

Pi-GPS: Enhancing Geometry Problem Solving by Unleashing the Power of Diagrammatic Information

Junbo Zhao^{1*} Ting Zhang^{1,3*†} Jiayu Sun¹ Mi Tian² Hua Huang^{1,3†}

¹Beijing Normal University ²TAL Education Group

³Engineering Research Center of Intelligent Technology and Educational Application (MOE)

Abstract

Geometry problem solving has garnered increasing attention due to its potential applications in intelligent education field. Inspired by the observation that text often introduces ambiguities that diagrams can clarify, this paper presents Pi-GPS, a novel framework that unleashes the power of diagrammatic information to resolve textual ambiguities, an aspect largely overlooked in prior research. Specifically, we design a micro module comprising a rectifier and verifier: the rectifier employs MLLMs to disambiguate text based on the diagrammatic context, while the verifier ensures the rectified output adherence to geometric rules, mitigating model hallucinations. Additionally, we explore the impact of LLMs in theorem predictor based on the disambiguated formal language. Empirical results demonstrate that Pi-GPS surpasses state-of-the-art models, achieving a nearly 10% improvement on Geometry3K over prior neural-symbolic approaches. We hope this work highlights the significance of resolving textual ambiguity in multimodal mathematical reasoning, a crucial factor limiting performance. The code for Pi-GPS is publicly available at: <https://github.com/hellozting/Pi-GPS>.

1. Introduction

Geometry Problem Solving (GPS) aims to derive solutions from a textual problem description and its corresponding diagram. As a distinct and pivotal aspect of multimodal mathematical reasoning, GPS requires a nuanced understanding of visual shapes, intricate spatial relationships, symbolic abstraction, and logical inference across both textual and diagrammatic inputs. This makes it a long-standing challenge in mathematical reasoning and artificial intelligence [8, 14, 21, 28, 35]. While recent notable milestones [3, 33] such as the gold-medal-level solution to geometry problems using AlphaGeometry2 [11] have ex-

Problem Text	Parsed Text Formal Language	Diagram
The rectangle is inscribed into the circle . Find the exact circumference of the circle .	<code>InscribedIn(Rectangle(\$),Circle(\$))</code> <code>Find(CircumferenceOf(Circle(\$)))</code>	
The two polygons are similar. Find UT .	<code>Similar(Polygon(\$1),Polygon(\$2))</code> <code>Find(LengthOf(Line(U,T)))</code>	
Find the area of the shaded region . Round to the nearest tenth.	<code>Find(AreaOf(Shaded(Shape(\$))))</code>	

Figure 1. Illustrating the ambiguity presented in text. Text alone offers insufficient information to resolve the ambiguity, and disambiguation becomes straightforward when supported by a diagram.

hibited remarkable achievements, these efforts predominantly focus on language processing, neglecting the diagrammatic component of the problem. However, GPS transcends language-based reasoning, demanding a profound understanding and manipulation of diagrammatic information, an enduring challenge in the field.

Existing approaches to GPS can be broadly categorized into symbolic [4, 30] and neural-based methods [8, 13, 16–18, 22, 31, 37, 38, 40, 42–45]. Symbolic methods, grounded in formal logic and mathematical rigor, rely on explicit theorem databases and symbolic manipulation to construct logically sound reasoning paths. These methods excel in providing interpretable steps and ensuring formal correctness. However, the predefined rules may struggle to accommodate diverse problem types. In contrast, neural-based approaches leverage data-driven learning to generate solution paths from vast training datasets. These models offer flexibility and scalability, handling problems of varying complexity. However, their reliance on large, high-quality annotated datasets and the lack of rigorous correctness guarantees pose significant limitations. Therefore, many works [21, 25, 36, 46] attempt to combine the procedural power of symbolic models with the general power of neural models. Such hybrid approaches in general involves two key steps: parsing and reasoning. Parsing entails ex-

*Equal contribution.

†Corresponding author.

tracting formal language representations from the diagram and the accompanying text, while reasoning employs these parsed elements to predict and apply relevant theorem rules, ultimately constructing a logical path that leads to the final solution. This paper also builds on this emerging direction, offering novel insights about the pivotal role of diagrammatic information.

In this paper, we propose Pi-GPS, unleashing the power of diagrammatic information for enhancing geometry problem solving. Our work is inspired by the observation that text often conveys ambiguity in ways that diagrams, by nature, cannot easily accommodate [32]. However current approaches typically parse text and diagrams independently, resulting in ambiguities remain unresolved in text and further undermines the subsequent theorem prediction stage as the predictor’s understanding of the problem is constrained. For instance, consider a text reference to ”a shape.” This could refer to a variety of geometric forms, such as a triangle, rectangle, or circle. Yet the text alone offers insufficient information to resolve the ambiguity. In contrast, we can easily disambiguate the reference when supported by a diagram, as the visual context clarifies the intended meaning. Figure 1 provides several examples that highlight the ambiguities present in the text.

In light of this, our objective is to enhance geometry problem solving by introducing a micro module that resolves textual ambiguities through the diagrammatic information. We identify three primary sources of these ambiguities: (1) unspecified points (e.g., missing point names), (2) unspecified shapes (e.g., missing shape names), and (3) unspecified areas (e.g., computing shaded areas). To address these, we leverage Multimodal Large Language Models (MLLMs) to develop an error-correcting tool, rectifier, capable of automatically detecting and rectifying these ambiguities given the diagram as input. Additionally, we design a verifier to mitigate MLLM’s hallucination by verifying the disambiguated text aligns with diagrammatic heuristics (e.g., closed-loop shapes), which we show is pivotal in the experiments. We also explore the impact of recent advanced LLMs for reasoning, o3-mini [24], in predicting theorem order, and present valuable analysis.

Experimentally we demonstrate our framework Pi-GPS, by resolving ambiguities in text, significantly outperforms state-of-the-art baselines on both Geometry3K [21] and PGPS9K [42] benchmarks. We hope this work will draw attention to the crucial need for resolving text ambiguity in formal language space, an aspect often overlooked in previous research, and underscores its significance in advancing geometry problem solving.

