GLA-MLLM: A Flowchart Understanding Method Integrating Global-Local Awareness With Multimodal Large Language Model Generation

Xinyue Liang[1], Mengyu Han[1], Hang Xiao[2]*, and Wenhao Zhu[1]
[1]School of Computer Engineering and Science, Shanghai University, Shanghai, China
[2]Shanghai Minhang Polytechnic, Shanghai, China
*Corresponding author: xiaohang@shmp.edu.cn

***Abstract*—Existing Multimodal Large Language Models (MLLMs) demonstrate excellent performance in handling visual tasks in everyday scenarios, but they struggle to understand structured images such as flowcharts and organizational charts. The complex non-linear structure and rich textual content of these images often cause structural hallucinations, resulting in logically inconsistent interpretations. In this paper, we propose GLA-MLLM, a novel flowchart understanding framework integrating global-local awareness with MLLM generation. GLA-MLLM operates in three stages: local structural perception, global semantic understanding, and MLLM-based generation. First, visual primitive detection, text recognition, and a geometry-aware Graph Neural Network (GNN) are integrated to transform raw flowchart images into high-fidelity semantic triplets. Subsequently, an open-source Vision-Language Model (VLM) is leveraged to extract a global semantic summary, providing high-level guidance and steering the model toward a coherent logical interpretation. Finally, these multi-scale cues are serialized into a structured prompt to fine-tune the multimodal model via Low-Rank Adaptation (LoRA), enabling grounded and topologically faithful logical reasoning. We evaluated GLA-MLLM on the Computerized Block Diagrams (CBD) dataset and the FC_B handwritten flowchart dataset, where it achieved state-of-the-art (SOTA) performance, demonstrating strong robustness to diagram complexity and improved cross-domain generalization.**

***Index Terms*—flowchart understanding, multimodal large language models, fine-tuning, structured images, structural hallucinations**

I. Introduction

Flowcharts are widely used visual tools for representing complex logical procedures and decision-making processes across diverse domains. In software engineering, they are commonly used for algorithm visualization and automated code generation [1]; in the medical domain, they support standardized clinical diagnosis and treatment guidelines [2]; and in business management, they facilitate the optimization of operational workflows [3]. Consequently, the ability to automatically parse and interpret flowchart logic into natural language is of significant importance.

Traditional approaches to flowchart understanding primarily relied on rigid rule-based systems and basic shape analysis [4]. While effective under specific constraints, these methods often struggled to generalize across diverse layouts and to handle the visual noise present in hand-drawn diagrams [5]. To address these limitations, detection-based approaches such as Arrow R-CNN were proposed to improve topological reconstruction by incorporating dedicated arrow keypoint prediction [6]. More recently, the field has shifted toward generative approaches based on Multimodal Large Language Models (MLLMs). Models such as FlowVQA [7] and mPLUG-PaperOwl [8] have demonstrated strong capabilities in resolving complex logical queries. However, due to the complex and non-linear structure of flowcharts, existing end-to-end MLLMs often struggle to capture fine-grained textual details and tend to produce structural hallucinations.

To overcome these challenges, we propose GLA-MLLM, a novel flowchart understanding framework integrating global-local awareness with MLLM generation, as illustrated in Fig. 1. First, the Local Structural Perception Module (LSPM) integrates visual primitive detection, text recognition, and a geometry-aware Graph Neural Network (GNN) to transform raw flowchart elements into high-fidelity semantic triplets; this process explicitly maps localized connectivity into a structured format that the MLLM can interpret. We then introduce the Global Semantic Understanding Module (GSUM) to extract a global semantic summary of the flowchart's macro-logic, which serves as a top-down semantic prior to align the generative process with the diagram's overarching intent. Finally, these multi-scale cues are serialized into a hierarchical structured prompt to fine-tune the MLLM via Low-Rank Adaptation (LoRA). This approach creates a controllable reasoning environment that ensures strict topological faithfulness while maintaining the linguistic fluency required for complex logical derivation. Experimental results demonstrate that GLA-MLLM exhibits excellent performance in understanding structured images. In summary, our contributions are as follows:

(1) We propose GLA-MLLM, a novel framework that integrates global–local-aware perception with MLLM for robust flowchart understanding. (2) We introduce a structured reasoning pipeline that explicitly models both fine-grained structural cues and global semantic context, effectively mitigating structural hallucinations in complex and non-linear flowcharts. (3) Extensive experiments on both computerized and handwritten flowchart datasets demonstrate that our approach achieves state-of-the-art performance on flowchart understanding tasks.

