROMPTS($\mathcal{I}, r, \psi_v, \psi_h, b_g$)
  end if
  return b^*
end procedure

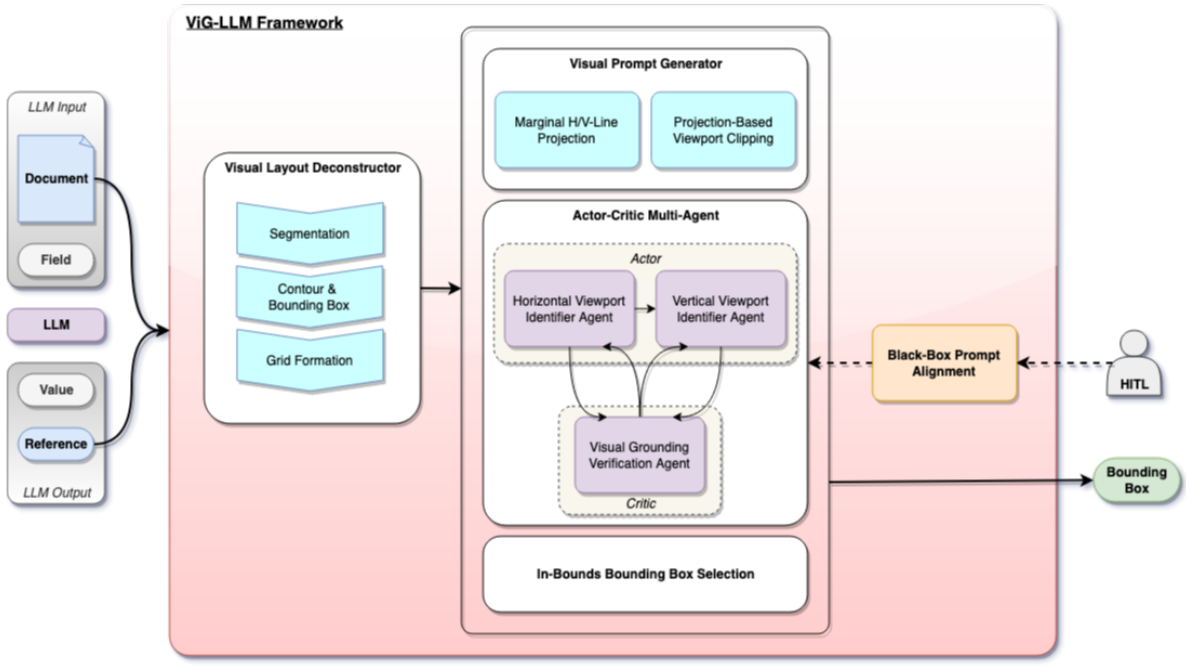

Figure 2: Architecture overview of the ViG-LLM framework. The system comprises three main components: (1) Visual Layout Deconstruction employing U-Net segmentation for grid formation, (2) Multi-Agent LLM system performing viewport identification through visual prompting, and (3) Human-in-the-Loop learning for accuracy refinement. Arrows indicate the flow of information through the system, with dotted lines representing optional feedback paths for continuous improvement.

activation layer of the U-Net.

The trained U-Net generates segmentation masks for input document images. Connected components within these masks are identified using contour detection algorithm like border-following algorithm (Suzuki and be 1985), from which rectangular bounding boxes are derived. The coordinates of these boxes are decomposed into x-axis and y-axis components to create a grid structure. To reduce complexity, lines within a specified threshold δ are merged while preserving individual component boundaries.

3.2 Reference Localization using Multi-Agent LLM

The framework employs a multi-agent approach to decompose localization into sequential viewport identification tasks. For horizontal viewport identification, the system extracts y-axis coordinates from the grid and generates an overlay of labeled horizontal lines on the original image. The Horizontal Viewport Identification Agent (HVIA) processes this enhanced image to determine the appropriate row span for the input reference. The original image is then clipped to this horizontal viewport.

Vertical viewport identification follows a similar process but focuses only on x-coordinates from bounding boxes within the identified row span. The system overlays labeled vertical lines on the clipped horizontal viewport image. The Vertical Viewport Identification Agent (VVIA) then determines the column span, resulting in the final localization bounding box. Figure 3 illustrates the progressive image processing techniques applied throughout this process.

