

A DATASET

A.1 DATA COLLECTION

We first collect approximately 900k secondary school-level mathematical problems and their corresponding solutions from public websites. Each sample contains at least one image in the solution explanation. These images vary in nature: some consist solely of texts about the problem, others depict mathematical figures such as geometric diagrams or function plots, while a portion are non-informative or irrelevant images. We label these images with Qwen2.5-VL-72B and retain only those questions containing mathematical images and discard the rest. To improve readability and ensure dataset consistency, we leverage GPT-4.1 to extract text from the remaining purely textual images. Subsequently, we task GPT-4.1 to rewrite each original sample into two standardized markdown files, representing the question and solution separately. In the final step, GPT-4.1 verifies these files to identify and remove incomplete or incoherent samples. Furthermore, to improve the accessibility of our dataset, we use GPT-4.1 to translate the samples into English, resulting

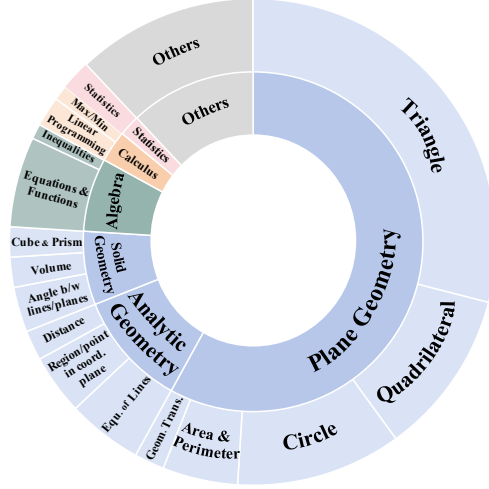


Figure 1: **Distribution of Knowledge Types in Math-VR Dataset.** Geometry constitutes the majority of problems (76%), with Algebra and Calculus comprising 12%.

A.2 DATASET CATEGORIZATION

In Math-VR dataset, approximately 29% of the questions are expressed using text only, while the remaining 71% are demonstrated with both text and mathematical images. Both types of questions require visual mathematical reasoning. In addition to organizing our dataset based on the modality of the questions, we can also classify them according to different knowledge types. First, we randomly sample several thousand question-solution pairs and input them into an VLM (here, GPT-4.1), which generates initial labels including a root knowledge type, a sub-knowledge type and a primary knowledge point, for each pair. Using these labels alongside formal definitions of mathematical concepts, we construct a hierarchical knowledge tree. This taxonomy then serves as a framework to categorize all 90k unique samples in our dataset. Based on this categorization, the dataset is divided into four main knowledge domains: Geometry, Algebra, Calculus, and Statistics. Geometry constitutes the largest portion, 81% of all samples. Within Geometry, there are three primary subcategories: Plane Geometry, Solid Geometry, and Analytic Geometry. These subcategories are further broken down into specific knowledge points such as Triangle, Circle, Quadrilateral, and Area & perimeter calculation. This hierarchical categorization is visually summarized in the pie chart shown in Figure 1.

Table 1: **Key Statistics for Math-VR Dataset**

Statistics	Number
Total Unique Samples	89,075
- Training	86,575
- Testing	2500
- Text questions	29%
- Multimodal questions	71%
- Single-part questions	51%
- Multiple-choice	40%
- Answer-based	55%
- Proof-based	5%
- Multi-part questions	49%
- Multiple-choice	3%
- Answer-based	72%
- Proof-based	25%
- Two sub-questions	51%
- Three sub-questions	40%
- Four sub-questions	7%
Question length (text tokens)	
- Average	131.64
Solution length (text tokens)	
- Average	510.76
Multimodal Question Image	
- Average number	1.05
- Average resolution	208x139
Solution Image	
- Average number	1.15
- Average resolution	199x151

A.3 DATASET STATISTICS

We summarize the statistics of our dataset in Table 1. The dataset comprises approximately 90k unique questions covering a wide range of types, including both single-part and multipart questions, with two- and three-part questions being the most common. It also includes multiple-choice, answer-based, and proof-based questions, among which answer-based questions are the most prevalent. On average, each solution contains at least one image, reflecting the importance of using images to support the reasoning process.

B BENCHMARK

B.1 TEMPLATES USED IN VLM-BASED EVALUATION PIPELINE

We use GPT-4.1 as our VLM evaluation tool. Figure 2 presents the template we use to prompt the VLM for generating the meta information of each sample. Figure 3 displays the template used by the VLM to evaluate the model response to each question.