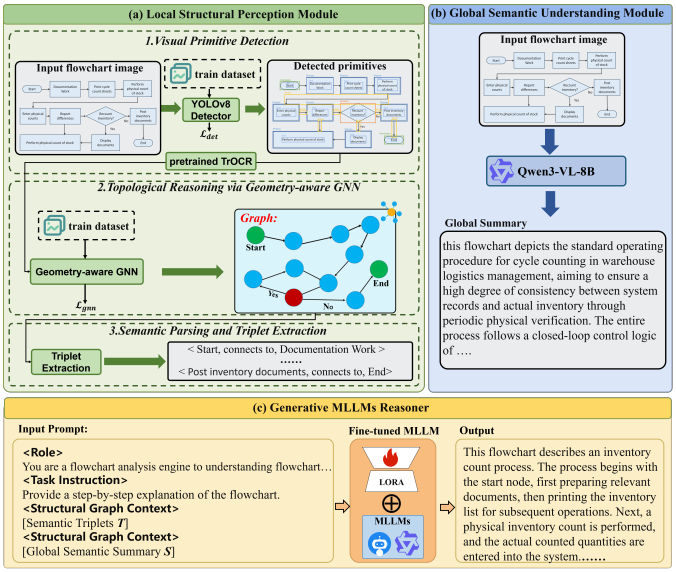

Fig. 1. Overall framework of GLA-MLLM, consisting of three main modules: (a) a Local Structural Perception Module (LSPM) to detect visual primitives and extract structural triplets; (b) a Global Semantic Understanding Module (GSUM) to generate macroscopic summaries; and (c) a Generative MLLM Reasoner that integrates structural and global information to perform enhanced flowchart logical reasoning.

II. Related Work

A. Multimodal Large Language Models

Large Language Models (LLMs) have demonstrated robust capabilities in language comprehension, reasoning, and generalization [9]. Building upon this, MLLMs extend these reasoning skills to additional modalities such as image [10], and video [11]. Typically, MLLMs align target features with textual features and then integrate them with LLMs for various text inference tasks. Some train the whole architecture from scratch [12] and others [13] utilize pretrained LLMs.

Currently, leveraging MLLMs for flowchart understanding has become an active research area. For example, FlowVQA [7] establish benchmarks for evaluating multi-directional logical reasoning and multi-dimensional comprehension in flowcharts. In addition, mPLUG-PaperOwl [8] improves diagram analysis by aligning scientific figures with textual context. However, these end-to-end approaches often struggle with structural hallucinations and the loss of fine-grained textual details when processing dense flowcharts. To address these challenges, we propose GLA-MLLM, a dual-path framework that explicitly injects local topological triplets and global semantic summaries into MLLMs to enable grounded and hallucination-resilient flowchart understanding.

B. Flowchart Understanding

Flowcharts have been widely used across multiple domains, including software engineering [1], healthcare [2], and business [3]. Therefore, many approaches have been proposed for their automatic understanding and parsing. Traditional methods typically relied on predefined symbol templates, image processing techniques, and rigid structural grammars [4]. Early researchers such as [5] utilized primitive-based structural analysis to decompose diagrams into basic geometric shapes and lines. While these rule-based approaches ensured precision within specific constraints, they suffered from limited adaptability to diverse layouts, varied line styles, and hand-drawn noise. To overcome this rigidity, detection-based AI approaches were introduced. Bernhard et al. Arrow R-CNN [6] achieves more accurate topological reconstruction, yet remains highly rigid when handling irregular hand-drawn strokes, diverse line styles, and cluttered layouts.

More recently, research has shifted toward generative approaches to enable more natural and semantically rich flowchart interpretations. Raghu et al. [14] proposed an end-to-end framework that parses flowcharts through interactive dialogues with GPT-2, leveraging a retrieval-augmented generation (RAG) architecture. [15] further introduced GenFlowchart, a generative AI framework that interprets flowchart logic by mapping visual components to structured semantic descriptions. Singh et al. [7] advanced this line of work by proposing FlowVQA, which leverages large-scale multimodal models to answer complex logical queries within flowchart structures. Despite these advances, existing generative approaches often struggle to maintain strict logical consistency across non-linear execution paths, primarily due to the absence of explicit structural constraints and a lack of awareness of global, macro-level intent. As a result, such models are prone to structural hallucinations and inconsistent reasoning when processing complex flowcharts. To address these challenges, we propose a geometry-aware framework. It combines precise topological refinement with global semantic guidance, anchoring the generative logic in grounded structural evidence to effectively mitigate hallucinations in complex flowchart parsing.