The framework incorporates a critic Visual Grounding Verification Agent (VGVA) to validate reference presence within identified viewports. When a reference is not found, the critic agent determines its location relative to the current viewport, enabling a binary search approach through actor-critic interaction.

3.3 Human-In-The-Loop Learning

The framework implements a feedback loop incorporating human validation of output bounding boxes. When corrections are provided, the system adapts through multiple potential mechanisms as per the business need. In repetitive template document settings, for example, the system can employ Retrieval Augmented Generation (RAG) to identify document templates through segmentation mask and leverage these examples for In-Context Learning (ICL). Alternative approaches for Closed-Box Prompt Alignment (Mahmud et al. 2025) of the Viewport Identification Agents (HVIA & VVIA) can be integrated based on specific requirements. These prompt alignment techniques prove particularly valuable in handling document ambiguities and adapting to domain-specific variations, thereby enhancing the framework's robustness across diverse document types.

This architecture ensures continuous improvement in localization accuracy while maintaining flexibility for various document processing scenarios. The combination of visual

Figure 3: Sequential stages of viewport identification in the ViG-LLM framework. The progression demonstrates the system's visual grounding process: (1) input document image, (2) segmentation mask generated by the trained U-Net model, (3) horizontal line overlay derived from grid coordinates, (4) horizontal viewport selection with subsequent vertical line overlay in the clipped region, and (5) final bounding box localization.

layout analysis, multi-agent LLM coordination, and human feedback creates a robust system for visual grounding tasks.

4 Experiments

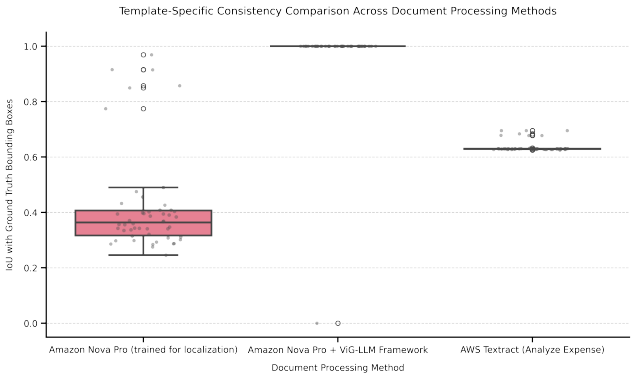

Figure 4: Comparative analysis of template-specific consistency across document processing methods. Performance evaluation depicts Intersection over Union (IoU) scores for (a) standalone Amazon Nova Pro trained for localization, (b) ViG-LLM framework integrated with Amazon Nova Pro and (c) AWS Textract Analyze Expense API. Results demonstrate consistency across multiple instances of identical templates, with box plots indicating median performance, quartile distribution, and outliers for each method. Higher IoU scores and smaller variance indicate superior consistency in visual grounding capabilities as showcased by the ViG-LLM framework.

4.1 Dataset

The experimental evaluation utilized two distinct datasets: the FATURA dataset (Limam, Dhiaf, and Kessentini 2023) and the CORD dataset (Park et al. 2019). FATURA contains 10,000 synthetic invoice images generated across 50 templates, with comprehensive bounding box annotations for all invoice components. CORD comprises 11,000 Indonesian receipts with annotated bounding boxes, categorized into 8 superclasses and 54 subclasses. The combined dataset was partitioned using an 80:10:10 split for training, validation, and testing respectively. Segmentation masks were generated from the annotated bounding boxes to train the U-Net segmentation model. For a preview of the datasets, refer to Supplementary Section. The U-Net architecture, consisting of encoding & decoding layers, was trained for 300 epochs, achieving 99% accuracy on the training and validation sets.