In summary, our key contributions are:

- **Perspective.** We identify that text ambiguity is a key factor hindering the performance of geometry problem solving, which has been overlooked in prior works.

- **Methodology.** We propose a micro module to address text ambiguity, comprising a rectifier and a verifier. The rectifier powered by a MLLM refines text with diagrammatic information, while the verifier ensures alignment with diagrammatic heuristics. These components work in tandem to reduce ambiguity, which is further evaluated on theorem prediction using a strong reasoning LLM.
- **Evaluation.** The resulting framework, Pi-GPS, achieves the state-of-the-art performance, with a nearly 10% improvement on Geometry3K over prior neural-symbolic approaches. We also provide strong evidence supporting the efficacy of the proposed module, and present an in-depth analysis.

2. Related Works

Geometry Problem Solving. Recent advancements in automated GPS [4, 28, 30] have attracted considerable attention due to the inherent complexity and unique challenges it presents. One prominent approach to GPS has been the use of language models that treat it as a specialized form of text generation. Notable examples include GeoQA [8] and Uni-Geo [9]. These models leverage large-scale pre-trained language models to generate solutions by interpreting geometry problems as text-based tasks. PGPS-Net [42] improved upon these models by enhancing the performance of neural network-based approaches. These methods struggle to accurately capture the complex relationships between geometric entities in diagrams. LANS [18] circumvents this issue by incorporating diagram annotation, however, such annotations are not always accessible. Additionally, the vector representations used by these models lack interpretability, resulting in unreliable or inconsistent solutions. In contrast, symbolic systems approach GPS from a more structured, interpretable angle, such as GEOS [29] and Inter-GPS [21]. They convert problem statements and diagrams into structured formats, enabling the application of symbolic solvers based on known geometric theorems. This approach enhances the interpretability of the problem solving process, yielding more precise solutions. A significant advancement in GPS was the introduction of PGDP [41], the first end-to-end diagram parsing method for geometry problems. This was further used in works like GeoDRL [25] and E-GPS [36], which enhanced solution accuracy and robustness through techniques such as theorem library augmentation and theorem sequence prediction. Our study introduces a novel micro module to resolve ambiguities in textual problem statements, which is orthogonal and can be plugged into existing neural-symbolic frameworks.

MLLMs for Mathematical Reasoning. Early research in multimodal learning focused on leveraging attention mechanisms to align image and text representations. A key breakthrough came with CLIP [27], which learned transferable visual representations through natural language su-

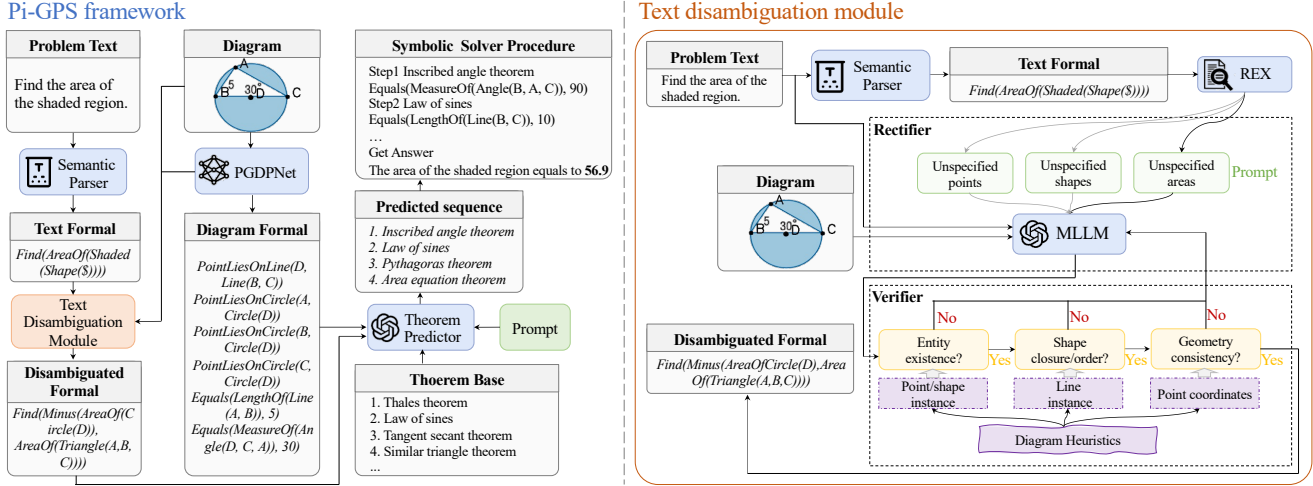


Figure 2. Illustrating the pipeline of our Pi-GPS: the overall framework is shown on the left and the text disambiguation module is depicted on the right, which plays a crucial role in resolving text ambiguity, enhancing performance. REX stands for regex pattern matching.

pervision, laying the foundation for subsequent large-scale multimodal models. Building on this, the LLaVA series [13, 19, 31] introduced visual instruction-tuning, linking a visual encoder to a language model via a simple multi-layer perceptron. Subsequent works [2, 7, 20, 34] expanded MLLMs by incorporating novel visual perception modules and hybrid vision encoders, enhancing their ability to address increasingly complex tasks. As model scale has grown, so too has their ability to perform sophisticated contextual and mathematical reasoning [5, 12, 26]. More recently, models like MathGLM-Vision [39] and Math-LLaVA [31] have introduced chain-of-thought reasoning and intermediate-step generation, enabling problem solving by breaking down complex tasks into manageable steps. Despite their impressive contextual reasoning abilities, MLLMs still face challenges such as hallucination, which is particularly problematic in mathematical reasoning tasks. In our work, we utilize MLLMs to generate diagrammatic information for text disambiguation, framing this as a more tractable task within the zero-shot capabilities of these models.

