GEOFLIP: DO MLLMS KNOW WHETHER GEOMETRY PROBLEMS ARE UNDERDETERMINED?

Anonymous authors
Paper under double-blind review

ABSTRACT

Advances in supervised fine-tuning and reinforcement learning have substantially improved the mathematical reasoning capabilities of multimodal large language models (MLLMs), enabling them to solve geometry problems when sufficient evidence is provided. However, whether MLLMs can reliably recognize underdetermination induced by removing essential visual evidence remains unclear. Existing reliability evaluations mainly focus on missing textual conditions, leaving the role of visual evidence in multimodal mathematical reasoning insufficiently explored. To address this gap, we evaluate MLLMs on paired solvable and underdetermined geometry problems, jointly assessing problem-solving ability and underdetermination recognition. Specifically, we propose Key-Information Removal (KIR), which uses source solutions to guide the removal of essential visual evidence from geometry images. To ensure the validity of the resulting problems, automatic problem auditing, targeted repair, and human verification are further applied so that the remaining conditions no longer uniquely determine the target. Through this process, we construct GeoFlip, a benchmark comprising hundreds of solvable–underdetermined geometry problem pairs. Experiments on GeoFlip reveal a substantial gap between problem-solving ability and underdetermination recognition, with models frequently repeating source answers even after necessary evidence has been removed. Moreover, underdetermination recognition varies substantially with the type and modality of removed evidence, while model scaling and reasoning modes do not consistently improve underdetermination recognition. Finally, to improve reasoning reliability in both solvable and underdetermined settings, we develop a unified symbolic reasoning pipeline that combines multimodal formalization with geometric deduction and algebraic computation. Compared with direct answering, our pipeline raises pair success rate from 0.3% to 59.5% with Claude Opus 5 and from 35.1% to 57.2% with GPT-5.5.

1 INTRODUCTION

Large language models (LLMs), such as GPT-4, demonstrate strong language understanding and reasoning capabilities (OpenAI, 2024). Efforts to strengthen mathematical reasoning include continued pre-training on curated mathematical corpora (Shao et al., 2024) and supervised fine-tuning (SFT) on synthetic data (Yu et al., 2024). Reinforcement learning (RL) with verifiable rewards further improves reasoning, as in DeepSeek-R1 (Guo et al., 2025). Multimodal large language models (MLLMs) build on LLMs by incorporating visual perception, enabling mathematical problem solving from textual questions and images (Gao et al., 2025; Ning et al., 2025).

Despite these advances, LLMs remain prone to hallucinations, producing convincing but factually incorrect statements (Tian et al., 2024). MLLMs may rely more heavily on language priors than on visual evidence (Guan et al., 2024) or derive incorrect mathematical solutions from misread images (Zhang et al., 2024). When essential information is missing, models may fabricate additional premises and use them to derive invalid answers (Li et al., 2024; Liu et al., 2026). The missing-condition evaluations in MathCheck-GEO and TUBench's geometry subset modify textual questions (Zhou et al., 2024; He et al., 2024). Subsequent missing-premise studies and ReliableMath also examine textual condition removal (Fan et al., 2025; Xue et al., 2025). MM-AQA further evaluates visual information loss through cropping, masking, darkening, and blurring in its STEM subset (Madhusudhan et al., 2026). For geometry problems, a more specific question remains un-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

derexplored: **can models recognize when removing necessary geometric information from an image makes a previously solvable problem underdetermined?** Here, the remaining conditions are consistent but do not uniquely determine the target.

To this end, we propose Key-Information Removal (KIR) to construct GeoFlip, a benchmark of hundreds of human-verified solvable–underdetermined geometry pairs from the Geometry3K test split (Lu et al., 2021). KIR selects visual evidence used in the source solution and removes it through localized editing with GPT Image 2 (OpenAI, 2026b). KIR then automatically audits the edited image–question pairs and repairs them as needed. Human verification and correction ensure the validity of the final pairs. Models process the original and generated problems independently, allowing joint assessment of geometry-solving ability and underdetermination recognition.

We evaluate MLLMs and geometry-specialized models on GeoFlip pairs, with the following findings. First, even models with strong problem-solving performance may still give numerical answers to underdetermined problems. Second, "memorization" is widespread: models often repeat their source answers after necessary evidence is removed. Third, scaling and thinking mode can improve problem-solving performance but do not guarantee better underdetermination recognition. Fourth, symbol-only and image-only removals are the most challenging for underdetermination recognition across visual evidence types and removal modalities, respectively. We also study the effects of generation budgets and examine model rethinking through response lengths and keyword distributions.

To improve performance on both source problems and their underdetermined counterparts, we develop a unified symbolic reasoning pipeline. A multimodal formalizer converts each question and image into a formal language, which is compiled into a problem state for the PyEuclid symbolic solver (Li et al., 2025). The solver combines geometric deduction with algebraic computation. It returns a numerical value when the target is uniquely determined, or UNDETERMINED when the target remains unresolved and no further information can be derived. We measure joint performance using Pair Success Rate (PSR), which requires a correct source answer and recognition of its counterpart's underdetermination. Compared with direct answering, our method raises PSR from 0.3% to 59.5% with Claude Opus 5 and from 35.1% to 57.2% with GPT-5.5 (Table 2).

Our main contributions are summarized as follows: **(1)** We introduce the first systematic study of underdetermination recognition after visual evidence removal in multimodal geometry reasoning. **(2)** We propose KIR and construct GeoFlip, a benchmark containing 587 human-verified solvable–underdetermined geometry pairs for evaluating problem solving and underdetermination recognition. **(3)** We conduct extensive evaluations of MLLMs and geometry-specialized models, revealing a substantial gap between solving geometry problems and recognizing insufficient information after evidence removal. **(4)** We develop a unified pipeline combining multimodal formalization and symbolic reasoning, improving both underdetermination recognition and pair-level performance.

2 RELATED WORK

Reliable mathematical reasoning. Reliability evaluations examine robustness to variations in mathematical problems. GSM-Plus introduces diverse perturbations, including question wording and arithmetic variations (Li et al., 2024), GSM-Symbolic varies numerical values (Mirzadeh et al., 2024), and the functional variant of MATH assesses generalization across instances with the same reasoning pattern (Srivastava et al., 2024). MATH-Perturb studies changes that invalidate the original solution strategy (Huang et al., 2025), while RoMMath evaluates multimodal reasoning robustness under textual and visual perturbations (Zhao et al., 2025). MathVerse examines visual dependence by varying how information is distributed between text and images (Zhang et al., 2024). For incomplete problems, GSM-Plus also includes missing-condition questions (Li et al., 2024), and MathCheck explicitly evaluates answerability (Zhou et al., 2024). Missing-premise evaluations reveal overthinking without appropriate abstention (Fan et al., 2025). ReliableMath constructs unsolvable questions through premise removal and contradiction (Xue et al., 2025), while Liu et al. (2026) analyze abstention failures and propose inference-time intervention. In broader multimodal settings, TUBench removes textual conditions from geometry questions and obscures data in table images (He et al., 2024). MM-AQA uses question-aware transformation selection, including cropping, masking, darkening, and blurring for STEM questions (Madhusudhan et al., 2026). However, existing studies mainly address textual incompleteness and answerability, whereas GeoFlip studies underdetermination induced by multimodal evidence removal in geometry problems.

Multimodal large language models. MLLMs connect visual encoders with LLMs to follow instructions involving images and text (Liu et al., 2023). Their mathematical capabilities are evaluated on geometry calculation (Lu et al., 2021; Zhang et al., 2023), geometric proving (Chen et al., 2022), and mathematical questions in diverse visual contexts (Lu et al., 2024). G-LLaVA uses geometry-specific alignment and instruction tuning (Gao et al., 2025), while Math-PUMA aligns next-token distributions across textual and visual versions of the same problem (Zhuang et al., 2025). MM-Eureka further optimizes multimodal reasoning through RL with answer-correctness and format rewards (Meng et al., 2025). Formalized representations provide another route: GeoX aligns images with structured descriptions and generates solution programs for symbolic execution (Xia et al., 2025). Our pipeline uses multimodal formalization and a shared symbolic inference procedure to handle solvable and underdetermined problems.

3 GeoFlip Benchmark

3.1 Benchmark Objective and Construction Criteria

GeoFlip evaluates multimodal geometry reasoning in both solvable and underdetermined settings. A reliable model should solve a problem when sufficient conditions are available, while also recognizing when it becomes underdetermined after relevant evidence is removed from the problem.

Definition of Solvability. A problem is *solvable* when its textual conditions and visual evidence jointly provide sufficient information to derive a valid reasoning chain and uniquely determine the target. We focus on *underdetermined* problems, where key textual or visual evidence required for solving the original problem is removed. After removal, the remaining conditions are still consistent but insufficient to reconstruct a valid derivation or uniquely determine the target.

We construct each pair according to four criteria. **(1) Source validity.** The source problem must be coherent and solvable, and its reference solution must correctly derive the answer. **(2) Evidence relevance.** The initial edit removes visual evidence used by the source solution. Further edits remove evidence that supports alternative solutions or repeats the deleted information. **(3) Content preservation.** Edits preserve the target, unrelated textual and visual content, leaving a coherent question and a visually natural image. **(4) Counterpart underdetermination.** The remaining conditions do not uniquely determine the target, including through alternative derivations. Thus, disrupting the original solution is necessary for construction but is not sufficient for accepting a counterpart.