4.2 Methodology

The framework's efficacy was assessed through comprehensive comparisons with state-of-the-art visual grounding techniques, incorporating both OCR-based systems (like AWS Textract) and various Multimodal Large Language Model (MLLM) architectures trained for localization output. The evaluation centered on standardized fields common to both the Textract Analyze Expense response model and the ground truth dataset. For MLLM-based localization, we implemented prompting techniques adhering to established protocols, for example, utilizing scaled coordinates (0-1000) for output generation for the Amazon Nova models (noa 2025). To isolate the performance of the actor agent, we disabled the critic agent, eliminating its corrective contributions and allowing only a single pass through the actor agent. Furthermore, to mitigate uncertainties arising from the primary document information extraction LLM's interpretation of semantically similar terms (e.g., "total," "subtotal," "amount due," "total in words"), we considered the maximum Intersection over Union (IoU) across all such fields. The complete mathematical formulation of accuracy metrics is provided in Supplementary Section 6.3. All coordinate outputs were normalized to a standard format to ensure consistent comparison across methods.

To assess the framework's robustness across different model architectures, experiments were conducted using multiple LLM variants across the Viewport Identifier Agents (HVIA & VVIA). The evaluation encompassed two model

Table 1: Performance Comparison of Document Processing Methods: ViG-LLM Framework demonstrates superior consistency across IoU thresholds, achieves over 90% accuracy for FATURA and CORD datasets with Claude models and significantly outperforms standalone LLM across all architectures.

Approach	Method	Accuracy @ IoU Threshold					
		FATURA			CORD		
		0.25	0.5	0.75	0.25	0.5	0.75
OCR	Amazon Textract (AnalyzeExpense)	**1**	**1**	0.27	**0.96**	**0.96**	**0.96**
	LayoutLMv3 (DocVQA)	0.56	0	0	0.06	0	0
MLLM	Claude Sonnet 4	**0.96**	0.51	0.33	0.31	0.08	0.05
	Claude Opus 4.1	**0.97**	0.7	0.47	0.38	0.14	0.11
	Nova Pro v1	0.8	0.1	0.07	0.27	0.06	0.06
	Nova Premium v1	0.78	0.63	0.09	0.09	0	0
	Qwen2.5-VL-72B-Instruct	**1**	0.63	0.11	0.31	0	0
ViG-LLM (Ours)	Claude Sonnet 4 + ViG-LLM	**0.94**	**0.94**	**0.94**	**0.96**	**0.92**	**0.92**
	Claude Opus 4.1 + ViG-LLM	**1**	**1**	**1**	**0.9**	**0.9**	**0.9**
	Nova Pro v1 + ViG-LLM	**0.91**	**0.91**	**0.91**	0.72	0.67	0.67
	Nova Premier v1 + ViG-LLM	**1**	**1**	**1**	0.88	0.88	0.88

Table 2: Ablation Analysis of ViG-LLM Components: Study showing the impact of different ViG-LLM components on accuracy performance at 0.75 IoU threshold. Starting from the baseline with no ViG-LLM, each row demonstrates the cumulative effect of adding components: Visual Layout Deconstructor (VLD), Horizontal/Vertical Viewport Identification Agent (H/VVIA), Visual Grounding Verification Agent (VGVA), and Human-in-the-Loop (HITL) refinement. Results show progressive improvements in accuracy, reaching perfect performance with all components combined.

ViG-LLM Components	**0.75**
No ViG-LLM	0.33
VLD + Single VIA	0.82
VLD + HVIA + VVIA	0.94
VLD + HVIA + VVIA + VGVA	0.98
VLD + HVIA + VVIA + VGVA + HITL	1

families - Claude and Amazon Nova - with two different model sizes within each family. This analysis aimed to demonstrate the framework's independence from specific LLM capabilities.

Template-specific consistency was evaluated by analyzing performance across multiple instances of the same template within the FATURA dataset. The comparison involved the ViG-LLM Framework, Amazon Nova trained for localization, and AWS Textract as the baseline. To minimize variables, Amazon Nova Pro was utilized within the ViG-LLM Framework for this analysis.

4.3 Results

The experimental results demonstrate the effectiveness of the ViG-LLM Framework across multiple performance metrics. Table 1 shows that the framework achieves accuracy at par with AWS Textract while significantly outperforming the fine-tuned LLM. The framework exhibited consistent performance across different LLM families and model sizes, as shown in Table 1. Template-specific analysis, presented in Figure 4, reveals superior consistency in the framework's performance compared to standalone implementations.