III. Proposed Method

A. Local Structural Perception Module

The Local Structural Perception Module (LSPM) aims to convert raw flowchart images into structured, fine-grained cues for downstream reasoning. It first detects visual primitives and recognizes associated text, and then binds the recognized content to the detected elements to support subsequent topological parsing and triplet construction.

1) Visual Primitive Detection and Text Recognition: We formulate primitive extraction as a multi-class object detection task using YOLOv8. The detector localizes six functional components: *terminator*, *process*, *decision*, *data*, *text*, and *arrow*.

After localization, textual content within *text* and functional boxes is extracted using a pretrained TrOCR model, which employs an encoder–decoder architecture to recognize irregular handwritten and printed text. To mitigate the visual noise prevalent in the FC_B dataset, detected text patches undergo adaptive contrast enhancement and deskewing before recognition. The recognized strings are then associated with

their corresponding nodes or arrows via spatial containment and proximity rules, forming the semantic basis for triplet extraction. This binding provides geometric and linguistic constraints for initial connectivity inference, enabling the model to distinguish the flow's source from its destination.

2) Topological Reasoning via Geometry-aware GNN: While visual detection yields a set of discrete primitives, it does not directly provide a reliable logical topology. To obtain a more robust reconstruction of the flowchart structure, we introduce a geometry-aware GNN to refine connectivity. Each detected element d_i is mapped to a graph node v_i characterized by a hybrid node feature vector $\mathbf{x}_i$:

$$\mathbf{x}_i = [\mathrm{norm}(bbox_i) \parallel \text{one-hot}(cls_i)], \tag{1}$$

where $\mathrm{norm}(bbox_i) = [x_c, y_c, w, h]$ denotes the normalized spatial attributes of the i-th element. Here (x_c, y_c) and (w, h) are respectively the center coordinates, width, and height, scaled relative to the image dimensions (W, H) for resolution invariance. Meanwhile, $\text{one-hot}(cls_i) \in \{0,1\}^6$ is a one-hot vector encoding the functional category of the element, such as a decision, process, or arrow. This term provides explicit semantic identity to the node's feature space, allowing the GNN to distinguish among different primitive types during topological reasoning.

For any node pair (i, j), we compute a geometric feature vector consisting of relative displacement $(\Delta x_{ij}, \Delta y_{ij})$, Euclidean distance $dist_{ij}$, and azimuth angle θ_{ij}. A non-linear bias layer ψ transforms these features into a prior matrix:

$$\mathbf{B}_{ij} = \psi(\Delta x_{ij}, \Delta y_{ij}, dist_{ij}, \theta_{ij}). \tag{2}$$

Node-wise hidden representations $\mathbf{h}_i$ are obtained from the GNN encoder, and pairwise features are constructed as:

$$\mathbf{H}_{ij} = [\mathbf{h}_i \parallel \mathbf{h}_j]\,. \tag{3}$$

The final adjacency probability $\mathbf{A}_{ij}$ is determined by fusing these pairwise node representations with the geometric bias:

$$\mathbf{A}_{ij} = \sigma(\mathrm{MLP}(\mathbf{H}_{ij}) + \mathbf{B}_{ij}), \tag{4}$$

where σ denotes the sigmoid activation function. This design down-weights geometrically implausible edges and enhances the robustness of long-range connection predictions. The resulting matrix $\mathbf{A}$ is interpreted as a probabilistic adjacency matrix that encodes the refined flowchart topology for subsequent semantic parsing.

3) Semantic Parsing and Triplet Extraction: Finally, LSPM derives a symbolic knowledge base from the graph. Text within shape boundaries is bound to the corresponding node and used as its semantic label, while text located in the vicinity of arrows is interpreted as a relational descriptor (e.g., "Yes/No" branches). Specifically, given the refined adjacency matrix $\mathbf{A}$ and a set of localized text segments $\mathcal{S}$, we first define a mapping function $L(v_i)$ that assigns text to node v_i based on spatial containment, and let e_{ij} denote the directed edge from v_i to v_j. The module then constructs a set of semantic triplets $\mathcal{T}$ as follows:

$$\mathcal{T} = \{\langle L(v_i), R(e_{ij}), L(v_j)\rangle \mid \mathbf{A}_{ij} > \tau\}, \tag{5}$$

where τ is a confidence threshold, and $R(e_{ij})$ is the relation extraction function defined as:

$$R(e_{ij}) = \begin{cases} s_k, \text{ if } \exists s_k \in \mathcal{S} \text{ s.t. } \mathrm{dist}(s_k, e_{ij}) < \delta, \\ \text{connected with, otherwise.} \end{cases} \tag{6}$$

where $\mathrm{dist}(s_k, e_{ij})$ measures the spatial distance between text segment s_k and the arrow e_{ij}, and δ represents the proximity threshold. The module ultimately outputs a set of semantic triplets $(\langle\text{Head}\rangle, \langle\text{Relation}\rangle, \langle\text{Tail}\rangle)$, forming a structured foundation for downstream logical reasoning.

4) Training Objectives and Loss Functions: LSPM is trained in two stages: a detection loss for visual primitive extraction and a graph-based loss for topological refinement. The overall optimization target is a weighted sum:

$$\mathcal{L} = \lambda_{\text{det}}\mathcal{L}_{\text{det}} + \lambda_{\text{gnn}}\mathcal{L}_{\text{gnn}}, \tag{7}$$

where λ_{det} and λ_{gnn} balance the contributions of the two stages.

a) YOLOv8 Loss for Visual Primitive Detection: We adopt the standard YOLOv8 multi-task loss, which jointly supervises box regression and category classification. Let $\mathcal{D}$ denote the set of matched predictions and ground-truth targets produced by the label assignment strategy in YOLOv8. The detection loss is defined as:

$$\mathcal{L}_{\text{det}} = \mathcal{L}_{\text{box}} + \lambda_{\text{cls}}\mathcal{L}_{\text{cls}}. \tag{8}$$

For box regression, YOLOv8 employs an IoU-based loss with distribution-aware regression. For clarity of presentation, we abstract it as a unified IoU loss, which effectively handles scale and aspect-ratio variations during training:

$$\mathcal{L}_{\text{box}} = \frac{1}{|\mathcal{D}|}\sum_{(p,g)\in\mathcal{D}} \left(1 - \mathrm{IoU_c}(bbox_p, bbox_g)\right), \tag{9}$$

where $\mathrm{IoU_c}(\cdot)$ denotes a complete IoU variant and (p, g) represents a matched prediction–ground-truth pair.

The classification term is implemented as a binary cross-entropy loss over the 6 functional categories:

$$\mathcal{L}_{\text{cls}} = \frac{1}{|\mathcal{D}|}\sum_{(p,g)\in\mathcal{D}} \mathrm{BCE}\left(\hat{\mathbf{y}}_p, \mathbf{y}_g\right), \tag{10}$$

where $\hat{\mathbf{y}}_p \in [0,1]^6$ is the predicted class probability vector and $\mathbf{y}_g \in \{0,1\}^6$ is the one-hot ground-truth label.

b) GNN Loss for Topological Reasoning: The geometry-aware GNN predicts a probabilistic adjacency matrix $\mathbf{A} \in [0,1]^{N\times N}$, where $\mathbf{A}_{ij}$ indicates the likelihood of a directed connection from v_i to v_j. Given the ground-truth adjacency matrix $\mathbf{G} \in \{0,1\}^{N\times N}$ constructed from annotated flow

edges, we optimize the GNN via a weighted binary cross-entropy to address the strong class imbalance between positive and negative edges:

$$\mathcal{L}_{\text{bce}} = -\sum_{i \neq j} \left(\alpha \, \mathbf{G}_{ij} \log \mathbf{A}_{ij} + (1 - \mathbf{G}_{ij}) \log(1 - \mathbf{A}_{ij}) \right), \tag{11}$$

where $\alpha > 1$ up-weights positive edges.

In addition, to encourage a coherent directed structure, we optionally regularize the adjacency predictions with a sparsity prior:

$$\mathcal{L}_{\text{spr}} = \frac{1}{N^2} \sum_{i \neq j} \mathbf{A}_{ij}. \tag{12}$$

The final GNN objective is:

$$\mathcal{L}_{\text{gnn}} = \mathcal{L}_{\text{bce}} + \lambda_{\text{spr}} \mathcal{L}_{\text{spr}}, \tag{13}$$

where $\mathcal{L}_{\text{bce}}$ denotes the weighted binary cross-entropy term in Eq. (11). Through the combined optimization, the detector learns robust primitive localization, while the GNN learns geometry-consistent edge inference, resulting in a reliable structural graph for subsequent semantic triplet extraction.