B.2 MANUAL SAMPLE SELECTION AND META INFORMATION VERIFICATION

We combine the selection of samples that require non-trivial visual reasoning with the verification of meta information into a single manual review process. Figure 4 shows the interface used for this review. The interface displays the question, solution, and meta information all within one view. Annotators are tasked with flagging the unqualified samples. For the remaining samples, they verify that the extracted answers match the final answers in the ground-truth solutions, and that the identified scoring points and their values are reasonable and consistent. All manual verifications involved in constructing the benchmark are performed by 15 senior college students majoring in STEM disciplines.

C TRAINING DETAILS

We leverage Qwen2.5VL-32B-Instruct as the base model for both MatPlotCode and the CodePlot-CoT model. During training, we pad all input images into squares before resizing them randomly with lengths between 224 and 560. For rendered images in reasoning chain, we pad and embed them at a fixed resolution of 448×448. The training for MatPlotCode is a two-stage process: we first align the visual components by training only the vision encoder (ViT) and the MLP projector for one epoch, and then perform full-parameter fine-tuning for an additional two epochs. We use a batch size of 512 and a learning rate of 2e-5. For the main CodePlot-CoT model, we initialize its weights from our vision-aligned converter after the completion of Stage 1. We then fine-tune this model on our curated SFT dataset for 5000 steps, using a batch size of 256 and a learning rate of 3e-5. All settings are conducted on 32 NVIDIA H200 GPUs, with each training stage taking approximately 36 hours.

D CODEPLOT-COT SAMPLES ON MATH-VR BENCHMARK

We present some samples of CodePlot-CoT on Math-VR Benchmark in Figure 5 and 6. Figure 5 includes a math figure in input, while Figure 6 is purely text-based inputs. However, both problems require visual reasoning. As shown, our model demonstrates strong reasoning capabilities in both textual and visual reasoning, highlighting our code-driven visual reasoning paradigm.

E LIMITATIONS

Due to the limitations of data scale and model size, our MatPlotCode has not yet achieved a 100% fidelity rate on images-to-code conversion, which means that some visual reasoning images are not entirely accurate. As a result, our final model may also produce slightly imperfect visual reasoning in certain problems. For instance, in Figure 7, point H should lie on edge AD, but the generated image shows a slight displacement. We hope that future work will further advance the task of image-to-code conversion, thereby enabling a more robust realization of our paradigm.

<USER>: I will give you the question, analysis, and answer of a mathematical problem with the ID: {**idd**}. The problem may consist of a single question or multiple sub-questions.

1. Summarize the Answers: Clearly summarize all answers for the entire problem. Indicate how many answers there are and specify which answer corresponds to which question or sub-question.
2. List Scoring Points: For each question or sub-question, identify the scoring points. Scoring points are the critical steps needed to solve the problem, such as applying theorems, making necessary reasoning steps, or performing correct calculations. Please assign a score (e.g., 1, 2) for each scoring point based on its difficulty

Present your answer in the following JSON format: {**"id"**: [question id], **"scoring_points"**: [here should be a dictionary with scoring points: **"p1"**: [scoring point 1], **"p2"**: [scoring point 2],.....], **"scores"**: [here should be a dictionary with scores for the scoring points: **"s1"**: [score for p1], **"s2"**: [score for p2],.....], **"total_answer"**: [the number of all answers], **"answer_summary"**: [here should be a dictionary, the keys should match the questions, the values should be the answers]}

...Question...

<ASSISTANT>:

```
{ "id": 25006,
  "scoring_points": {
    "p1": "Recognize that AB is the diameter, thus angles ACB and ADB are right angles by the inscribed angle theorem.",
    "p2": "Apply the Pythagorean theorem to calculate BC in triangle ACB and BD in triangle ADB.",
    "p3": "Determine that triangle ABC is an isosceles right triangle to find angle BAC.",
    "p4": "Determine angle BAD in triangle ABD using calculated lengths.",
    "p5": "Classify the different configurations (AC and AD on the same or opposite sides of AB), and compute corresponding values for angle CAD.",
    "p6": "Combine stepwise results to conclude that the possible values for angle CAD are 15° or 75°." },
  "scores": { "s1": 1, "s2": 2, "s3": 1, "s4": 1, "s5": 2, "s6": 1 },
  "total_answer": 1,
  "answer_summary": { "1": "The possible values of angle CAD are 15° or 75°." },
  "max_score": 8 }
```

Figure 2: **Template used by GPT-4.1 to generate meta information for evaluation.** The text highlighted in cyan should be replaced with information of the specific question.