Operational metrics were evaluated based on per-field localization in single images. With Amazon Nova Pro, the framework consumed an average of 4K input tokens and 0.3K output tokens per analysis, resulting in an operational cost of $4.16 per 1,000 document pages (at $0.0008 per 1K input token and $0.0032 per 1K output token (noa)). Similarly, processing times with Amazon Nova and Qwen2.5-VL-72B models averaged 3-5 seconds per document. These metrics demonstrate performance comparable to typical cloud-based OCR solutions.

However, both cost and latency metrics can substantially vary with alternative foundation models. For example, the Claude family of models exhibited increased operational costs and processing times: Claude Sonnet required $16.50 per 1,000 documents with 7-10 seconds processing time per document, while Claude Opus incurred $82.50 per 1,000 documents with 16-20 seconds per document. These variations in operational characteristics are primarily attributed to the underlying architectural differences among foundation models. Future optimization opportunities exist through the implementation of specialized Small Language Models (SLMs), which could potentially improve both cost efficiency and processing speeds while maintaining performance quality.

Further, ablation studies were conducted to evaluate individual components of the ViG-LLM framework. Components were sequentially incorporated, and accuracy metrics were measured at a 0.75 IoU threshold. Table 2 demonstrates the systematic contribution of each framework component to accuracy improvements, with final performance gains achieved through human-in-the-loop refinements.

Thus, the segmentation model successfully established appropriate localization granularity, while the grid formation and visual prompting techniques effectively simplified the localization task. The multi-agent system, including the critic component, demonstrated robust validation and correction capabilities. The integration of human-in-

the-loop refinements enabled further tail-end accuracy improvements. These components collectively contributed to the framework's accuracy, repeatability, and efficiency in visual grounding tasks.

5 Conclusion

This paper introduces ViG-LLM, a framework enabling visual grounding capabilities in closed-box LLMs without OCR dependencies. Through a multi-agent system and U-Net-based layout deconstruction, our approach effectively decomposes complex document information localization into manageable viewport identification tasks. Experimental results on the FATURA and CORD datasets demonstrate our framework's superior performance, showing improved accuracy and consistency across different LLM architectures while maintaining comparable cost and latency to OCR-based solutions when leveraging optimized language models. In a finance document processing pipeline, the framework helped improve the verifiability by human analysts of the LLM-based document information extraction system. The incorporation of human-in-the-loop learning ensures continuous improvement while maintaining adaptability across various document types. This flexibility, combined with closed-box compatibility, makes ViG-LLM particularly valuable for enterprise applications using proprietary LLM APIs. Future work could explore handling more complex document structures and cross-lingual visual grounding capabilities, expanding the framework's potential applications beyond document processing.

References

???? Bounding box detection | Generative AI on Vertex AI.

???? Document Data Extraction in 2025: LLMs vs OCRs.

???? Image understanding - Amazon Nova.

???? LLMs vs OCR APIs for Document Processing: The Hidden Cost Trap.

???? Pricing.

???? Vision.

2025. Benchmarking document information localization with Amazon Nova | Artificial Intelligence.

Ariki, Y.; and Motegi, Y. 1995. Segmentation and recognition of handwritten characters using subspace method. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, volume 1, 120–123 vol.1.

Bhattacharyya, A.; Tripathi, A.; Das, U.; Karmakar, A.; Pathak, A.; and Gupta, M. 2025. Information Extraction from Visually Rich Documents using LLM-based Organization of Documents into Independent Textual Segments.

Blog, A. C. ???? DocPath: A fine-tuned large language model for information extraction from documents.

Boros, E.; Nguyen, N. K.; Lejeune, G.; and Doucet, A. 2022. Assessing the impact of OCR noise on multilingual event detection over digitised documents. *International Journal on Digital Libraries*, 23.

Cheng, J.; Liu, X.; Zheng, K.; Ke, P.; Wang, H.; Dong, Y.; Tang, J.; and Huang, M. 2024. Black-Box Prompt Optimization: Aligning Large Language Models without Model Training. arXiv:2311.04155.

Christiano, P.; Leike, J.; Brown, T. B.; Martic, M.; Legg, S.; and Amodei, D. 2023. Deep reinforcement learning from human preferences. arXiv:1706.03741.