B. Global Semantic Understanding Module

In the task of understanding structured flowcharts and block diagrams, purely bottom-up detection methods focus on fine-grained visual elements but lack the capacity to capture the macro intent and overall control logic. To mitigate this, we introduce the Global Semantic Understanding Module (GSUM), which leverages a Qwen3-VL-8B as a global information extractor. To effectively exploit the VLM, GSUM employs a task-oriented prompt $\mathcal{P}$ that explicitly instructs the model to adopt a global perspective. The raw image is presented with the following concise instruction:

> *From a **global perspective**, provide a concise, single-paragraph **global semantic summary** of the given flowchart. Your analysis must first identify the **application domain** and **overall processing goal**, then synthesize the **macro control-flow pattern**, including start/end conditions, primary processing stages, and critical decision logic, to capture the diagram's overarching intent without getting lost in local details.*

The primary output of GSUM is a global semantic summary denoted as $\mathcal{S}$. This textual descriptor captures high-level semantics in a joint visual–textual feature space, representing the macro-logic of the diagram.

C. Generative MLLM Reasoner

We propose an MLLM-based generative reasoner that utilizes semantic triplets and the global summary to perform flowchart understanding. Specifically, we serialize the semantic triplets $\mathcal{T}$ and the global semantic summary $\mathcal{S}$ into an structured-evidence prompt and apply parameter-efficient fine-tuning (PEFT) to adapt the MLLM to this domain-specific task. At inference time, the model generates a logically coherent interpretation of the flowchart, conditioned on the prompt context.

1) Prompt Design: Recent studies emphasize that the performance of MLLMs are highly sensitive to prompt formulations, especially when interpreting complex structural data. For flowchart understanding, MLLMs risk hallucinating logical paths or skipping critical decision nodes if the reasoning is not strictly grounded in the extracted topology. To effectively exploit the synergy between bottom-up extractions and top-down semantics, our prompt design adheres to three guiding principles: hierarchical evidence grounding, attribute-aware serialization, and structural role-based constraints. First, we explicitly inject both the fine-grained semantic triplets $\mathcal{T}$ and the macro-structural summary to anchor the model's reasoning in observed connectivity rather than speculative parametric bias. This graph structure is then serialized into a consistent $(\langle\text{Head}\rangle, \langle\text{Relation}\rangle, \langle\text{Tail}\rangle)$ format and augmented with node-specific attributes, such as node types, ensuring a stable symbolic understanding across diverse layouts. Furthermore, the system prompt establishes the model's role as a structural analyst, imposing strict constraints that encourage Chain-of-Thought (CoT) reasoning while forbidding the fabrication of non-existent edges to mitigate hallucination. Based on these principles, we adopt the following hierarchical prompt template:

> ***Role**: You are a flowchart analysis engine. You must perform reasoning based strictly on the provided graph structure and semantic summary.*
> ***Task Instruction**:* [TASK].
> ***Structural Graph Context**:* [GRAPH_TRIPLETS].
> ***Global Semantic Summary**:* [GLOBAL_SUMMARY].

2) Parameter-Efficient Fine-Tuning and Inference with LLM: To mitigate the limited domain awareness and hallucination tendencies of general-purpose MLLMs in flowchart reasoning, we employ PEFT to adapt the model to structured symbolic evidence. Specifically, we adopt LoRA by freezing the pretrained backbone parameters Θ and injecting trainable low-rank matrices Φ into the attention and cross-modal projection layers. The model is optimized using a standard cross-entropy loss over target tokens:

$$\mathcal{L} = -\sum_{t=1}^{T} \log P(\bar{y}_t \mid \mathcal{I}, \mathcal{P}, \bar{y}_{<t}; \Theta, \Phi), \tag{14}$$

where $\mathcal{I}$ is the input image, $\mathcal{P}$ is the structured prompt, and $\bar{y}_{<t} = \{\bar{y}_1, \ldots, \bar{y}_{t-1}\}$ denotes the ground-truth prefix. At inference time, we use the same prompt format and provide the semantic triplets and the global semantic summary as contextual evidence for flowchart understanding.

TABLE I
HYPERPARAMETER SETTINGS FOR FINE-TUNING AND INFERENCE.