3.2 Solvable Data Collection

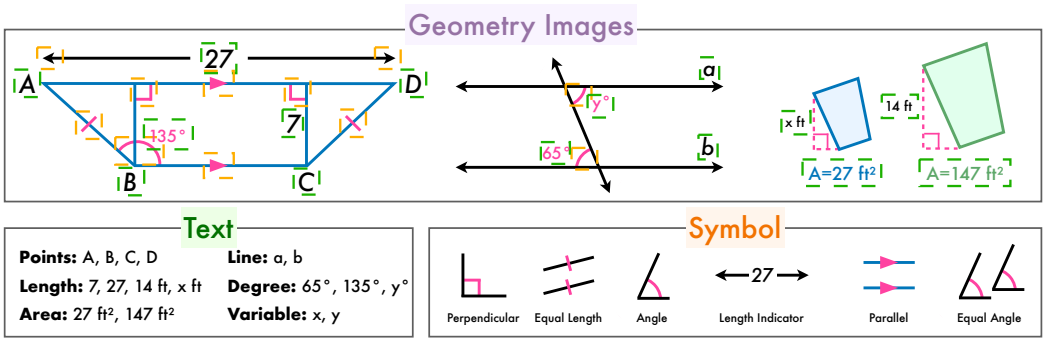


Figure 1: Examples of visual-evidence annotations on three geometry images. Green boxes localize text notations, and orange boxes localize symbol notations.

Solvable Geometry Problem. We represent each solvable source problem as $\mathrm{GP} = (I, Q)$, where I is the geometry image and Q is the textual question. Together, they provide sufficient conditions for a valid derivation of a unique target value. An accompanying solution program SP derives this value through an ordered sequence of operations:

$$\mathrm{SP} = \langle \rho_1, \ldots, \rho_T \rangle, \qquad \rho_t = (\mathrm{op}_t, \mathrm{args}_t),$$

where op_t denotes a predefined operation in PGPS9K's solution-program language (Zhang et al., 2023), and args_t specifies its operands: constants, variables, or intermediate results. For example, the program in Figure 2, `Equal 19 3y+7 Get y`, equates the congruent side lengths as $19 = 3y + 7$ and returns the solution for y. Thus, SP explicitly represents one valid reasoning process for solving the problem without enumerating all possible derivations.

Visual Evidence Annotations. Beyond its geometric layout, an image expresses problem conditions through visible notations. We group these notations into two annotation types: *text* and *symbol*. Text notations label geometric objects or state quantities, including point and line labels, variables, lengths, angle measures, and areas. Symbol notations represent geometric relations or measurement conventions, including perpendicularity, equality of lengths or angles, parallelism, and length indicators. We denote their annotations by $\mathcal{L} = \{(v_k, b_k)\}_{k=1}^{m(I)}$, where $m(I)$ is the number of annotated notations in I, v_k is the k-th visible text or symbol notation, and b_k is its annotated bounding box. Figure 1 illustrates both categories. Appendix A gives the complete annotation taxonomy.

Solvable GP Collection. KIR requires bounding-box annotations $\mathcal{L}$ to locate visual evidence for removal. We therefore use Geometry3K problems (Lu et al., 2021) included in PGPS9K (Zhang et al., 2023), which provides SP and $\mathcal{L}$. For a fair comparison with existing geometry-solving systems, we restrict the collection to the Geometry3K test split.

3.3 UNDERDETERMINED COUNTERPART REPRESENTATION

For each source GP, we construct a counterpart $\mathrm{GP}^* = (I^*, Q^*)$ by removing key information needed to solve the problem and provide a *removal rationale* (RR) explaining why the original target becomes underdetermined. Here, I^* and Q^* are the generated image and question. Together, GP and GP^* form a matched GeoFlip pair.

3.4 GEOFLIP CONSTRUCTION

We construct each GP^* from a solvable GP through a four-stage MLLM-assisted *Key-Information Removal* (KIR) pipeline: *KIR proposal*, *image editing*, *problem audit*, and *problem repair* (Figure 2). We use GPT-5.5 (high) (OpenAI, 2026a) for KIR proposal, problem audit, and textual repair, and GPT Image 2 (OpenAI, 2026b) for all image edits. Prompts are provided in Appendix B.

Step 1: KIR Proposal. Given GP and $\mathcal{L}$, KIR first checks the visual-evidence relevance criterion and skips the source problem if it is not met. For an eligible problem, it traces SP to identify the relevant visual evidence used in the derivation and randomly selects one localized item for removal. It outputs RR, and a grounded edit instruction (EI) specifying what and where to erase. When necessary, it also forms Q^* by removing text that reconstructs the selected evidence or enables an independent solution route.

Step 2: Image Editing. Given I and EI, the image editor erases the specified visual evidence within its annotated region. It reconstructs the local background and preserves all unrelated geometry and notations, yielding a candidate I^*. It neither redraws the full image nor adds replacement clues.

Step 3: Problem Audit. Before accepting a candidate, KIR evaluates the current (I^*, Q^*) against the source problem through separate question-quality and image-quality checks. *Question quality* checks whether Q^* retains conditions that reconstruct the removed information or support a solution to the target. *Image quality* checks whether I^* retains such visual evidence, as well as complete erasure, preservation of unrelated content, and visual naturalness. Both checks assess the quality of (I^*, Q^*) against the source pair (I, Q) as a reference and return feedback for the corresponding repairs. As illustrated in Figure 2, erasing $3y + 7$ still leaves two routes to $y = 4$: the right-angle mark gives $20y + 10 = 90$, while the congruence premise together with collinear Y–X–Z also implies a right angle. The retained right-angle mark and congruence premise cause the image- and question-quality failures, respectively. Appendix C.1 details both derivations.

Step 4: Problem Repair. Guided by feedback from failed or inconclusive checks, KIR revises GP^*, RR, and EI as needed, and the image editor applies visual corrections while preserving unaffected content. A repair round can update both modalities when both checks fail. In Figure 2, repair erases

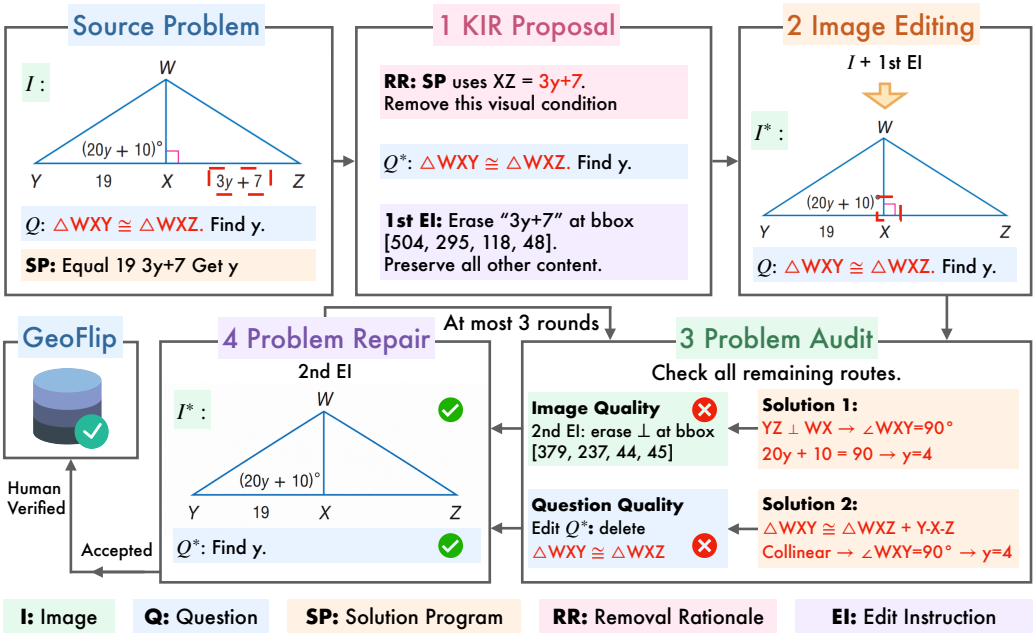


Figure 2: GeoFlip construction. KIR proposal and image editing remove solution-relevant visual evidence. Problem audit detects surviving solution routes and editing errors to guide targeted repair. Repaired pairs are re-audited, and accepted pairs undergo final human verification and correction.

Table 1: GeoFlip removal statistics. Left: scope of removal evidence across modalities. Right: type of removed visual evidence. Text, symbol, and both refer to visible notations.

Evidence removal scope			Visual evidence type		
All	Image-only	Image + Question	Text-only	Symbol-only	Both
587 (100.0%)	492 (83.8%)	95 (16.2%)	392 (66.8%)	110 (18.7%)	85 (14.5%)

the right-angle mark and deletes the congruence premise, leaving Q^* = "Find y". The complete pair returns to Step 3 after repair and is accepted only when both checks pass. A joint pass records an automatic pass, while unresolved failures after three repair rounds will undergo human verification.

Human Verification. We assign three annotators to review every GP* against the four criteria. Identified errors are corrected before all three recheck the pair. Only pairs approved by all three enter GeoFlip. The complete repair and agreement procedures are described in Appendix C.2.

3.5 Benchmark Statistics

GeoFlip retains 587 of the 589 source problems. Appendix C.3 reports statistics for each construction stage, and Appendix C.4 presents the two skipped questions. As shown in Table 1, most GeoFlip pairs involve image-only removal (83.8%), while the remaining pairs remove evidence from both the image and question (16.2%). Regarding the type of removed visual evidence, text notations account for the majority of removals (66.8%), followed by symbol notations (18.7%) and joint removal of both types (14.5%). These categories are defined in Section 3.2 and may overlap within a pair.