Eskenazi, S.; Gomez-Krämer, P.; and Ogier, J.-M. 2016. A comprehensive survey of mostly textual document segmentation algorithms since 2008. *Pattern Recognition*, 64.

Guo, Q.; Wang, R.; Guo, J.; Li, B.; Song, K.; Tan, X.; Liu, G.; Bian, J.; and Yang, Y. 2025. EvoPrompt: Connecting LLMs with Evolutionary Algorithms Yields Powerful Prompt Optimizers. arXiv:2309.08532.

Gutowska, A. 2025. Human in the loop tutorial.

Kanerva, J.; Ledins, C.; Käpyaho, S.; and Ginter, F. 2025. OCR Error Post-Correction with LLMs in Historical Documents: No Free Lunches. arXiv:2502.01205.

Lee, J. H.; and Shin, J. 2024. How to optimize prompting for large language models in clinical research. *Korean J. Radiol.*, 25(10): 869–873.

Limam, M.; Dhiaf, M.; and Kessentini, Y. 2023. FATURA: A Multi-Layout Invoice Image Dataset for Document Analysis and Understanding. arXiv:2311.11856.

Liu, Y.; Yang, B.; Liu, Q.; Li, Z.; Ma, Z.; Zhang, S.; and Bai, X. 2024. TextMonkey: An OCR-Free Large Multimodal Model for Understanding Document. arXiv:2403.04473.

Loukil, F.; Cadereau, S.; Verjus, H.; Galfre, M.; Salamatian, K.; Telisson, D.; Kembellec, Q.; and Le van, O. 2024. LLM-centric pipeline for information extraction from invoices.

Lu, J.; Yu, H.; Wang, Y.; Ye, Y.; Tang, J.; Yang, Z.; Wu, B.; Liu, Q.; Feng, H.; Wang, H.; Liu, H.; and Huang, C. 2025. A Bounding Box is Worth One Token: Interleaving Layout and Text in a Large Language Model for Document Understanding. arXiv:2407.01976.

Mahmud, S.; Nakamura, M.; Wray, K. H.; and Zilberstein, S. 2025. Inference-Aware Prompt Optimization for Aligning Black-Box Large Language Models. arXiv:2508.10030.

Mechi, O.; Mehri, M.; Ingold, R.; and ESSOUKRI BEN AMARA, N. 2019. Text Line Segmentation in Historical Document Images Using an Adaptive U-Net Architecture. 369–374.

Mohammadshirazi, A.; Firoozsalari, A. N.; Zhou, M.; Kulshrestha, D.; and Ramnath, R. 2024. DocParseNet: Advanced Semantic Segmentation and OCR Embeddings for Efficient Scanned Document Annotation. arXiv:2406.17591.

Park, S.; Shin, S.; Lee, B.; Lee, J.; Surh, J.; Seo, M.; and Lee, H. 2019. {CORD}: A Consolidated Receipt Dataset for Post-{OCR} Parsing. In *Workshop on Document Intelligence at NeurIPS 2019*.

Plaksyvyi, A.; Skublewska-Paszkowska, M.; and Powroźnik, P. 2023. A Comparative Analysis of Image Segmentation Using Classical and Deep Learning Approach. *Advances in Science and Technology Research Journal*, 17: 127–139.

Polak, M. P.; and Morgan, D. 2024. Extracting accurate materials data from research papers with conversational language models and prompt engineering. *Nature Communications*, 15(1).

Pryzant, R.; Iter, D.; Li, J.; Lee, Y.; Zhu, C.; and Zeng, M. 2023. Automatic Prompt Optimization with "Gradient Descent" and Beam Search. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 7957–7968. Singapore: Association for Computational Linguistics.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2018. Semantically Equivalent Adversarial Rules for Debugging NLP models. In Gurevych, I.; and Miyao, Y., eds., *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 856–865. Melbourne, Australia: Association for Computational Linguistics.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. arXiv:1505.04597.

Sahoo, P.; Singh, A. K.; Saha, S.; Jain, V.; Mondal, S.; and Chadha, A. 2025. A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications. arXiv:2402.07927.