3. Method

In geometry problem solving, the *Diagram* represents the geometric figure, while the *Problem Text* provides the textual description, including the problem’s objective (e.g., “Find the length of EH”). The goal is to solve for the correct answer corresponding to the given pair. To achieve this, we propose **Pi-GPS**, as illustrated in Figure 2. The framework comprises a parser and a reasoner. Distinct from previous symbolic approaches, we propose a text disambiguation module that leverages the diagrammatic information to resolve text ambiguity, and we introduce a theorem predictor using an LLM given the disambiguated formal language.

3.1. Parser

Text Parser. A critical step in solving geometry problems is extracting relevant information from the problem statement, particularly identifying the premises and the goal of the problem. This extraction process can be categorized into rule-based methods and deep neural network-based methods. Traditional rule-based parsing techniques have been shown to provide relatively precise results. Although deep neural networks excel in sequence-to-sequence (Seq2Seq) tasks such as machine translation, previous research [21] suggests that Seq2Seq-based semantic parsers struggle with geometry problems. This is primarily due to the limited size of geometric datasets and the tendency of neural parsers to introduce noise into the output.

To achieve more accurate parsing, we utilize a rule-based text parser following [21, 25, 36]. This parser analyzes the problem text by applying regular expressions to identify basic elements, numerical values, and their interrelationships. The parser automatically generates a set of propositions P_T and identifies the problem target t^* from the text.

Diagram Parser. As with previous work [25, 36], we employ PGDPNet [41], an end-to-end neural network-based model that efficiently extracts basic elements such as points, lines, and circles from geometric figures along with their logical relationships, and generates a formal set of propositions. This approach achieves state-of-the-art performance in geometric diagram parsing.

Text Disambiguation Module. Previous studies [25, 36] have typically concatenated the parsing results from text and diagram parsers in a straightforward manner. While each parser performs well within its respective modality, the disconnect between textual and visual representations often leads to unresolved ambiguities, impeding subsequent solving process and thus degrading the overall accuracy. To

Problem Text	Formal / Disambiguated Formal	Diagram
The rectangle is inscribed into the circle . Find the exact circumference of the circle .	$\text{InscribedIn}(\text{Rectangle}(\$), \text{Circle}(\$))$ $\text{Find}(\text{CircumferenceOf}(\text{Circle}(\$)))$ $\text{InscribedIn}(\text{Rectangle}(A, B, E, D), \text{Circle}(C))$ $\text{Find}(\text{CircumferenceOf}(\text{Circle}(C)))$	
Q is the centroid and $BE = 9$. Find BQ .	$\text{IsCentroidOf}(\text{Point}(Q), \text{Shape}(\$))$ $\text{IsCentroidOf}(\text{Point}(Q), \text{Triangle}(A, C, B))$	
Find the area of the shaded region . Assume that the polygon is regular unless otherwise stated.	$\text{Regular}(\text{Polygon}(\$))$ $\text{Find}(\text{AreaOf}(\text{Shaded}(\text{Shape}(\$))))$ $\text{Regular}(\text{Triangle}(A, E, G))$ $\text{Find}(\text{Minus}(\text{AreaOf}(\text{Triangle}(A, E, G)), \text{AreaOf}(\text{Circle}(D))))$	

Figure 3. Illustrating several examples, showing the proposed text disambiguation module is capable of resolving text ambiguity.

overcome this issue, we propose a novel module incorporating a rectifier and a verifier, as illustrated in Figure 2.

(i) **Rectifier using MLLM.** The primary objective of the rectifier is to resolve ambiguities in the output of the text parser by leveraging the diagram through a MLLM. Upon analyzing the sources of ambiguity, we categorize the root causes into three distinct types, which are as follows:

- **Unspecified points:** The text parser can identify specific geometric shape but fail to associate them with explicit points. For example, in the relationship $\text{CircumscribedTo}(\text{Square}(\$), \text{Circle}(\$))$, the parser recognizes that a square is circumscribed to a circle but does not specify the defining points (vertices) of the square or circle.
- **Unspecified shapes:** The text parser, although capable of recognizing certain geometric constructs, fails to correctly map or associate them with predefined shapes or geometric entities in the system. This limitation is evident in expressions like $\text{IsAltitudeOf}(\text{Line}(C, P), \text{Shape}(\$))$, where the absence of explicit shape identification impedes effective interpretation.
- **Unspecified areas:** The text parser indicates that the formal language specifies graphical elements, such as $\text{Find}(\text{AreaOf}(\text{Shaded}(\text{Shape}(\$))))$. This implies the need to determine the area of a shaded region, typically representing areas of interest within a diagram.

The rectification process begins by employing regular expressions to identify unknown identifiers, represented by the symbol '\$', and determine the type of ambiguity. For each identified ambiguity, a specific prompt is crafted based on its nature, and the MLLM is used to resolve the issue by referencing both the *Diagram* and *Problem Text*. The incorporation of diagrammatic information is crucial, as it provides supplementary context to improve resolution accuracy. However, the potential for hallucination must be carefully managed, as it could compromise the accuracy of the rectification. Meanwhile, generating output in a formal

language poses a significant challenge for MLLMs, as absolute precision is essential, any deviation such as an incorrect character and misplaced parenthesis can invalidate the output. This motivates us to design a verifier based on diagram heuristics, ensuring its correctness.

(ii) **Verifier using Diagram Heuristics.** We propose a Logical Reasoning Verifier that utilizes diagram heuristics derived from the diagram parser to ensure the consistency of outputs generated by the MLLM with the provided geometric diagram. This verifier incorporates three key heuristics for examining the rectified output:

- **Entity existence verification.** MLLMs may generate geometric entities such as points, lines, or circles that do not correspond to actual elements in the diagram. We use the entity instances identified by the diagram parser to cross-check the MLLM's output, ensuring consistency with the original diagram.
- **Shape closure and order validation.** A common issue occurs when points within a geometric shape fail to form a closed figure or are ordered incorrectly. For instance, a pentagon labeled $\text{Pentagon}(A, B, D, E, C)$ may be erroneously output as $\text{Pentagon}(A, B, C, D, E)$. To rectify this, we construct a graph representing the diagram, checking for connectivity and cyclic properties, and ensure each node has the correct degree for valid closure. If the points are ordered incorrectly, we reorder the vertices based on the graph structure to ensure proper shape formation.
- **Geometry consistency of vertices.** The MLLM-generated vertices may not always align with the intended geometry shape. We apply analytical geometry techniques to verify that the vertices match the intended shape based on their coordinates.