<USER>: You are an expert math teacher and grader. Your task is to evaluate a student's solution to a mathematical question and provide a score. You will be provided with the mathematical question (which may include multiple sub-questions), its ID, the student's solution, the correct answer, and the maximum possible score for the question below:

...Question...

```
{ 'id': {question_id}, 'student_solution': {model_response},
  'correct_answer': {correct_answer}, 'max_score': {max_score} }.
```

Please follow these steps precisely:

1. Initial Check for Correctness:
 - Thoroughly review the question and the student_solution to identify the student's final answer.
 - Compare this final answer directly with the provided correct_answer.
 - If the answers match exactly, award the full max_score.
2. Partial Credit Evaluation:
 - If the student's answer is not fully correct, evaluate its work for partial credit using the grading rubric: { 'scoring_points': {scoring_points}, 'point_values': {point_values} }.
 - Go through each scoring_point, indicate if the student successfully completed that step.
 - Write down all the point_ids that the student earned and calculate the total score by summing the values of those points.
3. Provide your evaluation in a strict JSON format:


```
{ "id": "string",
  "question_solution_analysis": "string"
  "is_fully_correct": "boolean",
  "check_scoring_point": "string",
  "awarded_points": ["all" OR a list of earned point_ids like "p1", "p2"],
  "final_score": "number" }
```

Field Explanations:

 - "id": question id.
 - "question_solution_analysis": Analyze the question requirements and compare the student's answer against the correct_answer."
 - "is_fully_correct": True if the student's solution is fully correct, otherwise False.
 - "check_scoring_point": If fully correct, provide an empty string "". If not fully correct, explain where in the student_solution each scoring point is fully met or not met.
 - "awarded_points": If fully correct, this should be ["all"]. If partially correct, provide a list containing the fully met point_ids (e.g., ["p1", "p3"]). If no points met, provide an empty list [].
 - "final_score": the max_score if fully correct, or the sum of partial scores otherwise.

Figure 3: Template used by GPT-4.1 to evaluate model response. The text highlighted in cyan should be replaced with information of the specific question. The orange model_response should be replaced with the response generated by the model given the question.

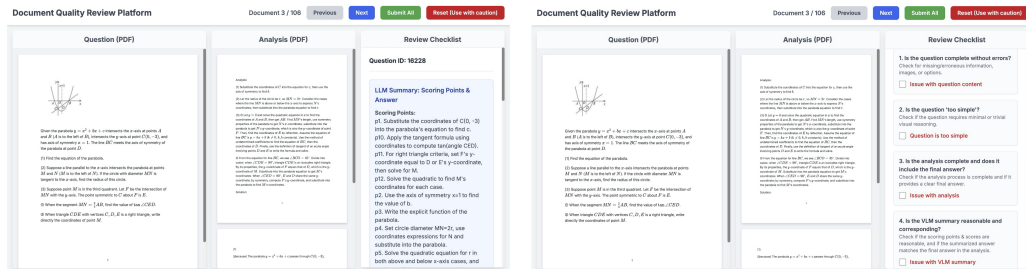
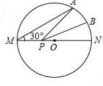


Figure 4: The GUI interface for manual sample selection and meta information verification. This interface displays the Question and Analysis files on the left and center panels, with the Review Checklist on the right. **Left:** The meta information extracted for each sample is displayed on the right panel. **Right:** Scrolling down reveals the items for annotators to check and flag.



As shown, MN is the diameter of circle $\odot O$ and the radius of $\odot O$ is 2. Point A lies on $\odot O$ and $\angle AMN = 30^\circ$. B is the midpoint of arc AN . P is a movable point on the diameter MN . Find the minimum value of $PA + PB$: _____.

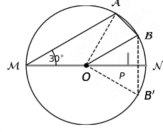
Analysis:

Reflect point B across line MN to get point B' , and connect AB' . By the property of axial symmetry, AB' gives the minimum value of $PA + PB$. From the properties of central and inscribed angles subtending the same arc:

- $\angle AON = 2\angle AMN = 2 \times 30^\circ = 60^\circ$
- By symmetry, $\angle B'ON = \angle BON = 30^\circ$

With these, calculate the measure of $\angle AOB'$, then use the Pythagorean theorem to obtain the answer.