Settles, B. 2010. Active Learning Literature Survey. *University of Wisconsin, Madison*, 52.

Shah, C. 2024. From Prompt Engineering to Prompt Science With Human in the Loop. arXiv:2401.04122.

Sinha, R.; and S, R. B. 2025. Digitization of Document and Information Extraction using OCR. arXiv:2506.11156.

Sivasubramanian, A.; Mohan, J.; and Sowmya, V. 2024. CASe_UNet: Multi-level Multi-scale UNet for Medical Image Segmentation. In Annappa, B.; Hong, W.-C.; Haritha, D.; and Devi, G. L., eds., *High Performance Computing, Smart Devices and Networks*, 257–270. Singapore: Springer Nature Singapore. ISBN 978-981-97-7794-5.

Soujanya, P.; Koppula, V.; Gaddam, K.; and Sruthi, P. 2013. Comparative Study of Text Line Segmentation Algorithms on Low Quality Documents. *Special Issue of International Journal of Computer Science & Informatics (IJCSI)*, 2231–5292.

Suzuki, S.; and be, K. 1985. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, 30(1): 32–46.

Wang, D.; Raman, N.; Sibue, M.; Ma, Z.; Babkin, P.; Kaur, S.; Pei, Y.; Nourbakhsh, A.; and Liu, X. 2023. DocLLM: A layout-aware generative language model for multimodal document understanding. arXiv:2401.00908.

Xiao, H.; Wang, P.; Yu, M.; and Robbiani, M. 2025. LLM A*: Human in the Loop Large Language Models Enabled A* Search for Robotics. arXiv:2312.01797.

Xu, Y.; Li, M.; Cui, L.; Huang, S.; Wei, F.; and Zhou, M. 2020. LayoutLM: Pre-training of Text and Layout for Document Image Understanding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, 1192–1200. ACM.

Yang, C.; Wang, X.; Lu, Y.; Liu, H.; Le, Q. V.; Zhou, D.; and Chen, X. 2024. Large Language Models as Optimizers. arXiv:2309.03409.

Zhang, H.; Sediq, A. B.; Afana, A.; and Erol-Kantarci, M. 2024. Generative AI-in-the-loop: Integrating LLMs and GPTs into the Next Generation Networks. arXiv:2406.04276.

Zhang, Q.; Wang, B.; Huang, V. S.-J.; Zhang, J.; Wang, Z.; Liang, H.; He, C.; and Zhang, W. 2025. Document Parsing Unveiled: Techniques, Challenges, and Prospects for Structured Information Extraction. arXiv:2410.21169.

Zhao, Y.; and Gao, H. 2024. Utilizing Large Language Models for Information Extraction from Real Estate Transactions. arXiv:2404.18043.

Zhou, Y.; Muresanu, A. I.; Han, Z.; Paster, K.; Pitis, S.; Chan, H.; and Ba, J. 2023. Large Language Models Are Human-Level Prompt Engineers. arXiv:2211.01910.

6 Supplementary

6.1 U-Net Architecture

The U-Net architecture leverages a convolutional neural network (CNN) framework, characterized by its encoder-decoder structure augmented with skip connections. This architectural design facilitates the simultaneous capture of fine-grained details and broader contextual information, making it particularly suitable for precise pixel-level classification tasks. Figure 5 demonstrates the U-Net architecture's application to invoice image processing, illustrating the transformation from input image to segmentation mask output.

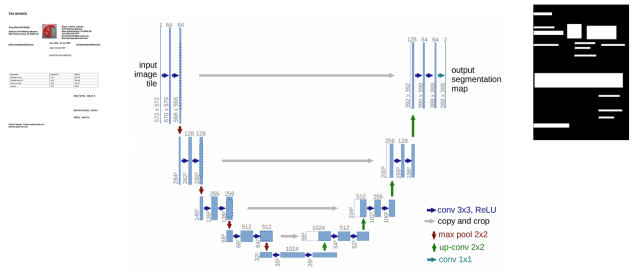

Figure 5: U-Net architecture used for document segmentation. The diagram shows the encoder-decoder structure with skip connections, including layer specifications and intermediate feature map dimensions. Input and output examples demonstrate the transformation from document image to segmentation mask.