When discrepancies arise, feedback from the verifier is incorporated into the rectifier, creating a loop that allows the MLLM to iteratively adjust and refine its output. Experimental results demonstrate that the verifier plays a crucial role in ensuring adherence, thereby enhancing geometric problem solving accuracy.

3.2. Reasoner

The reasoner typically comprises a predictor for theorem order prediction and a solver that applies the theorems in the predicted order to derive the final solution.

Theorem predictor. Accurately predicting the correct sequence of theorems is essential for deriving solutions and ensuring the interpretability. Previous approaches [21, 36] have used transformer-based models, framing theorem sequencing as a sequential prediction task. Additionally, some study [25] has applied reinforcement learning to improve theorem prediction accuracy. However, a major limitation of these methods is their reliance on annotated problem solving sequences for training, which are often scarce or expensive to generate, particularly in specialized domains

Methods	Accuracy	Steps	Question Type				Geometric Shape				
			Angle	Length	Area	Ratio	Line	Triangle	Quad	Circle	Other
Human [21]	56.9	–	53.7	59.3	57.7	42.9	46.7	53.8	68.7	61.7	58.3
Human Expert [21]	90.9	–	89.9	92.0	93.9	66.7	95.9	92.2	90.5	89.9	92.3
Gemini 2 [15]	60.7	–	58.9	61.8	57.5	68.8	54.1	62.7	45.5	57.7	58.3
Claude3.5 Sonnet [6]	56.4	–	54.9	57.3	53.6	64.6	49.4	58.6	40.9	57.9	53.9
GPT-4o [1]	58.6	–	55.6	59.3	55.1	70.6	51.4	60.4	43.1	59.0	56.7
Inter-GPS [21]	57.5	7.1	59.1	61.7	30.2	50.0	59.3	66.0	52.4	45.5	48.1
GeoDRL [25]	68.4	–	75.5	70.5	22.6	83.3	77.8	76.0	62.9	59.4	48.1
E-GPS [36]	67.9	1.63–2.28	78.3	67.2	27.7	72.2	76.1	75.6	59.4	55.0	51.8
Pi-GPS (ours)	77.8	2.31–4.12	83.9	81.4	59.0	81.2	79.6	83.9	76.4	73.0	69.4

Table 1. Comparison of geometry problem solving on the Geometry3K dataset. Our method consistently outperforms all baseline models. Accuracy, Steps, and additional metrics are reported for different question types and geometric shapes. Best results are highlighted in bold.

requiring domain-specific expertise. Building on recent advancements, we draw inspiration from AlphaGeo [33], which demonstrates the potential of LLMs in symbolic deduction. In this work, we explore the application of advanced LLMs, specifically the o3-mini [24], by prompting the model with a library of geometry theorem knowledge. The model then generates the most appropriate order of theorems based on disambiguated text and diagram formals. This approach reduces dependence on labeled data and leverages the generalization capabilities of modern LLMs.

Solver. Our solver framework builds upon the approach in [21] and incorporates the expanded theorem library from [25]. We have modified the logical framework to accommodate the extended formal language, specifically to address shadow regions and other special cases. The solver, along with the extended theorem library, is also employed in the experimental baselines for a fair comparison.

4. Experiments

4.1. Settings

Datasets. We conduct experiments using the Geometry3K [21] and PGPS9K [42] datasets. Geometry3K consists of 3,002 geometry problems, partitioned into 2,101 for training, 300 for validation, and 601 for testing. Each problem is accompanied by a geometric diagram, problem text, and formal language parsing annotations. It covers a diverse range of geometric shapes, including lines, triangles, circles, quadrilaterals, and other polygons, making it a comprehensive benchmark. PGPS9K, an expanded version of Geometry3K, contains 9,022 geometry problems paired with 4,000 unique diagrams. Of these, 2,891 problems with 1,738 diagrams are sourced from Geometry3K, while the remaining problems are collected from five widely-used mathematics textbooks for grades 6-12, covering nearly all plane geometry problem types for these educational levels.

Metrics. Building on the methodologies of prior studies [21, 42], we adopt two evaluation schemes: *Completion* and *Choice*, to assess the numerical performance of our

methods. The *Completion* metric gauges the model’s ability to generate the first executable solution program as its final output. The *Choice* metric measures the model’s ability to correctly select an option from four candidates, with random selection as a fallback when the generated answer does not match any provided options. Performance is evaluated based on accuracy.

Baselines. We conduct a comprehensive comparison between our proposed method and state-of-the-art models across various categories to analyze their performance in geometry problem solving tasks. For neural solvers, we evaluate several prominent models: NGS [8], which uses a ResNet-101 architecture for encoding geometric diagrams; Geoformer [10], which employs the VL-T5 model for diagram encoding followed by a Transformer-based processing architecture; SCA-GPS [23], which introduces a novel strategy for geometric problem-solving; PGPSNet [42], which combines CNN and GRU encoders to enhance geometric reasoning; LANS [18], a layout-aware neural solver; as well as GOLD [40], which converts geometry diagrams into natural language descriptions. In the realm of neural-symbolic solvers, we compare with the classical Inter-GPS [21] and two advanced models: GeoDRL [25], which improves Inter-GPS’s search strategy by integrating logical graph deduction and deep reinforcement learning; and E-GPS [36], which combines top-down and bottom-up reasoning to match the performance of other methods with fewer steps and improved explainability. Additionally, we report results from leading MLLMs, including Qwen-VL [7], GPT-4o [1], Gemini 2 [15], and Claude 3.5 Sonnet [6], which represent cutting-edge visual reasoning capabilities. It is important to note that our method does not require ground-truth parsing (neither diagram annotation nor text annotation). We adopt the expanded theorem set from GeoDRL, which is also utilized by other methods that require a theorem base for fair comparison.