Stage	Parameters
Fine-tuning	lora_rank: 64, lora_alpha: 128, lora_dropout: 0.05 cutoff_len: 2048, learning_rate: 1.0e-4 num_train_epochs: 10.0, warmup_ratio: 0.1, bf16: true
Inference	temperature: 0.9, top_p: 0.9

IV. EXPERIMENTS

We first describe the experimental settings in Section IV-A, followed by a presentation of the main results in Section IV-B, then the ablation study in Sections IV-C.

A. Experimental Settings

Datasets. To evaluate the performance of our proposed flowchart understanding framework, we utilize two public datasets: the Computerized Block Diagrams (CBD) dataset [16] and a handwritten flowchart dataset (FC_B) [16]. The CBD dataset is a benchmark designed for complex computerized flowcharts and contains 502 images. The FC_B dataset is a publicly available collection of online handwritten flowcharts, commonly used to evaluate the generalization capability of flowchart understanding models.

Evaluation Metrics. The flowchart understanding task in this paper is formulated as an image-to-text generation problem, where the model is required to produce a natural language description given a flowchart image. To comprehensively evaluate the quality of the generated descriptions, we adopt four widely used metrics that assess complementary aspects of generation performance, including BLEU [17], ROUGE-1/L [18] and Perplexity (PPL) [19].

Implementation Details. We utilized Qwen3-VL-8B as the primary model for our experimental framework. We use the LlamaFactory framework to fine-tune LLM based on LoRA and fine-tuning parameters and inference parameter settings at each stage can be found in the Table I. For LSPM, a YOLOv8-based primitive detector is trained for 100 epochs on the merged CBD and FC_B datasets to recognize functional components. Subsequently, a geometry-aware GNN composed of two GAT layers with a 256-dimensional hidden space is pre-trained for 50 epochs on the same merged dataset using the AdamW optimizer and a BCE loss to refine topological connectivity. Finally, the refined structural cues are parsed into semantic triplets using a confidence threshold $\tau = 0.5$ and a proximity threshold $\delta = 0.05$, providing a grounded logical foundation for the generative reasoner.

Baseline Models. To comprehensively evaluate the effectiveness of our approach, we compare against a broad range of baseline models covering four representative paradigms of flowchart understanding. First, we assess general vision language understanding using mPLUG [20] and the OCR-free Donut [21]. Second, we include multi-stage structural parsing pipelines that combine visual detectors (Faster R-CNN or BloSum) with sequence-to-sequence generators (BART or T5). Third, we evaluate recent large-scale MLLMs, including GPT-4o, Claude 3.5 Sonnet, and Gemini 2.5 Flash, as strong black-box baselines under both zero-shot and few-shot settings. Finally, we adopt BlockNet [22] as the primary task-specific baseline, which represents the current state-of-the-art for flowchart understanding.

TABLE II
PERFORMANCE COMPARISON ON THE CBD DATASET. **BOLD** INDICATES BEST, UNDERLINE INDICATES SECOND BEST.

Models	BLEU ↑	R-1 ↑	R-L ↑	PPL ↓
mPLUG	3.74	15.18	12.45	5.46
Faster R-CNN + BART_L	17.29	31.16	28.50	17.93
Faster R-CNN + T5_L	22.11	40.10	36.80	12.06
BloSum + BART_L	33.47	75.24	68.45	8.33
BloSum + T5_L	42.18	80.78	74.20	7.55
Donut	54.73	75.50	71.30	38.35
GPT-4o	69.50	85.33	79.50	5.95
Claude 3.5 Sonnet	72.50	88.40	82.15	5.82
Gemini 2.5 Flash	71.15	87.25	81.30	6.01
BlockNet	72.31	88.90	82.40	6.87
GLA-MLLM	**74.63**	**92.96**	**89.23**	**5.15**

TABLE III
PERFORMANCE COMPARISON ON THE FC_B DATASET. **BOLD** INDICATES BEST, UNDERLINE INDICATES SECOND BEST.