6.2 Training Dataset Characteristics

The FATURA dataset comprises 10,000 synthetic invoice document images, representing a comprehensive multi-layout collection. Each image is accompanied by detailed annotations including bounding box coordinates, textual content, and object class designations for key data fields.

The dataset encompasses 50 distinct templates and incorporates 24 unique object detection classes, corresponding to standard invoice components such as total amount, vendor information, and date fields.

The CORD dataset comprises 11,000 real receipts across various Indonesian shops and restaurants, designed for post-OCR parsing. It consists of hierarchical class labels divided into 5 superclasses and 42 subclasses. Each image in the dataset is accompanied by ground truths consisting of category-wise word-level bounding box coordinates.

The preparation of training data involved generating segmentation masks from annotated bounding boxes for each documented field. Figure 6, 7 presents representative examples of source images, their corresponding annotations, and the derived segmentation masks. These image-mask pairs constitute the primary training data for the U-Net model, while the original annotation bounding boxes serve as ground truth references during performance evaluation.

6.3 Experimentation: Accuracy Formulation

The evaluation methodology employs Intersection over Union (IoU) as the primary metric for quantifying detection accuracy. IoU measures the overlap between predicted and ground truth bounding boxes, calculated as the ratio of intersection area to union area (Equation 1). The metric ranges from 0 to 1, where 1 indicates perfect overlap.

A prediction is classified as correct based on a predetermined IoU threshold (Equation 3). For individual images, accuracy is computed by determining the total number of predictions meeting the IoU threshold criterion, with the constraint that each ground-truth object corresponds to at most one prediction (Equation 4). The overall dataset accuracy is then calculated by aggregating correct detections and ground-truth objects across all N images (Equation 5).

The presented equations rigorously define the mathematical framework for accuracy assessment, incorporating predicted bounding boxes (b^*), ground truth bounding boxes (b_g), IoU threshold (θ), and number of samples (N). This formulation ensures a comprehensive and standardized evaluation of the model's performance across varied document layouts and field types.

$$\text{Area}(b^* \cup b_g) = \text{Area}(b^*) + \text{Area}(b_g) - \text{Area}(b^* \cap b_g) \quad (1)$$

$$\text{IoU}(b^*, b_g) = \frac{\text{Area}(b^* \cap b_g)}{\text{Area}(b^* \cup b_g)} \quad (2)$$

$$I(b^*, b_g, \theta) = \begin{cases} 1 & \text{if IoU}(b^*, b_g) \geq \theta \\ 0 & \text{if IoU}(b^*, b_g) < \theta \end{cases} \quad (3)$$

$$\text{CorrectDetections} = \sum_{b^*} \max_{b_g} I(b^*, b_g, \theta) \quad (4)$$

$$\text{Accuracy}_\theta = \frac{\sum_{i=1}^{N} \text{CorrectDetections}^{(i)}}{\sum_{i=1}^{N} N_g^{(i)}} \quad (5)$$

Where

- b^* represents the predicted bounding box
- b_g represents the ground truth bounding box
- θ represents the threshold for Intersection over Union (IoU)
- N represents the total number of images in the dataset
- $N_g^{(i)}$ represents the number of ground truth objects in image i
- $\text{CorrectDetections}^{(i)}$ represents the number of correct detections in image i
- $\text{Area}(b^* \cup b_g)$ represents the union area of predicted and ground truth boxes
- $\text{Area}(b^* \cap b_g)$ represents the intersection area of predicted and ground truth boxes
- $I(b^*, b_g, \theta)$ is an indicator function that returns 1 if IoU exceeds threshold θ, 0 otherwise