4 Unified Symbolic Reasoning Method

Figure 3 presents our pipeline applied to the GP* and GP: a multimodal formalizer parses each problem into a formal language, which is compiled into a problem state for the symbolic solver.

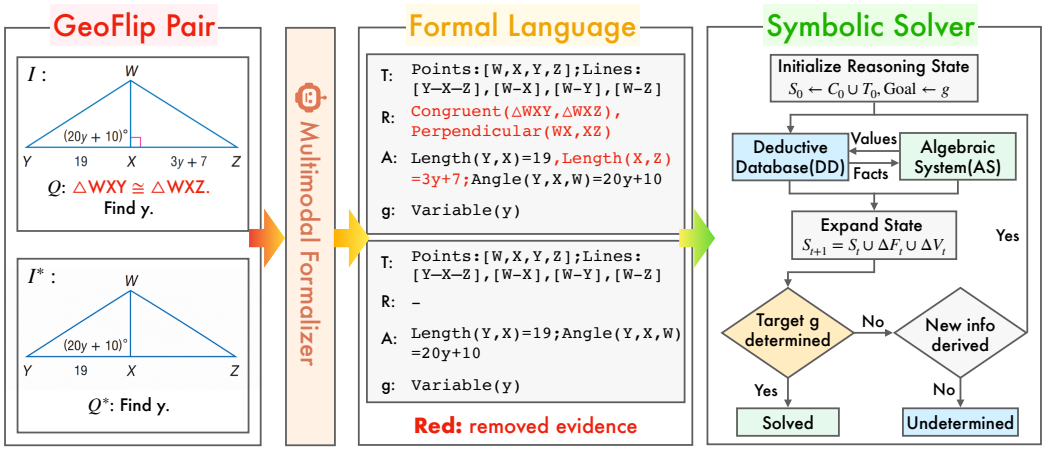


Figure 3: Overview of our unified symbolic reasoning. A multimodal formalizer parses the problem into a formal language, which is compiled into a problem state passed to the symbolic solver.

Multimodal Formalization. Given a geometry image I and a question Q, the multimodal formalizer F_θ converts them into a formal language (Lu et al., 2021):

$$F_\theta(I, Q, \mathcal{E}_N) = (\mathcal{T}, \mathcal{R}, \mathcal{A}, g).$$

where $\mathcal{E}_N$ denotes a few-shot prompt with task instructions and N demonstrations. $\mathcal{T}$ records the geometric objects and the arrangement of points on lines and circles, $\mathcal{R}$ contains geometric relations such as parallelism, $\mathcal{A}$ contains numerical values and algebraic equations, and g specifies the target. Our pipeline uses GPT-5.5 or Claude Opus 5 as the multimodal formalizer.

Formal Problem State. Following PyEuclid (Li et al., 2025), we compile the parsed formal language into an initial state S_0 for the target g through the mapping

$$S_0 = \mathcal{C}_0 \cup \mathcal{T}_0, \qquad \mathcal{C}_0 = \mathcal{R} \cup \mathcal{A}, \qquad \mathcal{T}_0 = \mathcal{T}.$$

Here, $\mathcal{T}_0$ records how the geometric objects are arranged, including collinearity and point order. The conditions $\mathcal{C}_0$ contain the geometric relations and algebraic constraints used for deduction and equation solving. The symbolic solver then reasons from S_0 toward the target g. The complete formalization prompt and demonstrations are described in Appendix E.2 and E.3.

Symbolic Solver. We use PyEuclid (Li et al., 2025) as the symbolic reasoning backend, combining geometric deduction and algebraic computation. At iteration t, the deductive database (DD) applies geometric theorems, represented as condition–conclusion rules, to the known relations. The resulting geometric facts and their corresponding equations form $\Delta\mathcal{F}_t$. The algebraic system (AS) then solves the accumulated equations using Gaussian elimination and symbolic substitution, producing new variable assignments, geometric measurements, or equalities between quantities, denoted by $\Delta\mathcal{V}_t$. After each iteration, both types of newly derived information are incorporated into the state:

$$S_{t+1} = S_t \cup \Delta\mathcal{F}_t \cup \Delta\mathcal{V}_t.$$

Because algebraic results may satisfy the premises of additional geometric rules, DD and AS alternate to derive further information. The solver returns SOLVED with a numerical value when g is uniquely determined, UNDETERMINED when neither module can derive further information while g remains unresolved, and TIMEOUT when the allotted time is exhausted.

5 EXPERIMENTS

5.1 EVALUATION SETTINGS

Formalization Metrics. We evaluate the parsed formal language $(\mathcal{T}, \mathcal{R}, \mathcal{A}, g)$ against human-verified annotations. Predictions and references are converted into fact sets $\widehat{\mathcal{F}}_i$ and $\mathcal{F}_i$ after deduplication, canonicalization of equivalent expressions, arithmetic normalization, and alignment of point

Table 2: Main results of MLLMs, geometry-specialized models, and our method on GeoFlip pairs. All metrics are reported as percentages. GT denotes human-verified formalizations of GP and GP^*. The best and second-best results in each type are in **bold** and underlined.

Models	Source GP			Counterpart GP*			PSR ↑	ARR ↓
	$\mathrm{EFR}_{\mathrm{GP}}$	$\mathrm{FailR}_{\mathrm{GP}}$	$\mathrm{SuccR}_{\mathrm{GP}}$	$\mathrm{EFR}_{\mathrm{GP}^*}$	$\mathrm{FailR}_{\mathrm{GP}^*}$	$\mathrm{SuccR}_{\mathrm{GP}^*}$		
Closed-source MLLMs								
GPT-5.5	0.0	13.3	**86.7**	0.0	58.8	**41.2**	**35.1**	26.7
GPT-4o	1.0	55.0	44.0	3.4	72.2	24.4	11.6	**22.1**
Claude Opus 5	0.0	18.9	81.1	0.0	99.1	0.9	0.3	41.1
Claude Sonnet 4.6	0.0	26.1	73.9	0.2	94.9	4.9	2.2	27.3
Claude Haiku 4.5	0.2	32.5	67.3	0.0	98.3	1.7	1.2	29.1
Gemini 3.8 Flash	8.3	6.8	84.8	59.8	38.2	2.0	1.9	28.4
Open-weight MLLMs								
GLM-5.3-Flash	0.2	23.3	76.5	3.1	93.4	3.6	2.4	28.3
InternVL3.5-241B-A28B	23.2	21.6	55.2	48.7	48.6	2.7	0.5	19.1
Kimi K3	2.0	23.2	74.8	5.3	89.8	**4.9**	**3.6**	25.4
DeepSeek-V4-Flash-Vision-Exp	3.7	24.5	71.7	25.2	70.5	4.3	3.2	24.2
Qwen3-VL-235B-A22B-Thinking	0.0	23.9	76.1	0.0	99.3	0.7	0.7	28.1
Qwen3-VL-235B-A22B-Instruct	7.5	23.3	69.2	15.7	80.9	3.4	2.0	24.5
Qwen3.8-Flash	18.7	11.9	69.3	61.2	38.3	0.5	0.5	**18.2**
Qwen3.8-Max	8.2	9.0	**82.8**	49.1	48.4	2.6	2.0	27.9
Geometry-specialized Models								
Math-PUMA (Zhuang et al., 2025)	0.0	79.9	20.1	0.0	96.1	3.9	1.0	30.2
G-LLaVA (Gao et al., 2025)	0.7	94.4	4.9	0.5	97.3	2.2	0.0	35.4
DFE-GPS (Zhang et al., 2025)	1.0	91.3	7.7	1.2	97.6	1.2	0.2	7.3
GeoX (Xia et al., 2025)	12.3	35.3	52.5	29.1	70.9	0.0	0.0	30.7
Ours (GPT-5.5)	10.4	9.0	80.6	15.7	18.4	65.9	57.2	11.9
Ours (Claude Opus 5)	7.8	4.9	87.2	13.3	22.8	63.9	59.5	16.5
Ours (GT)	4.8	1.0	**94.2**	9.0	0.0	**91.0**	**86.4**	**0.0**

identifiers by geometric role. We report dataset-level micro precision, recall, and F1:

$$P = \frac{\sum_i |\widehat{\mathcal{F}}_i \cap \mathcal{F}_i|}{\sum_i |\widehat{\mathcal{F}}_i|}, \qquad R = \frac{\sum_i |\widehat{\mathcal{F}}_i \cap \mathcal{F}_i|}{\sum_i |\mathcal{F}_i|}, \qquad F1 = \frac{2PR}{P+R}. \tag{1}$$

Evaluation Metrics. We classify each MLLM's response into one of three types, $\tau \in \{\mathrm{NV}, \mathrm{US}, \mathrm{EF}\}$: NV denotes a finite numerical value, US a textual statement of underdetermination, and EF an extraction failure due to failure to provide a `\boxed{}`. We use `math-verify` (Kydlíček, 2025) to parse numerical answers and compare them with the ground truth. For pair i, let τ_i and τ_i^* denote the source and counterpart output types, A_i and A_i^* their numerical values extracted from the final boxes when the type is NV, and g_i the source ground truth. For a source GP, an NV answer is correct when $|A_i - g_i| < 10^{-3}$ (denoted $A_i = g_i$). For a counterpart GP^*, an answer is correct only if it is classified as US. Accordingly, we report three mutually exclusive rates: **Extraction Failure Rate (EFR)**, **Failure Rate (FailR)**, and **Success Rate (SuccR)**, using subscripts GP and GP^* for source and counterpart problems. For our symbolic pipeline, the solver state directly determines the outcome categories. On GP, SOLVED with the correct value yields $\mathbf{SuccR_{GP}}$, while a SOLVED state with an incorrect value or an UNDETERMINED state yields $\mathbf{FailR_{GP}}$. On underdetermined GP^*, UNDETERMINED yields $\mathbf{SuccR_{GP^*}}$, while SOLVED yields $\mathbf{FailR_{GP^*}}$. TIMEOUT contributes to $\mathbf{EFR}$ for both problems.

