MMTutorBench: The First Multimodal Benchmark for AI Math Tutoring

Anonymous ACL submission

Abstract

Effective math tutoring requires not only solving problems but also diagnosing students' difficulties and guiding them step by step. While multimodal large language models (MLLMs) show promise, existing benchmarks largely overlook these tutoring skills. We introduce MMTutorBench, the first benchmark for AI math tutoring, consisting of 770 problems built around pedagogically significant keysteps. Each problem is paired with problem-specific rubrics that enable fine-grained evaluation across six dimensions, and structured into three tasks—Insight Discovery, Operation Formulation, and Operation Execution. We evaluate 12 leading MLLMs and find clear performance gaps between proprietary and open-source systems, substantial room compared to human tutors, and consistent trends across input variants: OCR pipelines degrade tutoring quality, few-shot prompting yields limited gains, and our rubric-based LLM-as-a-Judge proves highly reliable. These results highlight both the difficulty and diagnostic value of MMTutorBench for advancing AI tutoring.

1 Introduction

Math tutoring is one of the most important pillars of K-12 education. Many children either lack proper guidance in learning mathematics or receive ineffective support. Psychology research shows that such experiences can lead to "math anxiety." In mild cases, this anxiety causes children to lose confidence in learning math; in more severe cases, it dampens their motivation to learn knowledge and skills more broadly (Wigfield and Meece, 1988; Ashcraft, 2002; Barroso et al., 2021). Studies further reveal that parents themselves often experience anxiety when helping their children with math, and math-anxious parents can unintentionally undermine their children's performance, which can create an unconstructive cycle for children's math learning (Oh et al., 2022).

Can AI assist with math tutoring? Being an effective math tutor is non-trivial. Imagine a child working on a problem but getting stuck or making mistakes. For a human or an intelligent system to help effectively, it must have at least five key abilities. First, it needs to "see" the problem clearly, recognizing what is being asked. Second, it must understand the problem, apply knowledge, and use chain-of-thought reasoning to solve it correctly. Third, and more importantly, it should interpret *why* the child is struggling by analyzing the context of the child's problem-solving process and identifying the core concepts or ideas that need clarification. Fourth, it should then clarify *what* mathematical operation or method connects to that concept or idea. Finally, it should provide guidance on *how* to take the next concrete step, enabling the child to continue independently, rather than simply revealing the full solution.

Apparently, such an intelligent system would need to be a multimodal large language model (MLLM), and benchmarking is the primary way to evaluate its abilities. For the first two abilities, there already exist related benchmarks. For example, HME100K (Yuan et al., 2022), OCRBench (Fu et al., 2024), and MathWriting (Gervais et al., 2025) can evaluate an MLLM's capacity to extract handwritten text (including math formulas) from images. Math-Vision (Wang et al., 2024a), MM-Math (Sun et al., 2024), and others can assess a model's ability to solve math problems at various levels. However, when we focus on the three core tasks of AI math tutoring, namely, identifying the key insights, key operations, and next steps that provide effective support within the context of a child's problem-solving process, such benchmarks are still missing. Filling this gap is essential for enabling AI to deliver truly effective math tutoring.

We present MMTutorBench, the first multimodal benchmark for AI math tutoring. It evaluates MLLMs across diverse mathematical domains and