4.2. Results

We present a detailed comparison of our method with both MLLMs and neural-symbolic baselines on the Geometry3K

Category	Method	Geometry3K		PGPS9K	
		Completion	Choice	Completion	Choice
MLLMs	Qwen-VL [7]	22.1	26.7	20.1	23.2
	GPT-4o [1]	34.8	58.6	33.3	51.0
	Claude 3.5 Sonnet [6]	32.0	56.4	27.6	45.9
	Gemini 2 [15]	38.9	60.7	38.2	56.8
Neural Methods	NGS [8]	35.3	58.8	34.1	46.1
	Geoformer [10]	36.8	59.3	35.6	47.3
	SCA-GPS [23]	-	76.7	-	-
	GOLD* [40]	-	62.7	-	60.6
	PGPSNet-v2-S* [43]	65.2	76.4	60.3	69.2
	LANS (Diagram GT)* [18]	72.1	82.3	66.7	74.0
Neural-symbolic Methods	Inter-GPS [21]	43.4	57.5	-	-
	GeoDRL [25]	57.9	68.4	55.6	66.7
	E-GPS [36]	-	67.9	-	-
	Pi-GPS (ours)	70.6	77.8	61.4	69.8

Table 2. Comparison on Geometry3K and PGPS9K. Our method achieves the best performance (highlighted in bold) compared to the neural-symbolic methods. Note that all baselines except LANS use parsed results, while LANS uses textual clauses and point positions from diagram annotations. * indicates that GOLD, PGPSNet and LANS are trained on the larger dataset, PGPS9K.

dataset, as summarized in Table 1. Our method consistently outperforms all baseline models, demonstrating superior performance, and even surpassing human experts in certain subcategories, such as the ratio question type. While MLLMs excel in general multimodal tasks, they exhibit limitations when applied to specialized mathematical geometry problems. These challenges stem from MLLMs’ difficulty in accurately parsing geometric diagrams, performing complex reasoning, and executing precise numerical computations. In contrast, our method not only achieves significantly better results but also offers greater interpretability. Compared to neural-symbolic baselines, our approach achieves the highest performance, with an impressive improvement of nearly 10% over the two strong baselines, E-GPS and GeoDRL. This improvement highlights the substantial impact of text ambiguity, an often overlooked factor in prior work. Our category analysis reveals that text ambiguity affects all categories, with the most significant impact observed in the area question type, where the text frequently refers to “the area” without a specific identifier, exacerbating the ambiguity.

To further validate the effectiveness of our approach, we present additional comparisons on the PGPS9K dataset for both completion and choice evaluation tasks, as shown in Table 2. Notably, compared to the state-of-the-art neural-symbolic method, GeoDRL, our method achieves improvements of 5.8% and 3.1% on the PGPS9K dataset in terms of completion and choice respectively. These results highlight the efficacy of our approach in interpreting the semantic intent of problem statements, a capability enabled by the integration of our text disambiguation module. Additionally, we provide a comprehensive comparison with

Text disam.	Theorem pred.	Completion	Choice	Steps
		60.7	70.6	2.85-6.03
✓		68.9	76.6	2.85-6.03
	✓	63.2	72.3	2.31-4.12
✓	✓	70.6	77.8	2.31-4.12

Table 3. Illustrating the effect of text disambiguation module (Text disam.) and theorem predictor (Theorem pred.) on Geometry3K. The text disambiguation module plays a critical role with its especially significant impact in driving performance improvement.

competitive neural methods. Notably, LANS [18], the top-performing neural model, relies on textual clauses and point positions derived from diagram annotations. This reliance on ground-truth annotations significantly boosts performance. Both LANS and PGPSNET were trained on the large-scale PGPS9K dataset. However, neural-based methods typically suffer from a lack of interpretability. In contrast, our method advances the field of interpretable geometry problem solving by addressing the critical issue of text ambiguity, a challenge often overlooked in previous work.

4.3. Analysis

We provide an in-depth analysis about the framework of our Pi-GPS by conducting comprehensive ablation studies on Geometry3K dataset.

Text disambiguation module is pivotal in Pi-GPS. We first conduct an ablation study to analyze the impact of two key components in Pi-GPS: the text disambiguation module and the theorem predictor utilizing an LLM. When the theorem predictor is disabled, a traversal strategy is employed. The comparison results are presented in Table 3. Notably,

Method	Completion	Choice
Ours w/o Text disam.	63.2	72.3
+ Rectifier (general prompt)	62.4	71.9
+ Rectifier (specific prompt)	64.2	73.3
+ Verifier	70.6	77.8

Table 4. Illustrating the roles of the rectifier and verifier in the text disambiguation module on Geometry3K.

the superior baseline performance relative to prior work is largely attributed to our self-trained PGDP model, which may exhibit enhanced diagram parsing capability. However, even with this strong baseline, both proposed components continue to significantly improve performance. Specifically, by integrating the proposed theorem predictor, the number of solving steps is reduced. More importantly, the text disambiguation module plays a critical role in Pi-GPS, with its impact being especially pronounced in driving overall performance improvement, a consistent enhancement of over 5% across all cases.

The verifier is critical in text disambiguation module.

We further examine the roles of the rectifier and verifier components within the text disambiguation module. The experimental results, summarized in Table 4, also report the influence of a tailored prompt designed to address specific text ambiguity scenarios identified via regular expression (regex) pattern matching. A key observation is that, without a tailored prompt, applying the general rectifier degrades performance due to hallucinations and uncertainties inherent in MLLMs. These factors introduce erroneous modifications, reducing disambiguation accuracy. In contrast, incorporating a tailored prompt improves performance beyond the baseline, underscoring the importance of domain-specific guidance in enhancing the rectifier’s effectiveness. This suggests that explicit contextual cues help mitigate unintended alterations and improve rectification precision. Additionally, the verifier significantly enhances disambiguation, yielding performance gains of 4%–6%. This highlights its critical role in enforcing consistency and correctness by systematically validating and refining rectified outputs. These findings collectively demonstrate that effective coordination between rectification and verification, along with domain-specific prompting, is essential for robust and accurate text disambiguation.