$$\begin{aligned} \mathrm{EFR}_{\mathrm{GP}} &= \mathbb{E}_i \mathbf{1}[\tau_i = \mathrm{EF}] & \mathrm{EFR}_{\mathrm{GP}^*} &= \mathbb{E}_i \mathbf{1}[\tau_i^* = \mathrm{EF}] \\ \mathrm{FailR}_{\mathrm{GP}} &= \mathbb{E}_i \mathbf{1}[\tau_i = \mathrm{US} \vee (\tau_i = \mathrm{NV} \wedge A_i \neq g_i)] & \mathrm{FailR}_{\mathrm{GP}^*} &= \mathbb{E}_i \mathbf{1}[\tau_i^* = \mathrm{NV}] \\ \mathrm{SuccR}_{\mathrm{GP}} &= \mathbb{E}_i \mathbf{1}[\tau_i = \mathrm{NV} \wedge A_i = g_i] & \mathrm{SuccR}_{\mathrm{GP}^*} &= \mathbb{E}_i \mathbf{1}[\tau_i^* = \mathrm{US}] \end{aligned} \tag{2}$$

We also report **Pair Success Rate (PSR)** and **Answer Repetition Rate (ARR)**:

$$\mathrm{PSR} = \mathbb{E}_i \mathbf{1}[\tau_i = \mathrm{NV} \wedge A_i = g_i \wedge \tau_i^* = \mathrm{US}], \quad \mathrm{ARR} = \mathbb{E}_i \mathbf{1}[\tau_i = \tau_i^* = \mathrm{NV} \wedge A_i = A_i^*]. \tag{3}$$

PSR measures reliable geometry reasoning across both problems, requiring a correct answer on GP and correct recognition that GP* is underdetermined. **ARR** indicates a likely failure to recognize the removal, suggesting that the model may repeat its source-side reasoning on GP^*.

Prompting and Decoding. During evaluation, a multimodal reasoner receives only the image–question pair, either (I, Q) or (I^*, Q^*). Closed-source and open-weight MLLMs use the same prompt reproduced in Appendix E.1. It requests step-by-step reasoning and a final `\boxed{}` containing a numerical value or a concise statement of insufficient information. We use greedy decoding with temperature zero, with a maximum generation length of 4,096 tokens. Geometry-specialized models follow the input format, preprocessing, and decoding procedure of their original papers or released implementations.

5.2 Main Results

Table 2 reports the comparative results on GeoFlip pairs. Our key findings are as follows:

Gap Between Problem Solving and Underdetermination Recognition. Claude Opus 5 achieves 81.1% success on GP but only 0.9% on GP^*, a gap of 80.2 percentage points. Even GPT-5.5, the strongest baseline with 86.7% accuracy on GP, achieves only 41.2% success on GP^*. The listed open-weight and geometry-specialized models also struggle with underdetermination recognition, achieving at most 4.9% and 3.9% success on GP^*, respectively. PSR shows a similar pattern, reaching only 0.3% for Claude Opus 5 and 35.1% for GPT-5.5. Thus, solving a complete geometry problem does not ensure recognition of underdetermination after necessary evidence is removed.

Widespread "Memorization" Across MLLMs. Many models continue to return the same numerical answer after the key evidence has been removed. Claude Opus 5 repeats its source answer on 41.1% of pairs. This behavior is also prominent in geometry-specialized models, with G-LLaVA reaching an ARR of 35.4%. Our pipeline reduces ARR to 16.5% with Claude Opus 5 and 11.9% with GPT-5.5, and to 0% with GT formalizations. This residual repetition arises from formalization errors: some parses restore a deleted condition, which can make the solver reproduce the source answer even when most of the formalization is correct.

Strong Performance on GeoFlip. Compared with direct answering, our method with Claude Opus 5 raises $\mathrm{SuccR}_{\mathrm{GP}}$ from 81.1% to 87.2% and $\mathrm{SuccR}_{\mathrm{GP}^*}$ from 0.9% to 63.9%. With GPT-5.5, although $\mathrm{SuccR}_{\mathrm{GP}}$ decreases from 86.7% to 80.6%, $\mathrm{SuccR}_{\mathrm{GP}^*}$ rises from 41.2% to 65.9%. Similarly, our method raises PSR to 59.5% with Claude Opus 5 and 57.2% with GPT-5.5, gains of 59.2 and 22.1 percentage points, respectively. With GT formalizations, PSR further reaches 86.4% with no answer repetition. This highlights the importance of faithful formalization for GP and GP^*.

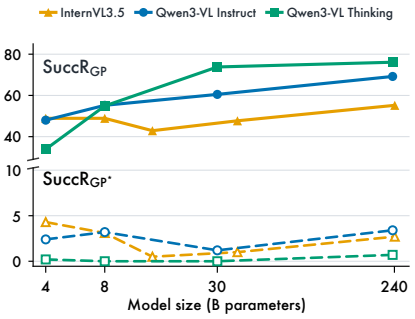


Figure 4: Effects of model scaling and thinking mode on GeoFlip pairs.

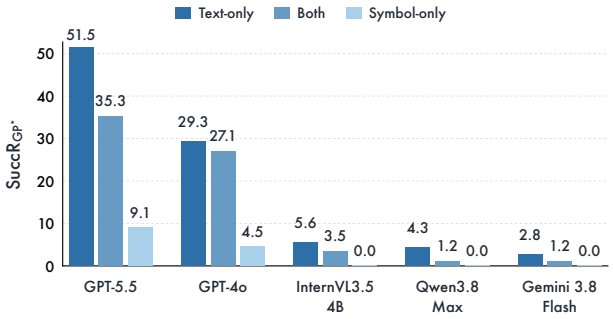


Figure 5: Underdetermination recognition on GeoFlip counterparts by the type of visual evidence removed.

5.3 Behavior Analysis

Effectiveness of Model Scaling and Thinking Mode. To examine their impact, we additionally evaluate Qwen3-VL-4B/8B/30B-A3B, including Instruct and Thinking variants, and InternVL3.5-4B/8B/14B/38B. As Figure 4 demonstrates, $\mathrm{SuccR}_{\mathrm{GP}}$ increases steadily with size for both Qwen3-

VL Instruct and Thinking, while InternVL3.5 shows a smaller overall gain with fluctuations. By contrast, $\mathrm{SuccR}_{\mathrm{GP}^*}$ remains below 5% with no consistent improvement as models grow. Notably, Qwen3-VL Thinking achieves lower $\mathrm{SuccR}_{\mathrm{GP}^*}$ than Instruct at every matched model size, remaining below 1% throughout. Thus, scaling and thinking mode can improve problem-solving performance but do not guarantee better underdetermination recognition.

Underdetermination Recognition across Visual Evidence Types. We compare this capability across counterparts with text-only, symbol-only, or combined removals of visual evidence. Across the five models in Figure 5, $\mathrm{SuccR}_{\mathrm{GP}^*}$ is highest for text-only removals, intermediate when both types are removed, and lowest for symbol-only removals. The gap is largest for GPT-5.5, whose $\mathrm{SuccR}_{\mathrm{GP}^*}$ is 51.5% for text-only removals versus 9.1% for symbol-only removals, a gap of 42.4 percentage points. These results indicate that recognizing underdetermination after removing geometric symbols is more challenging than after removing textual annotations.

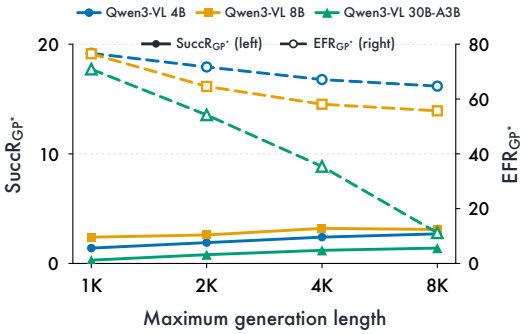


Figure 6: $\mathrm{SuccR}_{\mathrm{GP}^*}$ and $\mathrm{EFR}_{\mathrm{GP}^*}$ as the maximum generation length increases.

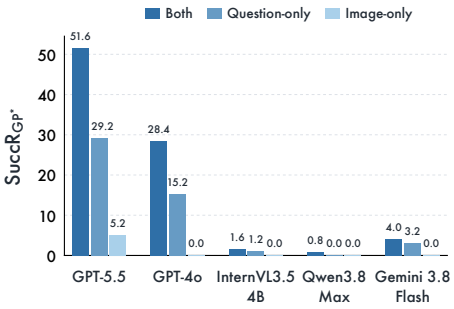


Figure 7: $\mathrm{SuccR}_{\mathrm{GP}^*}$ under question-only, image-only, and joint evidence removal.

Underdetermination Recognition across Removal Modalities. We randomly sample 250 source problems and use the same KIR pipeline (Section 3.4) to generate three variants for each problem by removing evidence from the question, the image, or both. As Figure 7 illustrates, all models achieve the highest $\mathrm{SuccR}_{\mathrm{GP}^*}$ under joint removal and the lowest under image-only removal. This pattern is most pronounced for GPT-5.5, whose SuccR_{GP^*} decreases from 51.6% under joint removal to 29.2% under question-only removal and further to 5.2% under image-only removal. These results highlight the difficulty of recognizing underdetermination caused by missing visual evidence.