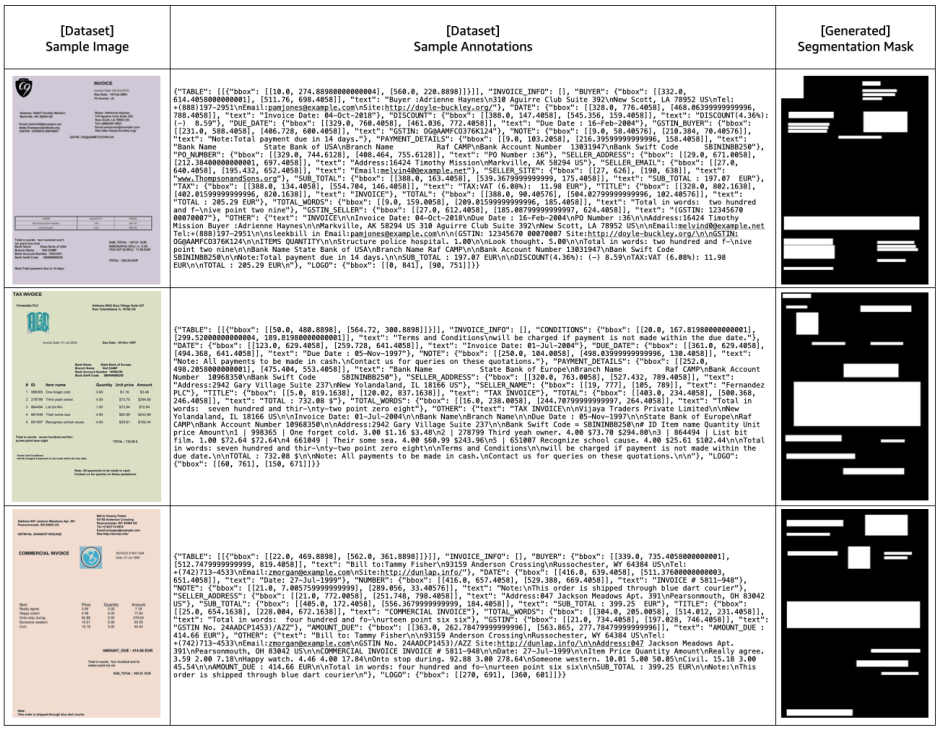

Figure 6: Representative samples from the FATURA dataset showing: (a) original invoice images, (b) corresponding ground truth annotations with bounding boxes, and (c) generated segmentation masks used for U-Net training.

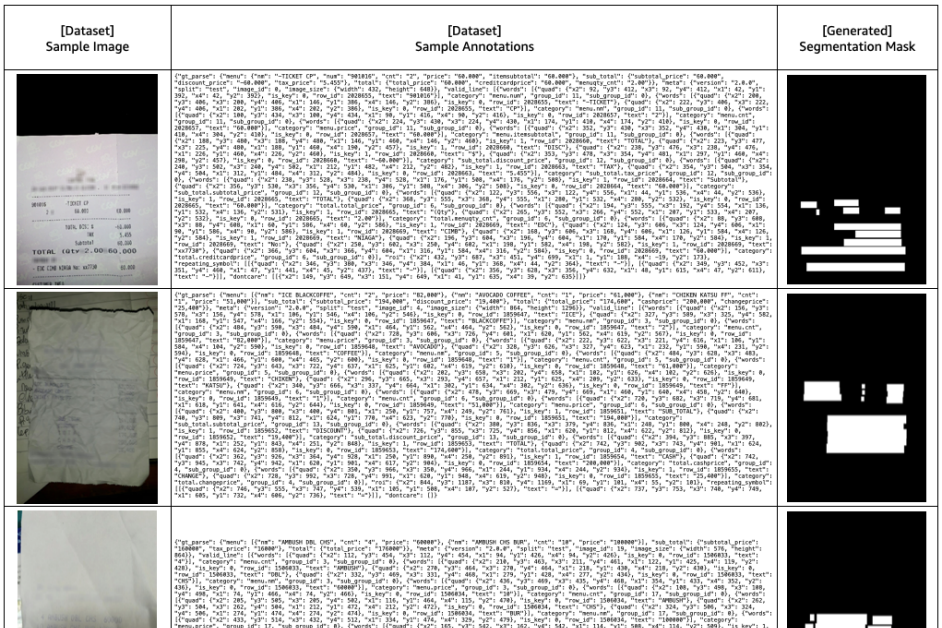

Figure 7: Representative samples from the CORD dataset showing: (a) original invoice images, (b) corresponding ground truth annotations with bounding boxes, and (c) generated segmentation masks used for U-Net training.