Theorem order prediction is a more manageable task for LLMs.

In our framework, we integrate an LLM to facilitate theorem prediction. A natural baseline is to directly employ an LLM to solve problems using the parsed formal representations. Given the growing interest in LLMs for mathematical reasoning, this approach warrants thorough investigation. We conduct experiments and evaluate LLMs and MLLMs on their ability to directly solve the problem. The results are summarized in Table 5. We have several

Task	Models	Completion	Choice
Direct solv. (MLLM)	GPT-4o	34.8	58.6
	Gemini 2	38.9	60.7
Direct solv. (LLM)	GPT-4o	36.5	59.7
	DeepSeek-R1	63.9	72.2
	o3-mini	66.4	75.5
	o3-mini w/o Text disam.	61.4	70.4
Theorem pred. (LLM)	o3-mini (ours)	70.6	77.8

Table 5. Illustrating different roles of (M)LLMs in GPS. While advanced LLMs exhibit strong mathematical reasoning in direct solution generation, our approach, leveraging LLMs for theorem prediction, improves both performance and interpretability.

key observations. (1) **MLLMs struggle with geometry reasoning from raw inputs.** When treated as LLMs and provided with parsed text and diagram formal representations, MLLM, GPT-4o in this case, outperform their direct processing of original problem text and diagrams. This suggests that current MLLMs face challenges in extracting logical information from visual diagrams. (2) **o3-mini exhibits superior reasoning capabilities.** Among the evaluated LLMs, o3-mini consistently achieves the best performance when directly applied to problem solving, reaffirming its effectiveness in mathematical reasoning tasks. (3) **Ambiguities in parsed input significantly degrade performance.** When tested with ambiguous parsed text and diagram representations, o3-mini’s accuracy drops substantially. This again validates our observation that disambiguating textual input is important to enhance the model’s reasoning capabilities, as even strong LLMs struggle with unresolved ambiguities in mathematical relationships. (4) **Theorem prediction enhances both accuracy and interpretability.** Rather than directly solving the problem, our method leverage LLMs to predict a sequence of theorems from a predefined theorem base, followed by a dedicated solver. This structured approach not only improves accuracy but also enhances interpretability, which is particularly valuable in educational settings where step-by-step justifications are crucial.

The effect of different LLMs in theorem predictor. We further conduct an ablation study on different LLMs used in our theorem predictor, with results on the Geometry3K dataset presented in Table 6. Compared to the vanilla traversal-based approach, incorporating LLMs improves solving accuracy while reducing the number of steps required, thereby enhancing overall efficiency. Notably, all evaluated LLMs achieve comparable performance, suggesting that theorem order prediction is a well-developed application for LLMs, demonstrating their robustness and reliability in this task.

The effect of different MLLMs in rectifier. In our approach, the rectifier within the text disambiguation module utilizes MLLMs to enhance performance. To explore the impact of different MLLMs on the rectification pro-

Predictor	Completion	Choice	Steps
Traversal	68.9	76.6	2.85 - 6.03
Claude 3.5 Sonnet	70.2	77.5	2.75 - 4.60
GPT-4o	69.4	77.2	2.62 - 4.41
Gemini 2	69.8	77.2	2.52 - 4.29
o3-mini	70.6	77.8	2.31 - 4.12

Table 6. Illustrating the effect of different LLMs used in theorem prediction on Geometry3K. All evaluated LLMs achieve comparable performance.

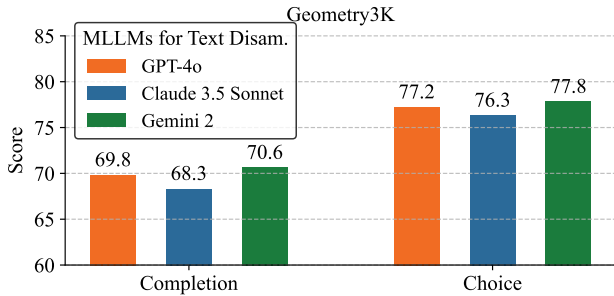


Figure 4. Illustrating the effect of different MLLMs used in rectifier within text disambiguation module.

cess, we conduct an ablation study evaluating multiple well-established MLLMs and present the results in Figure 4. The findings indicate that all tested models, regardless of their inherent capabilities and design variations, yield substantial improvements over the baseline. This demonstrates that our method’s effectiveness is not dependent on a specific MLLM but rather highlights its robustness and broad applicability across diverse architectures, reinforcing its generalizability.

5. Conclusion

In this work, we present Pi-GPS, a novel framework that integrates diagrammatic information to enhance geometry problem solving by resolving textual ambiguities. Central to our approach is a rectifier-verifier module, where the rectifier leverages MLLMs to refine textual descriptions using diagrammatic context, while the verifier ensures geometric consistency. This framework significantly improves problem representation, thereby improving the problem solving performance. Empirical evaluations on the Geometry3K and PGPS9K benchmarks demonstrate that Pi-GPS outperforms state-of-the-art neural-symbolic methods, achieving nearly a 10% performance gain on Geometry3K. These results advocate the critical role of ambiguity resolution in multimodal mathematical reasoning, a challenge that has been largely overlooked and warrants greater attention from the research community.