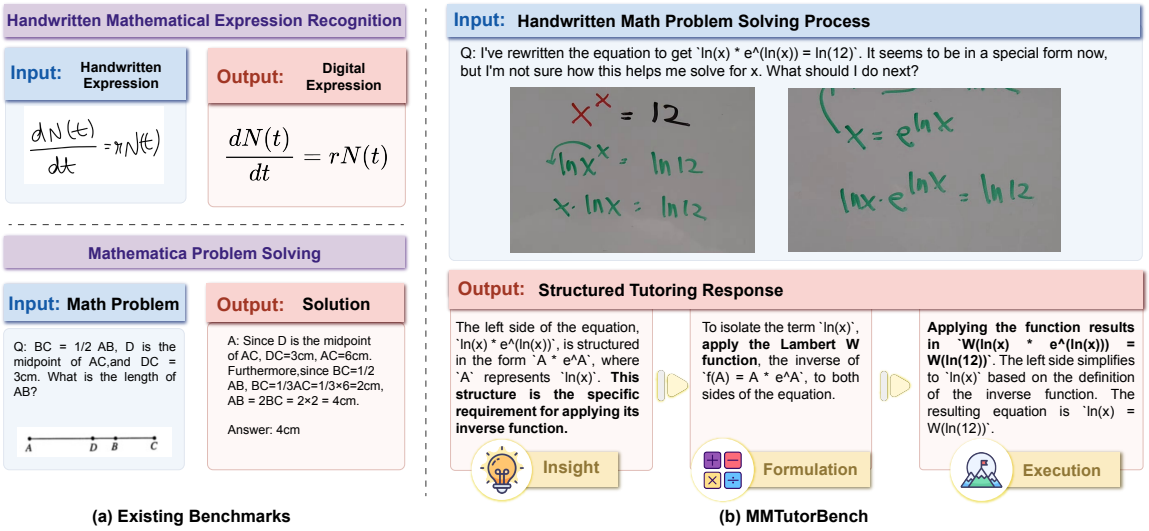


Figure 1: **(a)** Existing benchmarks usually target a single perspective, such as handwritten expression recognition or problem solving, which is insufficient for evaluating tutoring ability in real educational settings. **(b)** An example from MMTutorBench: we model the tutoring process in realistic classroom scenarios by taking a student's handwritten solution attempt and help-seeking question as input. The tutoring response is structured along three dimensions: Insight, Formulation, and Execution. We emphasize some key guidance in bold for illustration.

education levels, comprising 770 problems centered on pedagogically significant key-steps where students often struggle. Each problem includes three tasks—Insight Discovery, Operation Formulation, and Operation Execution (Figure 1)—reflecting the stepwise nature of tutoring. Evaluation follows a rubric-guided framework scoring six fine-grained dimensions to ensure comprehensive assessment. We benchmark 12 leading MLLMs, revealing clear performance stratification between proprietary and open-source models and across tutoring-specific aspects. Beyond overall scores, we analyze input configurations: OCR-first pipelines degrade performance by losing spatial and diagrammatic cues, while few-shot prompting yields limited, model-dependent gains. Our rubric-based LLM-as-a-Judge also shows high inter-judge agreement, confirming evaluation reliability.

2 Related Work

Our work is situated at the intersection of two active research areas: the application of Large Language Models (LLMs) in math tutoring and the development of multimodal mathematical reasoning capabilities. We review relevant literature in both domains to contextualize our contribution.

2.1 LLMs in Math Tutoring

Recent research has explored Large Language Models (LLMs) as scalable, personalized math tutors, primarily focusing on text-based dialogues. Studies have trained models to generate effective responses (Scarlatos et al., 2025) and predict tutor strategies (Ikram et al., 2025). The focus has also extended to evaluation, with systems like MathTutorBench (Macina et al., 2025) using reward models to assess tutors on dimensions such as subject expertise and student understanding, while other work has used LLMs as evaluators for human tutors (Thomas et al., 2025). However, this text-centric paradigm overlooks critical real-world complexities. Existing multimodal systems have centered on affective dimensions, such as student emotion (Kar et al., 2025), rather than on the interpretation of visual mathematical content. Furthermore, studies confirm that an LLM's problem-solving proficiency does not equate to effective tutoring (Gupta et al., 2025), and even state-of-the-art models remain prone to subtle reasoning errors (Zhang and Graf, 2025). To address the significant gap in visual-mathematical interpretation, MMTutorBench shifts the focus from purely textual dialogues to the multimodal task of interpreting a student's handwritten solution steps to provide effective feedback.

2.2 Multimodal Math Reasoning Benchmarks

In parallel, the field of multimodal mathematical reasoning has advanced through key benchmarks designed for visually-presented problems. MathVista (Lu et al., 2024) first establishes a foundational standard for comprehensive, reasoning-

centric evaluation. MATH-Vision (Wang et al., 2024b) enhances problem difficulty and diversity by drawing from real math competitions, while MathVerse (Zhang et al., 2024) probes the depth of visual understanding by presenting problems in multiple variations. Shifting the focus from outcomes to the reasoning process itself, We-Math (Qiao et al., 2024) pioneers fine-grained metrics for assessing principles like knowledge acquisition and generalization. However, these benchmarks are united by their singular focus on **problem-solving**. In contrast, MMTutorBench redefines the evaluation by assessing a model's ability to **act as a tutor**, a task centered on interpreting a student's incomplete handwritten intermediate steps to provide context-aware, scaffolded feedback.