Effectiveness of Longer Reasoning. To assess whether a larger output budget improves underdetermination recognition, we vary the maximum output length from 1,024 to 8,192 tokens for three Qwen3-VL Instruct models. Figure 6 indicates that $\mathrm{EFR}_{\mathrm{GP}^*}$ decreases substantially as the maximum generation length increases, while $\mathrm{SuccR}_{\mathrm{GP}^*}$ rises only slightly. Larger output budgets therefore reduce extraction failures but bring only limited gains in underdetermination recognition.

Further analyses of model rethinking and few-shot demonstrations are provided in Appendix D.

6 Conclusion

Reliable multimodal mathematical reasoning requires not only solving problems when sufficient evidence is available, but also recognizing when removing necessary evidence makes them underdetermined. To study these abilities, we propose Key-Information Removal (KIR) and construct GeoFlip with 587 human-verified solvable–underdetermined geometry problem pairs. Evaluations of MLLMs and geometry-specialized models reveal a gap between problem solving and underdetermination recognition, as well as widespread "memorization". Further analyses show that this capability is affected by the type of removed evidence and removal modality, while scaling and reasoning modes do not consistently improve recognition. Our unified pipeline combines multimodal formalization with symbolic reasoning, improving counterpart success and pair success over direct answering. GeoFlip enables evaluation beyond accuracy on solvable problems and supports systematic study of visual evidence awareness in multimodal mathematical reasoning.

AI USE STATEMENT

In this work, we used generative AI tools for GeoFlip synthesis. GPT-5.5 proposed evidence removals, audited candidate problems, and revised question text, while GPT Image 2 edited geometry diagrams. We have not used generative AI tools to develop the research framework or hypotheses, design the methodology or experiments, or implement the core reasoning method. Developing new theoretical models and proving new theorems are not applicable to this work. Additionally, we used generative AI tools to refine testing and evaluation code, prepare scientific figures and tables, and assist with language polishing and reference checks. We have manually reviewed all AI-assisted work without using AI for the review. In particular, three annotators independently reviewed every retained problem pair for solvability, evidence removal, and visual quality, with corrections rechecked until unanimous approval. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

We discuss potential ethical concerns as follows: (1) Data provenance and intellectual property. GeoFlip is derived from the Geometry3K test split included in PGPS9K. We cite the original datasets and will distribute the derived images and annotations in accordance with the licenses and terms governing the source data. (2) Human verification and privacy. Three annotators assess only the mathematical solvability, evidence removal, and visual quality of each edited problem. The released benchmark contains no personal, sensitive, or user-interaction data. (3) Intended use. GeoFlip is designed to evaluate whether multimodal models can solve complete geometry problems and recognize when a problem becomes underdetermined after necessary evidence is removed. It is not intended for high-stakes decision-making or as a comprehensive ranking of models' general mathematical capabilities. (4) Potential limitations and risk control. The benchmark may inherit limitations in topic coverage and problem structure from its source dataset, while AI-assisted image editing may introduce artifacts or unintended cues. To mitigate these risks, every edited problem is independently reviewed by three annotators and retained only with unanimous approval. Nevertheless, users should consider the benchmark's scope and inspect the released data before adapting it to other settings. (5) AI assistance. AI-assisted steps are disclosed in the AI Use Statement above.

REPRODUCIBILITY STATEMENT

We document the benchmark construction criteria, source data, KIR pipeline, automatic auditing, and human-verification procedure in Section 3. Section 4 specifies the formal problem representation and symbolic reasoning procedure, while Section 5.1 describes the evaluated models, prompting protocols, decoding settings, baselines, and metrics. The supplementary material provides the complete prompts, annotation taxonomy, human-verification guidelines, implementation details, and additional experimental results. We will release the 587 GeoFlip pairs, removal annotations and rationales, human-verified formal languages, model responses, evaluation scripts, and source code through an anonymous repository to support reproduction of the reported results.

REFERENCES

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 Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 3313–3323, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.218. URL https://aclanthology.org/2022.emnlp-main.218/.

Chenrui Fan, Ming Li, Lichao Sun, and Tianyi Zhou. Missing premise exacerbates overthinking: Are reasoning models losing critical thinking skill? In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=ufozo2Wc9e.

Jiahui Gao, Renjie Pi, Jipeng Zhang, Jiacheng Ye, Wanjun Zhong, Yufei Wang, Lanqing HONG, Jianhua Han, Hang Xu, Zhenguo Li, and Lingpeng Kong. G-LLaVA: Solving geometric problem with multi-modal large language model. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 3490–3511, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/09afabe33dc7db66530dea6483b22e5d-Paper-Conference.pdf.

Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. HallusionBench: An advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14375–14385, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Guan_HallusionBench_An_Advanced_Diagnostic_Suite_for_Entangled_Language_Hallucination_and_CVPR_2024_paper.html.

Daya Guo, Dejian Yang, Haowei Zhang, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645:633–638, 2025. doi: 10.1038/s41586-025-09422-z. URL https://www.nature.com/articles/s41586-025-09422-z.

Xingwei He, Qianru Zhang, A-Long Jin, Yuan Yuan, and Siu-Ming Yiu. TUBench: Benchmarking large vision-language models on trustworthiness with unanswerable questions, 2024. URL https://arxiv.org/abs/2410.04107.

Kaixuan Huang, Jiacheng Guo, Zihao Li, Xiang Ji, Jiawei Ge, Wenzhe Li, Yingqing Guo, Tianle Cai, Hui Yuan, Runzhe Wang, Yue Wu, Ming Yin, Shange Tang, Yangsibo Huang, Chi Jin, Xinyun Chen, Chiyuan Zhang, and Mengdi Wang. MATH-Perturb: Benchmarking LLMs' math reasoning abilities against hard perturbations, 2025. URL https://arxiv.org/abs/2502.06453.

Hynek Kydlíček. Math-Verify: Math verification library. https://github.com/huggingface/Math-Verify, 2025. Software.

Qintong Li, Leyang Cui, Xueliang Zhao, Lingpeng Kong, and Wei Bi. GSM-Plus: A comprehensive benchmark for evaluating the robustness of LLMs as mathematical problem solvers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2961–2984, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.163. URL https://aclanthology.org/2024.acl-long.163/.

Zhaoyu Li, Hangrui Bi, Jialiang Sun, Zenan Li, Kaiyu Yang, and Xujie Si. PyEuclid: A versatile formal plane geometry system in Python. In *Computer Aided Verification*, pp. 405–420. Springer Nature Switzerland, 2025. doi: 10.1007/978-3-031-98685-7_20.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916. Curran Associates, Inc., 2023. doi: 10.52202/075280-1516. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/6dcf277ea32ce3288914faf369fe6de0-Paper-Conference.pdf.

Yi Liu, Xiangyu Liu, Zequn Sun, and Wei Hu. Answering the unanswerable is to err knowingly: Analyzing and mitigating abstention failures in large reasoning models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(38):32231–32239, 2026. doi: 10.1609/aaai.v40i38.40496. URL https://ojs.aaai.org/index.php/AAAI/article/view/40496.

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 symbolic reasoning. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 6774–6786, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.528. URL https://aclanthology.org/2021.acl-long.528/.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations*, 2024. URL https://web.cs.ucla.edu/~kwchang/bibliography/lu2024mathvista/.

Nishanth Madhusudhan, Vikas Yadav, and Alexandre Lacoste. Knowing when not to answer: Evaluating abstention in multimodal reasoning systems, 2026. URL https://arxiv.org/abs/2604.14799.

Fanqing Meng, Lingxiao Du, Zongkai Liu, Zhixiang Zhou, Quanfeng Lu, Daocheng Fu, Tiancheng Han, Botian Shi, Wenhai Wang, Junjun He, Kaipeng Zhang, Ping Luo, Yu Qiao, Qiaosheng Zhang, and Wenqi Shao. MM-Eureka: Exploring the frontiers of multimodal reasoning with rule-based reinforcement learning, 2025. URL https://arxiv.org/abs/2503.07365.

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, 2024. URL https://arxiv.org/abs/2410.05229.

Maizhen Ning, Zihao Zhou, Qiufeng Wang, Xiaowei Huang, and Kaizhu Huang. GNS: Solving plane geometry problems by neural-symbolic reasoning with multi-modal LLMs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(23):24957–24965, Apr. 2025. doi: 10.1609/aaai.v39i23.34679. URL https://ojs.aaai.org/index.php/AAAI/article/view/34679.

OpenAI. GPT-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

OpenAI. GPT-5.5 model, 2026a. URL https://developers.openai.com/api/docs/models/gpt-5.5. Accessed: 2026-08-09.

OpenAI. GPT Image 2 model, 2026b. URL https://developers.openai.com/api/docs/models/gpt-image-2. Accessed: 2026-08-09.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Saurabh Srivastava, Annarose M B, Anto P V, Shashank Menon, Ajay Sukumar, Adwaith Samod T, Alan Philipose, Stevin Prince, and Sooraj Thomas. Functional benchmarks for robust evaluation of reasoning performance, and the reasoning gap, 2024. URL https://arxiv.org/abs/2402.19450.

Katherine Tian, Eric Mitchell, Huaxiu Yao, Christopher D. Manning, and Chelsea Finn. Fine-tuning language models for factuality. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/c361ae924c23cafca6033610d25dbc65-Abstract-Conference.html.