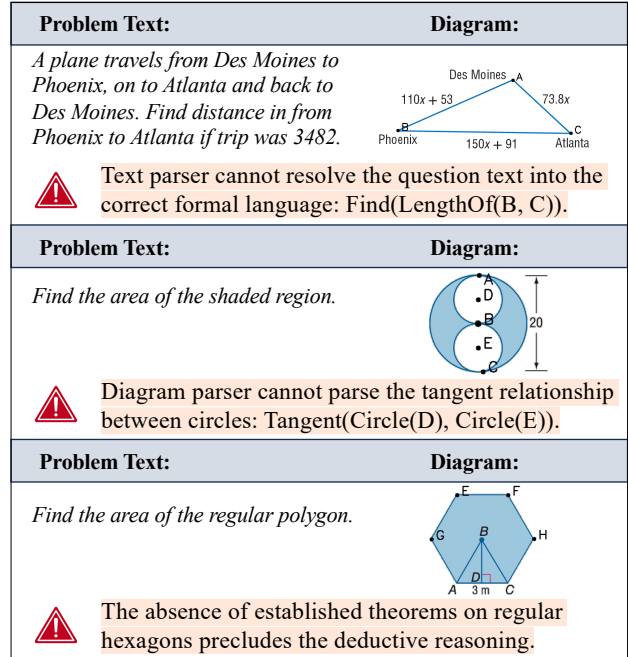


Figure 5. Illustrating the limitations in current GPS framework.

6. Limitations

While this work successfully identifies text ambiguity and introduces a dedicated module to resolve it, significantly enhancing system performance. Several limitations still remain as illustrated in Figure 5. These limitations highlight key areas for future improvement and offer directions for advancing automated geometric problem solving systems.

- **Limited Text Parsing Capability:** The current text parser struggles to accurately map certain syntactic variations to their formalized representations. Despite the integration of our text disambiguation module, these challenges persist, often leading to incomplete or erroneous formalizations.
- **Inadequate Diagram Parsing for Complex Relations:** The diagram parser struggles to accurately identify complex geometric relationships, such as tangency, due to their subtle and often implicit nature. This limitation hampers precise geometric analysis and interpretation, as misrecognition can distort structural understanding and compromise downstream computations.
- **Insufficient Theorem Base:** The absence of essential theorems necessary for solving specific problem categories significantly constrains the system’s ability to generate comprehensive and accurate solutions. For example, in the case of a regular hexagon, the fundamental theorem asserting that each interior angle measures 120 degrees is critical for various geometric deductions.

Acknowledgments

This work was jointly supported by the Fundamental Research Funds for the Central Universities (2243100004), the Beijing Municipal Science and Technology Project (Z241100001324011), and the National Natural Science Foundation of China (62437001).

References

- [1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 5, 6
- [2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022. 3
- [3] DeepMind AlphaProof and AlphaGeometry Teams. Ai achieves silver-medal standard solving international mathematical olympiad problems. Online resource, 2024. Available from <https://deepmind.google/discover/blog/ai-solves-imo-problems-at-silver-medal-level/>. 1
- [4] Chris Alvin, Sumit Gulwani, Rupak Majumdar, and Supratik Mukhopadhyay. Synthesis of solutions for shaded area geometry problems. In *30th International Florida Artificial Intelligence Research Society Conference*, pages 14–19. AAAI, 2017. 1, 2
- [5] Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2357–2367, 2019. 3
- [6] Anthropic. Claude 3.5 sonnet. Online resource, 2024. Available from <https://www.anthropic.com/news/claude-3-5-sonnet>. 5, 6
- [7] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization. *Text Reading, and Beyond*, 2, 2023. 3, 5, 6
- [8] Jiaqi Chen, Jianheng Tang, Jinghui Qin, Xiaodan Liang, Lingbo Liu, Eric Xing, and Liang Lin. GeoQA: A geometric question answering benchmark towards multimodal numerical reasoning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 513–523, Online, 2021. Association for Computational Linguistics. 1, 2, 5, 6
- [9] Jiaqi Chen, Tong Li, Jinghui Qin, Pan Lu, Liang Lin, Chongyu Chen, and Xiaodan Liang. Unigeo: Unifying geometry logical reasoning via reformulating mathematical expression. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3313–3323, 2022. 2
- [10] Jiaqi Chen, Tong Li, Jinghui Qin, Pan Lu, Liang Lin, Chongyu Chen, and Xiaodan Liang. Unigeo: Unifying geometry logical reasoning via reformulating mathematical expression. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3313–3323, 2022. 5, 6
- [11] Yuri Chervonyi, Trieu H Trinh, Miroslav Olšák, Xiaomeng Yang, Hoang Nguyen, Marcelo Menegali, Junehyuk Jung, Vikas Verma, Quoc V Le, and Thang Luong. Gold-medalist performance in solving olympiad geometry with alphageometry2. *arXiv preprint arXiv:2502.03544*, 2025. 1
- [12] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021. 3
- [13] Jiahui Gao, Renjie Pi, Jipeng Zhang, Jiacheng Ye, Wanjun Zhong, Yufei Wang, Lanqing Hong, Jianhua Han, Hang Xu, Zhenguo Li, et al. G-llava: Solving geometric problem with multi-modal large language model. *arXiv preprint arXiv:2312.11370*, 2023. 1, 3
- [14] Herbert Gelernter, James R Hansen, and Donald W Loveland. Empirical explorations of the geometry theorem machine. In *Papers presented at the May 3-5, 1960, western joint IRE-AIEE-ACM computer conference*, pages 143–149, 1960. 1
- [15] Google Gemini Team. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 5, 6
- [16] Zhenya Huang, Qi Liu, Weibo Gao, Jinze Wu, Yu Yin, Hao Wang, and Enhong Chen. Neural mathematical solver with enhanced formula structure. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1729–1732, 2020. 1
- [17] Zhihao Li, Yao Du, Yang Liu, Yan Zhang, Yufang Liu, Mengdi Zhang, and Xunliang Cai. Eagle: Elevating geometric reasoning through llm-empowered visual instruction tuning. *arXiv preprint arXiv:2408.11397*, 2024.
- [18] Zhong-Zhi Li, Ming-Liang Zhang, Fei Yin, and Cheng-Lin Liu. Lans: A layout-aware neural solver for plane geometry problem. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 2596–2608, 2024. 1, 2, 5, 6
- [19] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024. 3
- [20] Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, et al. Deepseek-vl: towards real-world vision-language understanding. *arXiv preprint arXiv:2403.05525*, 2024. 3
- [21] Pan Lu, Ran Gong, Shibiao Jiang, Liang Qiu, Siyuan Huang, Xiaodan Liang, and Song-Chun Zhu. Inter-gps: Interpretable geometry problem solving with formal language and sym-