3 MMTutorBench

MMTutorBench is a comprehensive benchmark consisting of 770 carefully curated samples from real-world educational settings, designed to evaluate the tutoring capabilities of MLLMs. In each sample, the model acts as a math tutor, interpreting students' handwritten solutions and generating responses to guide them through challenging steps.

To construct these scenarios, we collect educational video frames and student-posed questions to simulate authentic handwritten problem-solving and help-seeking processes (§ 3.1). We then decompose the tutoring objective into three task dimensions—Insight, Formulation, and Execution—following Pólya's problem-solving principle (Schoenfeld, 1987) (§ 3.2). Finally, we design a rubric-guided evaluation framework that employs problem-specific rubrics for fine-grained, multi-perspective assessment of MLLM outputs (§ 3.3). Comprehensive statistics are provided in Table 1 and Appendix F.

3.1 Problem Collection

Video Selection. Mathematics educational videos that visually capture handwritten problem-solving processes with step-by-step explanations provide a natural foundation for constructing MMTutorBench. To this end, we curate a corpus of 292 high-quality instructional videos drawn from 14 mathematics-focused YouTube channels[1]. The collection spans diverse mathematical domains (e.g., Algebra, Calculus) and educational stages ranging from junior and senior high school to university level.

[1] https://youtube.com.

Statistic	Number
Total Problems	770
Total Images	1,414
Images per Problem (1 / 2 / ≥3)	415 / 205 / 150
Question	
Total Words	233,447
Total / Unique Tokens	330,460 / 1,342
Avg. / Max. / Min. Tokens	429.17 / 463 / 399
Reference Answer	
	Insight / OpForm. / OpExec. / Total
Total Words	26,794 / 15,874 / 24,375 / 67,043
Total Tokens	40,947 / 22,344 / 47,831 / 111,122
Unique Tokens	1,930 / 1,433 / 128 / 3,491
Avg. Tokens	53.2 / 29.0 / 62.1 / 144.3
Max. Tokens	123 / 89 / 192 / 404
Min. Tokens	21 / 7 / 13 / 41
Rubrics	
Total Words	262,182
Total / Unique Tokens	363,276 / 2,547
Avg. / Max. / Min. Tokens	442.48 / 624 / 314

Table 1: Statistics of MMTutorBench. The token number is counted by GPT-4o tokenizer. `Insight`, `OpForm.`, `OpExec.` are abbreviations for `Insight Discovery`, `Operation Formulation`, `Operation Execution`.

Key-step Identification. Since mathematical problem solving involves multiple intermediate steps, we focus on the pedagogically critical ones—*key-steps*, where learners often face confusion or require deeper reasoning. These typically involve applying a core theorem (e.g., Pythagorean theorem) or executing a pivotal algebraic operation (e.g., polynomial factoring). To identify them in tutoring videos, we use Gemini-2.5-Pro (Comanici et al., 2025) to detect key-step timestamps, extract corresponding frames, and conduct manual quality checks meticulously. Further details appear in Appendix A.1.

Context Reconstruction. Educational videos are inherently dynamic (e.g., camera movement, page turning), which often causes crucial information—such as the problem statement or earlier steps—to move outside the frame. Thus, a single key-step frame may lack the broader context needed for comprehension. We first detect scene changes and extract representative frames. Human annotators then refine these frames by removing redundancy and filling gaps to ensure coherent context (details in Appendix A.2). Based on this contextualized representation, we then construct a tutoring question that simulates the inquiry a student would typically pose upon encountering difficulty at the key-step.