Models	BLEU ↑	R-1 ↑	R-L ↑	PPL ↓
mPLUG	3.51	14.81	12.10	4.68
Faster R-CNN + BART_L	22.19	41.63	37.40	13.58
Faster R-CNN + T5_L	27.81	50.47	44.90	10.91
BloSum + BART_L	15.49	35.39	31.20	14.10
BloSum + T5_L	20.04	40.85	36.15	13.46
Donut	69.29	73.37	69.10	26.89
GPT-4o	62.17	89.06	82.40	4.79
Claude 3.5 Sonnet	65.40	90.15	83.20	4.75
Gemini 2.5 Flash	64.20	89.70	82.85	4.92
BlockNet	67.74	90.63	83.15	5.13
GLA-MLLM	**70.77**	**93.29**	**83.69**	**4.42**

B. Main Results

The performance of the proposed GLA-MLLM framework on the CBD and FC_B datasets is summarized in Table II and Table III, establishing a new state-of-the-art across all metrics by consistently outperforming traditional pipelines, document-level models, and general-purpose MLLMs. Specifically, on the CBD dataset, GLA-MLLM achieves a BLEU score of 74.63 and a ROUGE-L of 89.23, surpassing the previous task-specific leader, BlockNet, due to the LSPM's ability to refine detections into a probabilistic adjacency matrix via a geometry-aware GNN. This structural grounding yields exceptional cross-domain robustness; on the handwritten FC_B benchmark, GLA-MLLM attains a ROUGE-1 score of 93.29 and the lowest PPL, maintaining high performance where rigid OCR-dependent models like BloSum + T5_L degrade.

Furthermore, by integrating GSUM's macro-logic summaries with LSPM's bottom-up semantic triplets, GLA-MLLM surpasses general-purpose models like GPT-4o and OCR-free models like Donut. A key observation from our comparative analysis is the necessity of fine-tuning over pure prompting. While general-purpose MLLMs often rely on zero-shot or few-shot prompting to interpret diagrams, such methods frequently struggle to eliminate structural hallucinations, especially in dense or non-linear layouts where linguistic biases can lead the model to infer incorrect paths. In contrast, our LoRA-based fine-tuning explicitly forces the MLLM to align its generative reasoning with the provided topological

signals. This supervised adaptation ensures superior logical faithfulness and linguistic quality, yielding the lowest PPL scores across all benchmarks while remaining closely anchored to the underlying flowchart topology.

TABLE IV
ABLATION RESULTS ON THE CBD AND FC_B DATASETS. W/O LSPM, W/O GSUM, AND W/O FT DENOTE REMOVING THE CORRESPONDING COMPONENTS.

Variant	CBD			FC_B		
	BLEU ↑	R-1 ↑	PPL ↓	BLEU ↑	R-1 ↑	PPL ↓
GLA-MLLM (Full)	**74.63**	**92.96**	**5.15**	**70.77**	**93.29**	**4.42**
w/o LSPM	63.80	86.90	10.20	61.12	85.30	5.80
w/o GSUM	65.45	84.12	9.12	57.30	81.55	8.45
w/o Fine-tuning	42.15	45.30	14.25	48.40	72.15	12.80

C. Ablation Study

To evaluate the contribution of each component within the GLA-MLLM framework, we conduct an ablation study and the results are summarized in Table IV. The removal of LSPM causes a significant performance drop across all metrics, proving that GNN-derived topological triplets provide a critical logical prior that goes beyond simple node-level semantics. Excluding GSUM leads to increased PPL and degraded coherence, highlighting the necessity of macro-intent extraction for guiding the generative reasoner and maintaining a consistent logical flow. Finally, the collapse in zero-shot performance underscores that supervised fine-tuning is indispensable for bridging the substantial domain gap between general-purpose MLLMs and the specialized task of structured flowchart interpretation.

V. CONCLUSION

In this paper, we propose GLA-MLLM, a novel flowchart understanding framework integrating global-local awareness with MLLM Generation. By harmonizing local topological triplets with global semantic summaries, our approach anchors the generative process in grounded structural evidence, effectively mitigating the structural hallucinations common in end-to-end models. Furthermore, the integration of hierarchical structural prompting and fine-tuning allows the MLLM to overcome domain-specific challenges and layout irregularities. Extensive experiments on computerized and handwritten benchmarks demonstrate that GLA-MLLM achieves logically faithful, domain-aware, and linguistically coherent interpretations. These results indicate state-of-the-art performance on flowchart understanding tasks.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China under Grant No. 12471500.

REFERENCES

[1] C. Supaartagorn, "Web application for automatic code generator using a structured flowchart," in *2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS)*. IEEE, 2017, pp. 114–117.