Renqiu Xia, Mingsheng Li, Hancheng Ye, Wenjie Wu, Hongbin Zhou, Jiakang Yuan, Tianshuo Peng, Xinyu Cai, Xiangchao Yan, Bin Wang, Conghui He, Botian Shi, Tao Chen, Junchi Yan, and Bo Zhang. GeoX: Geometric problem solving through unified formalized vision-language pre-training. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 15123–15141, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/2722a0ccf6acfe3d144fdbb0dedd80b5-Paper-Conference.pdf.

Boyang Xue, Qi Zhu, Rui Wang, Sheng Wang, Hongru Wang, Minda Hu, Fei Mi, Yasheng Wang, Lifeng Shang, Qun Liu, and Kam-Fai Wong. ReliableMath: Benchmark of reliable mathematical reasoning on large language models, 2025. URL https://arxiv.org/abs/2507.03133.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T. Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. MetaMath: Bootstrap your own mathematical questions for large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=N8N0hgNDRt.

Ming-Liang Zhang, Fei Yin, Yi-Han Hao, and Cheng-Lin Liu. Plane geometry diagram parsing. In Lud De Raedt (ed.), *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pp. 1636–1643. International Joint Conferences on Artificial Intelligence Organization, 7 2022. doi: 10.24963/ijcai.2022/228. URL https://doi.org/10.24963/ijcai.2022/228. Main Track.

Ming-Liang Zhang, Fei Yin, and Cheng-Lin Liu. A multi-modal neural geometric solver with textual clauses parsed from diagram. In Edith Elkind (ed.), *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, pp. 3374–3382. International Joint Conferences on Artificial Intelligence Organization, 8 2023. doi: 10.24963/ijcai.2023/376. URL https://doi.org/10.24963/ijcai.2023/376. Main Track.

Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Yu Qiao, Peng Gao, and Hongsheng Li. MathVerse: Does your multi-modal LLM truly see the diagrams in visual math problems? In *European Conference on Computer Vision*, 2024. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/papers/01270.pdf.

Zeren Zhang, Jo-Ku Cheng, Jingyang Deng, Lu Tian, Jinwen Ma, Ziran Qin, Xiaokai Zhang, Na Zhu, and Tuo Leng. Diagram formalization enhanced multi-modal geometry problem solver. In *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2025. doi: 10.1109/ICASSP49660.2025.10889286.

Yilun Zhao, Guo Gan, Chengye Wang, Chen Zhao, and Arman Cohan. Are multimodal LLMs robust against adversarial perturbations? RoMMath: A systematic evaluation on multimodal math reasoning. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 11653–11665, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.naacl-long.582. URL https://aclanthology.org/2025.naacl-long.582/.

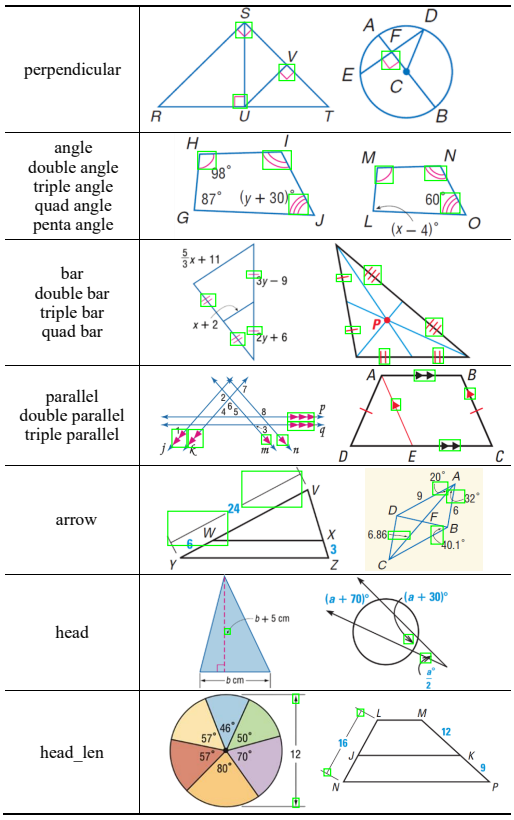
Zihao Zhou, Shudong Liu, Maizhen Ning, Wei Liu, Jindong Wang, Derek F. Wong, Xiaowei Huang, Qiufeng Wang, and Kaizhu Huang. Is your model really a good math reasoner? evaluating mathematical reasoning with checklist, 2024. URL https://arxiv.org/abs/2407.08733.

Wenwen Zhuang, Xin Huang, Xiantao Zhang, and Jin Zeng. Math-PUMA: Progressive upward multimodal alignment to enhance mathematical reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 26183–26191, 2025. doi: 10.1609/aaai.v39i24.34815. URL https://ojs.aaai.org/index.php/AAAI/article/view/34815.

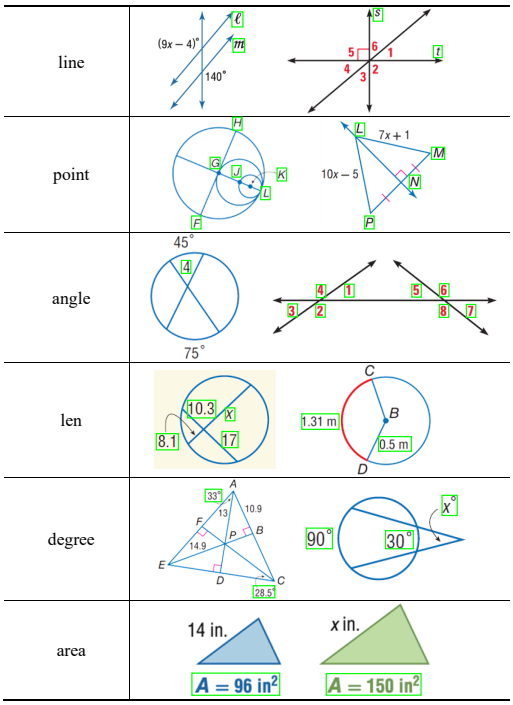
A VISUAL EVIDENCE TAXONOMY

Our construction uses the visual evidence annotations provided in PGPS9K for the Geometry3K problems (Zhang et al., 2023). These annotations follow the complete *text* and *symbol* taxonomies of PGDP (Zhang et al., 2022). Text annotations identify geometric objects and quantities, such as point and line labels, variables, lengths, angle measures, and areas. Symbol annotations represent geometric relations and measurement conventions, including perpendicularity, equal lengths or angles, parallelism, and length indicators. Each annotation record stores its category and localized image region, together with the visible content for text annotations. Figure 8 gives the complete taxonomy, while Figure 1 shows the compact version used in the main text.

KIR uses the same two categories for construction and pair-level statistics. Each removed notation is a visually complete item rather than an isolated glyph. For example, an angle measurement includes its degree sign, and a dimension label includes its leader when the leader would otherwise reveal the deleted measurement role.



(a) Symbol annotations



(b) Text annotations

Figure 8: Complete symbol and text annotation taxonomy from the appendix of Zhang et al. (2022).

B KIR CONSTRUCTION PROMPTS

We use GPT-5.5 (high) (OpenAI, 2026a) for KIR proposal, problem audit, and problem repair, and GPT Image 2 (OpenAI, 2026b) for image editing and visual repairs. Tables 3–6 summarize the prompts for the four construction stages. The audit judges the current rendered image and question jointly, rather than relying on the edit instruction or removal rationale alone. The source SP identifies how the removed evidence supports the original derivation, while counterpart solvability is judged solely from (I^*, Q^*).

Role and inputs. Given a source image I, question Q, solution program SP, reference answer, and visual localizations $\mathcal{L}$, propose at most one underdetermined counterpart.
Suitability and skip decision. Determine whether the source has at least one localized visual-evidence item used by its expression or solving chain. Skip if no such item exists, the source is invalid, or no local deletion can preserve a coherent diagram and a well-defined target.
Evidence selection. Trace the source solution to enumerate the localized visual evidence it uses, then randomly select one item from this set for removal. Select a complete visible value, algebraic label, or relation mark. A label or local construction unit is eligible only if its removal preserves the identity and grounding of the requested target.
Recovery-path check. Enumerate alternative solution routes supported by the remaining visual or textual evidence and record them in the removal rationale. Record surviving or uncertain routes for the subsequent automatic audit. Disrupting the supplied solution alone is not evidence of underdetermination.
Textual recovery paths. Preserve the source question unless it states or logically reconstructs the selected visual evidence or independently supports another solution route. In either case, remove or rewrite only the relevant condition while preserving the target and unrelated text.
Output. Return a status tag of either `skipped` or `candidate`. For `skipped`, provide a concise reason. For `candidate`, return JSON containing the source identifier, revised question, `removal_rationale`, `removed_info` with its localization and reason, and a grounded `image_edit_instruction`. Proposal status does not imply acceptance.

Table 3: Step 1: KIR proposal prompt. Each candidate proceeds to image editing and automatic auditing.

Role and inputs. Given the source geometry diagram and KIR instruction, edit the specified visual evidence in the source image.
Local removal. Erase exactly the requested localized visual evidence at the provided source localization. Remove a complete measurement token, including its unit or degree sign, and remove an associated leader only when it belongs to the same visual item.
Preservation. Keep all unrelated geometry lines, circles, points, labels, values, marks, colors, image dimensions, and textbook style unchanged. Fill the erased region naturally with the background. Do not add replacement text, hints, blur, boxes, shadows, explanations, or new geometry.
Output. Return one edited image. If the requested local edit cannot be completed faithfully, report failure to the audit–repair loop. Do not substitute a newly drawn diagram or declare the candidate accepted.