- bolic reasoning. In *The 59th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2021. 1, 2, 3, 4, 5, 6
- [22] Seyed Iman Mirzadeh, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. GSM-symbolic: Understanding the limitations of mathematical reasoning in large language models. In *The Thirteenth International Conference on Learning Representations*, 2025. 1
- [23] Maizhen Ning, Qiu-Feng Wang, Kaizhu Huang, and Xiaowei Huang. A symbolic characters aware model for solving geometry problems. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 7767–7775, 2023. 5, 6
- [24] OpenAI. Openai o3-mini pushing the frontier of cost-effective reasoning. Online resource, 2025. Available from <https://openai.com/index/openai-o3-mini/>. 2, 5
- [25] Shuai Peng, Di Fu, Yijun Liang, Liangcai Gao, and Zhi Tang. GeoDRL: A self-learning framework for geometry problem solving using reinforcement learning in deductive reasoning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13468–13480, Toronto, Canada, 2023. Association for Computational Linguistics. 1, 2, 3, 4, 5, 6
- [26] Stanislas Polu and Ilya Sutskever. Generative language modeling for automated theorem proving. *arXiv preprint arXiv:2009.03393*, 2020. 3
- [27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2
- [28] Mrinmaya Sachan and Eric Xing. Learning to solve geometry problems from natural language demonstrations in textbooks. In *Proceedings of the 6th joint conference on lexical and computational semantics (*SEM 2017)*, pages 251–261, 2017. 1, 2
- [29] Minjoon Seo, Hannaneh Hajishirzi, Ali Farhadi, Oren Etzioni, and Clint Malcolm. Solving geometry problems: Combining text and diagram interpretation. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1466–1476, 2015. 2
- [30] Minjoon Seo, Hannaneh Hajishirzi, Ali Farhadi, Oren Etzioni, and Clint Malcolm. Solving geometry problems: Combining text and diagram interpretation. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1466–1476, 2015. 1, 2
- [31] Wenhao Shi, Zhiqiang Hu, Yi Bin, Junhua Liu, Yang Yang, See Kiong Ng, Lidong Bing, and Roy Lee. Math-llava: Bootstrapping mathematical reasoning for multimodal large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 4663–4680, 2024. 1, 3
- [32] Keith Stenning and Jon Oberlander. A cognitive theory of graphical and linguistic reasoning: Logic and implementation. *Cognitive science*, 19(1):97–140, 1995. 2
- [33] Trieu H Trinh, Yuhuai Wu, Quoc V Le, He He, and Thang Luong. Solving olympiad geometry without human demonstrations. *Nature*, 625(7995):476–482, 2024. 1, 5
- [34] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 3
- [35] Wu Wen-Tsun. Basic principles of mechanical theorem proving in elementary geometries. *Journal of automated Reasoning*, 2:221–252, 1986. 1
- [36] Wenjun Wu, Lingling Zhang, Jun Liu, Xi Tang, Yaxian Wang, Shaowei Wang, and Qianying Wang. E-gps: Explainable geometry problem solving via top-down solver and bottom-up generator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13828–13837, 2024. 1, 2, 3, 4, 5, 6
- [37] Renqiu Xia, Mingsheng Li, Hancheng Ye, Wenjie Wu, Hongbin Zhou, Jiakang Yuan, Tianshuo Peng, Xinyu Cai, Xiangchao Yan, Bin Wang, et al. Geox: Geometric problem solving through unified formalized vision-language pre-training. *arXiv preprint arXiv:2412.11863*, 2024. 1
- [38] Tong Xiao, Jiayu Liu, Zhenya Huang, Jinze Wu, Jing Sha, Shijin Wang, and Enhong Chen. Learning to solve geometry problems via simulating human dual-reasoning process. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 6559–6568, 2024. 1
- [39] Zhen Yang, Jinhao Chen, Zhengxiao Du, Wenmeng Yu, Weihang Wang, Wenyi Hong, Zhihuan Jiang, Bin Xu, Yuxiao Dong, and Jie Tang. Mathglm-vision: Solving mathematical problems with multi-modal large language model. *arXiv preprint arXiv:2409.13729*, 2024. 3
- [40] Jiaxin Zhang and Yashar Moshfeghi. Gold: Geometry problem solver with natural language description. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 263–278, 2024. 1, 5, 6
- [41] Ming-Liang Zhang, Fei Yin, Yi-Han Hao, and Cheng-Lin Liu. Plane geometry diagram parsing. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 1636–1643, 2022. 2, 3
- [42] Ming-Liang Zhang, Fei Yin, and Cheng-Lin Liu. A multi-modal neural geometric solver with textual clauses parsed from diagram. In *IJCAI*, 2023. 1, 2, 5
- [43] Ming-Liang Zhang, Zhong-Zhi Li, Fei Yin, Liang Lin, and Cheng-Lin Liu. Fuse, reason and verify: Geometry problem solving with parsed clauses from diagram. *arXiv preprint arXiv:2407.07327*, 2024. 6
- [44] Renrui Zhang, Xinyu Wei, Dongzhi Jiang, Ziyu Guo, Shicheng Li, Yichi Zhang, Chengzhuo Tong, Jiaming Liu, Aojun Zhou, Bin Wei, et al. Mavis: Mathematical visual instruction tuning with an automatic data engine. *arXiv preprint arXiv:2407.08739*, 2024.
- [45] Wenwen Zhuang, Xin Huang, Xiantao Zhang, and Jin Zeng. Math-puma: Progressive upward multimodal alignment to enhance mathematical reasoning. *arXiv preprint arXiv:2408.08640*, 2024. 1

- [46] Jia Zou, Xiaokai Zhang, Yiming He, Na Zhu, and Tuo Leng. Fgeo-drl: Deductive reasoning for geometric problems through deep reinforcement learning. *Symmetry*, 16(4), 2024. [1](#)