[2] M. Tanaka, C. Fernández-del Castillo, T. Kamisawa, J. Y. Jang, P. Levy, T. Ohtsuka, R. Salvia, Y. Shimizu, M. Tada, and C. L. Wolfgang, "Revisions of international consensus fukuoka guidelines for the management of ipmn of the pancreas," *Pancreatology*, vol. 17, no. 5, pp. 738–753, 2017.

[3] M. Dumas, L. M. Rosa, J. Mendling, and A. H. Reijers, *Fundamentals of business process management*. Springer, 2018.

[4] A. Winkelmann and B. Weiß, "Automatic identification of structural process weaknesses in flow chart diagrams," *Business Process Management Journal*, vol. 17, no. 5, pp. 787–807, 2011.

[5] M. Rusiñol, L.-P. de las Heras, and O. R. Terrades, "Flowchart recognition for non-textual information retrieval in patent search," *Information retrieval*, vol. 17, no. 5, pp. 545–562, 2014.

[6] B. Schäfer, M. Keuper, and H. Stuckenschmidt, "Arrow r-cnn for handwritten diagram recognition," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 24, no. 1, pp. 3–17, 2021.

[7] S. Singh, P. Chaurasia, Y. Varun, P. Pandya, V. Gupta, V. Gupta, and D. Roth, "Flowvqa: Mapping multimodal logic in visual question answering with flowcharts," *arXiv preprint arXiv:2406.19237*, 2024.

[8] A. Hu, Y. Shi, H. Xu, J. Ye, Q. Ye, M. Yan, C. Li, Q. Qian, J. Zhang, and F. Huang, "mplug-paperowl: Scientific diagram analysis with the multimodal large language model," in *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024, pp. 6929–6938.

[9] I. Team, "Internlm: A multilingual language model with progressively enhanced capabilities," 2023.

[10] P. Gao, J. Han, R. Zhang, Z. Lin, S. Geng, A. Zhou, W. Zhang, P. Lu, C. He, X. Yue *et al.*, "Llama-adapter v2: Parameter-efficient visual instruction model," *arXiv preprint arXiv:2304.15010*, 2023.

[11] K. Li, Y. He, Y. Wang, Y. Li, W. Wang, P. Luo, Y. Wang, L. Wang, and Y. Qiao, "Videochat: Chat-centric video understanding," *Science China Information Sciences*, vol. 68, no. 10, p. 200102, 2025.

[12] S. Huang, L. Dong, W. Wang, Y. Hao, S. Singhal, S. Ma, T. Lv, L. Cui, O. K. Mohammed, B. Patra *et al.*, "Language is not all you need: Aligning perception with language models," *Advances in Neural Information Processing Systems*, vol. 36, pp. 72 096–72 109, 2023.

[13] B. Li, Y. Zhang, L. Chen, J. Wang, F. Pu, J. A. Cahyono, J. Yang, C. Li, and Z. Liu, "Otter: A multi-modal model with in-context instruction tuning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[14] D. Raghu, S. Agarwal, S. Joshi *et al.*, "End-to-end learning of flowchart grounded task-oriented dialogs," *arXiv preprint arXiv:2109.07263*, 2021.

[15] A. Arbaz, H. Fan, J. Ding, M. Qiu, and Y. Feng, "Genflowchart: parsing and understanding flowchart using generative ai," in *International Conference on Knowledge Science, Engineering and Management*. Springer, 2024, pp. 99–111.

[16] S. Bhushan and M. Lee, "Block diagram-to-text: Understanding block diagram images by generating natural language descriptors," in *Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022*, 2022, pp. 153–168.

[17] M. Post, "A call for clarity in reporting bleu scores," *arXiv preprint arXiv:1804.08771*, 2018.

[18] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74–81.

[19] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[20] Q. Ye, H. Xu, G. Xu, J. Ye, M. Yan, Y. Zhou, J. Wang, A. Hu, P. Shi, Y. Shi *et al.*, "mplug-owl: Modularization empowers large language models with multimodality," *arXiv preprint arXiv:2304.14178*, 2023.

[21] G. Kim, T. Hong, M. Yim, J. Nam, J. Park, J. Yim, W. Hwang, S. Yun, D. Han, and S. Park, "Ocr-free document understanding transformer," in *European Conference on Computer Vision*. Springer, 2022, pp. 498–517.

[22] S. Bhushan, E.-S. Jung, and M. Lee, "Unveiling the power of integration: Block diagram summarization through local-global fusion," in *Findings of the Association for Computational Linguistics ACL 2024*, 2024, pp. 13 837–13 856.