Table 4: Step 2: GPT Image 2 prompt for localized image editing. The same preservation rules apply to visual repairs.

Role and inputs. Audit one KIR candidate. Inspect the source (I, Q, SP), reference answer, localizations, current (I^*, Q^*), removal rationale, cumulative edit instruction, and any preceding audit feedback. Judge the current rendered pair, not the intended edit alone.
Question-quality check. Inspect Q^* for conditions that restate or reconstruct the removed information, or support a surviving solution route together with the remaining image evidence. Identify the textual condition to remove for each such route. Verify that the question remains coherent, its target remains well defined and unchanged, and unrelated text is preserved.
Image-quality check. Inspect I^* for remaining values, labels, relation marks, or construction clues that reconstruct the removed information or support a surviving solution route together with the question. Identify the visual condition to remove for each such route. A faithful edit can still fail this check if other visual evidence keeps the target solvable. Verify complete erasure of the targeted evidence, including units or degree signs. Also check that unrelated geometry and style are preserved, the image looks natural, and readable annotations remain attached to the correct objects.
Joint validity and underdetermination. Assess both modalities together, checking source validity, removal relevance, and the consistency of the retained evidence. Use source-only information to check the edit, never as a retained premise. Do not infer exact measurements from apparent drawing scale. Explain why the remaining conditions are consistent but permit different values of the same target, for example by identifying a free parameter or two admissible configurations. A broken source program, an unsuccessful solution attempt, contradictory conditions, or a missing target referent is not sufficient. If a solution survives, give its derivation and assign repair feedback to the modality containing the condition to remove. Flag both checks when both modalities require changes. An uncertain joint judgment cannot authorize acceptance.
Output. Return separate `question_quality` and `image_quality` decisions. For a question-quality failure, provide feedback identifying the textual condition to remove. For an image-quality failure, provide a localized EI identifying the visual item and region to edit. Set `decision=accept` only when both checks pass on this exact pair. Otherwise, set `decision=repair`. The controller flags a failed candidate for human review when the repair budget is exhausted.

Table 5: Step 3: problem-audit prompt. One audit produces two quality decisions and feedback for targeted repair. An automatic pass remains subject to final human verification.

Step 4: Question and rationale repair.
Role and inputs. Repair the textual component of a failed KIR candidate using (Q, Q^*), the current RR, and the question-quality feedback. Preserve the source identifier and requested target.
Repair. Remove only textual conditions that reconstruct deleted visual evidence or support a surviving solution route. Restore unrelated question content if it was changed unintentionally. Update RR to match the revised question. Do not introduce new premises, reveal the source answer, state that the problem is unsolvable in the question, or remove the target's referent.
Output. Return the revised Q^*, RR, and a list of textual changes. Do not modify I^* or the image-edit instructions. If no permitted repair is available, report the unresolved question-quality failure.

Table 6: Step 4: question and rationale repair prompt. Image repair directly uses the EI produced by problem audit and follows the Step 2 image-editing prompt.

C Additional Construction Details

C.1 Derivations of the Remaining Solution Routes

After the visual condition $XZ = 3y + 7$ is removed in Figure 2, the source equation $19 = 3y + 7$ is no longer available. However, the edited image–question pair still determines $y = 4$ through two remaining solution routes.

Solution 1: surviving visual evidence. The retained right-angle mark gives $m\angle WXY = 90°$. Together with the retained angle label $m\angle WXY = 20y + 10$, this gives

$$20y + 10 = 90 \implies y = 4. \tag{4}$$

Solution 2: surviving textual condition. The retained premise $\triangle WXY \cong \triangle WXZ$ makes the corresponding angles $\angle WXY$ and $\angle WXZ$ equal. Because Y–X–Z are collinear, these adjacent angles are supplementary and therefore both equal $90°$. The retained angle label then gives

$$\begin{gathered} \triangle WXY \cong \triangle WXZ \Longrightarrow m\angle WXY = m\angle WXZ, \\ m\angle WXY + m\angle WXZ = 180° \Longrightarrow m\angle WXY = 90°, \\ 20y + 10 = 90 \Longrightarrow y = 4. \end{gathered} \tag{5}$$

C.2 Human Verification

Three members of the research team reviewed every source question and proposed counterpart: two undergraduate students and one doctoral student. All three worked on LLM-based geometry reasoning and had completed secondary-school geometry coursework. The annotation was conducted as part of this research collaboration, with no separate annotation payment.

Annotators inspect each constructed GP^* alongside its source (I, Q). They assess each pair against the four construction criteria in Section 3.1. They compare the source and counterpart to check evidence relevance and content preservation. To verify underdetermination, each annotator identifies the role of the removed evidence and searches for duplicate or alternative solution routes. Automatic audit outcomes provide diagnostic feedback but do not determine final inclusion. Every constructed candidate enters human verification, which leads to acceptance without changes or manual correction.

For a constructed pair requiring correction, one annotator applies the following actions:

- **Question-evidence removal.** If Q^* reconstructs the removed visual evidence, remove or rewrite only the reconstructing condition. If Q^* independently supports another solution route, remove or rewrite only the condition supporting that route. In both cases, preserve the target and unrelated text.
- **Visual-evidence removal.** If other visual evidence supports a surviving solution route, update EI to remove that evidence while preserving unrelated geometry and notations.
- **Image regeneration.** If I^* does not faithfully implement EI, such as by leaving the specified evidence partly visible, altering unrelated content, or producing an unnatural edit, regenerate I^* from I using the corrected EI while leaving Q^* and RR unchanged.

Whenever the retained evidence changes, RR and, when relevant, the cumulative EI are updated to match the final pair. One annotator corrects the pair, and the other two review the revised GP^*. Corrections continue until all three annotators approve the final version under all four criteria. Final human verification required corrections to 89 of the 587 retained pairs (15.2%). The remaining 498 pairs (84.8%) required no manual changes.

C.3 Pipeline Statistics

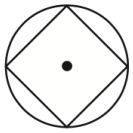
We begin with 589 Geometry3K test problems included in PGPS9K. Among these source problems, two are strategy skips, while 587 candidates enter automatic audit and subsequent human verification. Table 7 reports the number of candidates at each construction stage.

Construction stage	Problems/pairs
Source problems	589
Strategy skip	2
Passed audit, 0 repair	522
Passed audit, 1 repair	53
Passed audit, 2 repairs	8
Passed audit, 3 repairs	4
Candidates entering human review	587

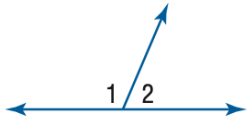
Table 7: **GeoFlip construction-stage statistics.** Repair counts report the number of rounds required before passing the automatic audit.

C.4 Skipped Questions

The two source problems in Figure 9 can be solved from their questions alone. For `prob_3153`, the stated circle area 18π gives $r = 3\sqrt{2}$, so the inscribed square has diagonal $6\sqrt{2}$ and side length 6. For `prob_3390`, a linear pair sums to $180°$, so $m\angle 1 = 180° - 67° = 113°$.



(a) `prob_3153`. A square is inscribed in a circle of area 18π square units. Find the length of a side of the square.



(b) `prob_3390`. $\angle 1$ and $\angle 2$ form a linear pair and $m\angle 2 = 67°$. Find the measure of $\angle 1$.

Figure 9: The two source problems excluded from GeoFlip because their solutions do not use diagram evidence.

D Additional Behavior Analysis

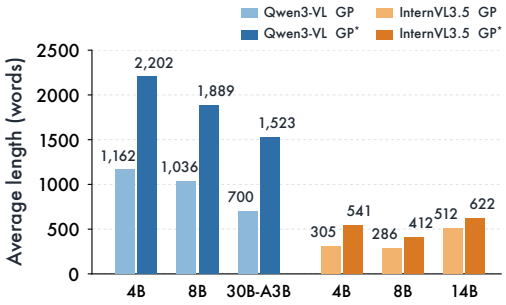


Figure 10: Average length of model responses on GP and GP^*

Analysis of Model Rethinking. We analyze response lengths and keyword distributions when models encounter missing evidence. All six models in Figure 10 produce longer responses on GP^* than on GP. For both model families, the absolute length gap narrows as model size increases. The word cloud in Figure 11 (left) includes "missing," "not given," and "cannot determine," indicating that models acknowledge missing information. The rethinking cloud in Figure 11 (right) prominently features "wait," "check," and "look again," reflecting attempts to check earlier reasoning steps.



Figure 11: Keywords in model responses to underdetermined problems. The left shows expressions related to missing information, while the right word cloud shows expressions related to rethinking.

Effectiveness of Few-shot Demonstrations. We evaluate GPT-5.5 and Claude Opus 5 with $N \in \{2, 4, 6, 8\}$ formalization demonstrations. Table 8 indicates that increasing N from 2 to 4 generally improves parsing F1 on I and I^*. From $N = 4$ to 8, F1 changes little, except for Claude Opus 5 on I^*, where it drops by 5.2 percentage points. Downstream performance follows a similar pattern: increasing N from 2 to 4 improves SuccR_{GP} and SuccR_{GP^*} for both models, with PSR improving or remaining unchanged. Larger N brings no consistent gains across these metrics. We therefore use $N = 4$, which offers the best overall balanced performance.

Table 8: Few-shot formalization and downstream performance (%). We report $N = 4$ in Table 2. The best and second-best results within each model are in **bold** and underlined.