Solution:



- Reflect point B across MN to get B' , and connect OA , OB , OB' , and AB' .
- The intersection of AB' with MN is the location where $PA + PB$ attains its minimum, thus $PA + PB$ minimum $= AB'$.

$$\because \angle AMN = 30^\circ$$

$$\therefore \angle AON = 2\angle AMN = 2 \times 30^\circ = 60^\circ$$

- B is the midpoint of arc AN :

$$\therefore \angle BON = \frac{1}{2} \times 60^\circ = 30^\circ$$

- By symmetry:

$$\angle B'ON = \angle BON = 30^\circ$$

$$\therefore \angle AOB' = \angle AON + \angle B'ON = 60^\circ + 30^\circ = 90^\circ$$

- Thus $\triangle AOB'$ is an isosceles right triangle:

$$AB' = \sqrt{2} \cdot OA = 2\sqrt{2}$$

Answer:

$$2\sqrt{2}$$

Figure 5: A multimodal math question from Math-VR benchmark and CodePlot-CoT generated solution.

For an isosceles triangle $\triangle ABC$ with a perimeter of 36 cm, if the altitude to its base BC is 12 cm, what is the value of $\cos B$? ()

Options:

A. $\frac{1}{2}$

B. $\frac{3}{2}$

C. $\frac{12}{13}$

D. $\frac{5}{13}$

Problem Analysis:

Let $AB = x$ cm, then $AC = AB = x$ cm, and $BC = (36 - 2x)$ cm.

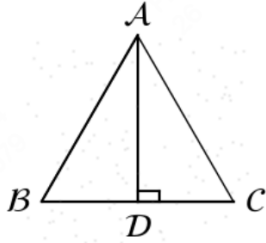
Calculate $BD = DC = \frac{1}{2}BC = (18 - x)$ cm.

In right triangle ABD , by the Pythagorean theorem, we have:

$$x^2 = 12^2 + (18 - x)^2$$

Solving gives $x = 13$, that is, $AB = 13$ cm, $BD = 5$ cm.

Problem Explanation:



Let $AB = x$ cm, then $AC = AB = x$ cm, and $BC = (36 - 2x)$ cm.

Since $AB = AC$ and AD is the altitude,

So $BD = DC = \frac{1}{2}BC = (18 - x)$ cm.

In right triangle ABD , by the Pythagorean theorem:

$$AB^2 = AD^2 + BD^2 \text{ That is, } x^2 = 12^2 + (18 - x)^2$$

Solving gives $x = 13$.

So $AB = 13$ cm, $BD = 5$ cm.

$$\text{Thus } \cos B = \frac{BD}{AB} = \frac{5}{13}.$$

Therefore, the answer is D.

Figure 6: A pure textual math question from Math-VR benchmark and CodePlot-CoT generated solution.

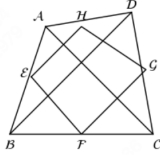
If the diagonals of a quadrilateral are equal, then the quadrilateral formed by sequentially connecting the midpoints of its sides must be ()

- Options:
 A. Trapezoid
 B. Rectangle
 C. Rhombus
 D. Square

Analysis:

Given that the diagonals of the quadrilateral are equal, by the Midline Theorem of triangles, it follows that all four sides of the resulting quadrilateral are equal, so the quadrilateral formed is a rhombus.

Solution:



As shown in the figure, $AC = BD$, and E, F, G, H are the midpoints of segments AB, BC, CD , and AD , respectively.

Since EH and FG are the midlines of $\triangle ABD$ and $\triangle BCD$, respectively, and EF and HG are the midlines of $\triangle ACD$ and $\triangle ABC$, respectively,

$$\therefore EH = FG = \frac{1}{2}BD, EF = HG = \frac{1}{2}AC$$

Since $AC = BD$

$$\therefore EH = FG = EF = HG$$

Therefore, quadrilateral $EFGH$ is a rhombus. Thus, the answer is C.

Figure 7: A pure textual math question from Math-VR benchmark and CodePlot-CoT generated solution. The textual reasoning is correct, but the model generates a slightly imperfect image in reasoning by placing H not on edge AD.