	Source GP			Counterpart GP^*					
N	P	R	F1	P	R	F1	SuccR_{GP}	SuccR_{GP^*}	PSR
GPT-5.5									
2	84.3	78.9	**81.5**	79.6	75.3	77.4	79.6	62.0	53.5
4	83.9	78.0	80.8	80.8	75.1	77.9	**80.6**	**65.9**	**57.2**
6	83.6	78.9	81.2	79.7	76.5	78.1	79.9	61.5	53.5
8	82.4	78.7	80.5	80.5	76.4	**78.4**	80.2	63.9	56.1
Claude Opus 5									
2	84.9	79.2	81.9	80.7	76.6	78.6	86.2	63.7	**59.5**
4	84.7	80.6	**82.6**	81.4	78.1	**79.7**	87.2	**63.9**	**59.5**
6	83.8	81.1	82.4	79.7	79.3	79.5	87.1	63.2	58.4
8	83.6	81.6	**82.6**	75.4	73.7	74.5	**87.7**	58.6	55.7

E Prompts of Evaluation and Formalization

E.1 Evaluation Prompt

Closed-source and open-weight MLLMs use the same prompt for a source problem and its underdetermined counterpart. The system instruction is reproduced verbatim below:

```
You are a careful geometry solver.

Analyze the problem using the diagram and the text.
Explain your reasoning step by step.

At the end of your response, put the final result in
LaTeX boxed format:

\boxed{...}

Rules for the boxed result:
```

```
- Put exactly one final result in the last \boxed{}.
- The boxed content must be either a numerical value or a concise textual statement.
- Do not include reasoning, explanations, or units inside \boxed{}.
```

E.2 Formalization Prompt and Demonstrations

GPT-5.5 and Claude Opus 5 use the same prompt for multimodal formalization. The task instruction is followed by four fixed in-context demonstrations, each containing an image–question pair and its reference Logic Form. The task instruction is reproduced below:

```
You are a geometry formalization engine.

Given a geometry diagram and its question, transcribe all information
explicitly stated in the text or shown in the diagram into one complete
Logic Form. Do not solve the problem or add relations that must be
derived using geometric theorems.

Return exactly one JSON object with the following fields:

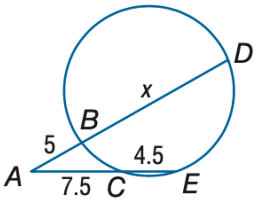
{
  "points": [...],
  "point_positions": [...],
  "lines": [...],
  "circles": [...],
  "relations": [...],
  "equations": [...],
  "goal": {
    "expression": "...",
    "unit": "..."
  }
}

Rules:
- Record every point used in the Logic Form.
- Each line must contain all points on that line in spatial order.
- Each circle must contain its center and all points on its
  circumference.
- Record geometric relations explicitly stated in the question or
  indicated by visual symbols.
- Record all numerical values and algebraic expressions as equations.
- Mark the source of each relation or equation as "text", "diagram",
  or "both".
- The goal must be the quantity requested by the question, not its
  value.
- Do not infer unstated conditions, restore missing information, or
  include intermediate deductions.
- Return the JSON object only, without explanations or proof steps.
```

E.3 Fixed In-context Demonstrations

The question text and reference Logic Form for each of the four fixed demonstrations are reproduced below, in the same order in which they are provided to the formalizer.

Demonstration 1.

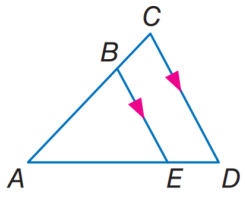


Question: Find x.
Point assignment: Use X for the unlabelled circle centre; lowercase x is the segment value.

Output:


```
{
  "format": "pyeuclid-oriented-logic-form",
  "version": "1.0",
  "topology_complete": true,
  "points": ["A","B","C","D","E","X"],
  "lines": [["A","B","D"],["A","C","E"],["B","C"]],
  "circles": [{"center": "X","points": ["B","D","E","C"]}],
  "relations": [],
  "equations": [
    {"left": "Length(A,B)", "right": "5", "unit": "scalar",
     "source": "diagram"},
    {"left": "Length(A,C)", "right": "7.5", "unit": "scalar",
     "source": "diagram"},
    {"left": "Length(C,E)", "right": "4.5", "unit": "scalar",
     "source": "diagram"},
    {"left": "Length(B,D)", "right": "x", "unit": "scalar",
     "source": "diagram"}
  ],
  "goal": {"expression": "Variable(x)","unit": "scalar"},
  "point_positions": [
    {"point": "A","x": 3,"y": 188}, {"point": "B","x": 59,"y": 154},
    {"point": "C","x": 100,"y": 185}, {"point": "D","x": 222,"y": 60},
    {"point": "E","x": 176,"y": 182}, {"point": "X","x": 134,"y": 97}
  ]
}
```


Demonstration 2.



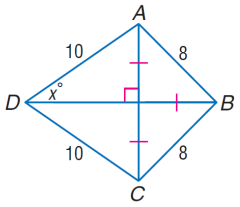
Question: If $AD = 27$, $AB = 8$, and $AE = 12$, find BC.

Output:


```
{
  "format": "pyeuclid-oriented-logic-form",
  "version": "1.0",
  "topology_complete": true,
  "points": ["A","B","C","D","E"],
  "lines": [["A","B","C"],["A","E","D"],["B","E"],["C","D"]],
  "circles": [],
  "relations": [
    {"type": "Parallel", "points": ["B","E","C","D"],
     "source": "diagram"}
  ],
  "equations": [
```

```
    {"left": "Length(A,D)", "right": "27", "unit": "scalar",
     "source": "text"},
    {"left": "Length(A,B)", "right": "8", "unit": "scalar",
     "source": "text"},
    {"left": "Length(A,E)", "right": "12", "unit": "scalar",
     "source": "text"}
  ],
  "goal": {"expression": "Length(B,C)","unit": "scalar"},
  "point_positions": [
    {"point": "A","x": 1,"y": 217}, {"point": "B","x": 152,"y": 59},
    {"point": "C","x": 207,"y": 1}, {"point": "D","x": 323,"y": 218},
    {"point": "E","x": 238,"y": 218}
  ]
}
```


Demonstration 3.

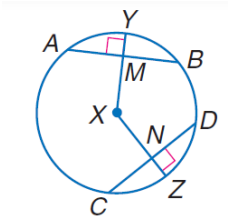


Question: Find $\sin x$.

Output:


```
{
  "format": "pyeuclid-oriented-logic-form",
  "version": "1.0",
  "topology_complete": true,
  "points": ["A","B","C","D","E"],
  "lines": [
    ["A","E","C"], ["D","E","B"], ["A","B"], ["B","C"], ["C","D"],
    ["D","A"]
  ],
  "circles": [],
  "relations": [
    {"type": "Perpendicular", "points": ["A","E","D","E"],
     "source": "diagram"}
  ],
  "equations": [
    {"left": "Length(A,D)", "right": "10", "unit": "scalar",
     "source": "diagram"},
    {"left": "Length(C,D)", "right": "10", "unit": "scalar",
     "source": "diagram"},
    {"left": "Length(A,B)", "right": "8", "unit": "scalar",
     "source": "diagram"},
    {"left": "Length(B,C)", "right": "8", "unit": "scalar",
     "source": "diagram"},
    {"left": "Length(A,E)", "right": "Length(B,E)", "unit": "scalar",
     "source": "diagram"},
    {"left": "Length(B,E)", "right": "Length(C,E)", "unit": "scalar",
     "source": "diagram"},
    {"left": "Angle(A,D,E)", "right": "x", "unit": "degree",
     "source": "diagram"}
  ],
  "goal": {"expression": "sin(Angle(A,D,E))","unit": "scalar"},
  "point_positions": [
    {"point": "A","x": 223,"y": 0}, {"point": "B","x": 377,"y": 155},
    {"point": "C","x": 224,"y": 309}, {"point": "D","x": 0,"y": 155},
    {"point": "E","x": 223,"y": 155}
  ]
}
```

Demonstration 4.



Question: In $\odot X$, $AB = 30$, $CD = 30$, and $m\widehat{CZ} = 40$. Find $m\widehat{AB}$.

Output:


```
{
  "format": "pyeuclid-oriented-logic-form",
  "version": "1.0",
  "topology_complete": true,
  "points": ["A","B","C","D","M","N","X","Y","Z"],
  "lines": [["A","M","B"],["C","N","D"],["Y","M","X"],["X","N","Z"]],
  "circles": [{"center": "X","points": ["Y","B","D","Z","C","A"]}],
  "relations": [
    {"type": "Perpendicular", "points": ["X","Y","A","B"],
     "source": "diagram"},
    {"type": "Perpendicular", "points": ["X","Z","C","D"],
     "source": "diagram"}
  ],
  "equations": [
    {"left": "Length(A,B)", "right": "30", "unit": "scalar",
     "source": "text"},
    {"left": "Length(C,D)", "right": "30", "unit": "scalar",
     "source": "text"},
    {"left": "ArcMeasure(X,C,Z)", "right": "40", "unit": "degree",
     "source": "text"}
  ],
  "goal": {"expression": "ArcMeasure(X,A,B)","unit": "degree"},
  "point_positions": [
    {"point": "A","x": 35,"y": 18}, {"point": "B","x": 144,"y": 32},
    {"point": "C","x": 78,"y": 175}, {"point": "D","x": 176,"y": 100},
    {"point": "M","x": 96,"y": 26}, {"point": "N","x": 128,"y": 137},
    {"point": "X","x": 90,"y": 91}, {"point": "Y","x": 98,"y": 0},
    {"point": "Z","x": 143,"y": 156}
  ]
}
```