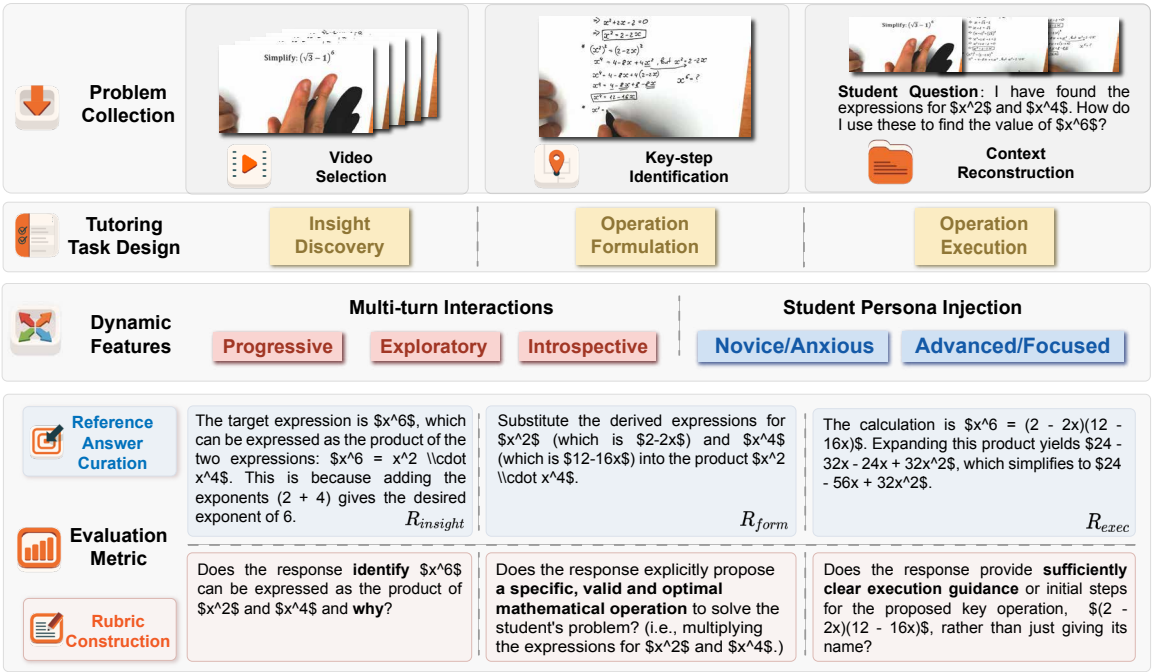


Figure 2: The data curation pipeline of MMTutorBench. We start by collecting problems including both images and questions. The model are instructed to fulfill 3 tutoring tasks for the input problem.

3.2 Tutoring Task Design

With the specified visual and question inputs, we design tutoring-centered tasks that specify how models should respond. Rather than providing complete solutions, the tasks are structured to guide learners step by step, thereby cultivating transferable problem-solving skills. Inspired by Pólya's problem-solving methodology (Schoenfeld, 1987), which frames reasoning as a staged process, including understanding the problem, devising a plan, and carrying out the plan, our benchmark operationalizes each stage at the level of a key-step through three tasks:

- **[Insight Discovery]** demonstrates the "*why*": the core principle or observation needed to make progress. It aims to help the student understand the underlying concept rather than only memorizing a procedure.
- **[Operation Formulation]** clarifies the "*what*": the specific mathematical operation or concept that should be applied based on the key insight.
- **Operation Execution]** explains the "*how*": the concrete execution of the prescribed operation, showing the immediate next step in the calculation without revealing the entire solution.

At inference, the tutor model receives the contextualized visuals, the student's question, and the task instructions, and completes the tasks sequentially (full prompt in Appendix B).

Dynamic Interaction and Adaptability. To simulate realistic educational dialogues beyond single-turn QA, we extend the task design into two advanced dimensions:

- **Multi-turn Interactions:** We strictly categorize student queries into three pedagogical levels—*Progressive* (linear follow-up), *Exploratory* (lateral clarification), and *Introspective* (deep conceptual justification)—to test the model's ability to maintain scaffolding over time.
- **Student Persona Injection:** We introduce specific personas (e.g., *Novice/Anxious* vs. *Advanced/Focused*) via system prompts to evaluate whether models can dynamically adjust their tone and granularity.

Detailed definitions and results analysis for these scenarios are provided in Appendix H and Appendix D.

3.3 Evaluation Metric

Evaluating tutoring responses is challenging because the task is open-ended: there is no single "correct" answer that can be matched by accuracy or n-gram overlap. Traditional metrics such as BLEU (Papineni et al., 2002) are thus inadequate. LLM-as-a-Judge methods offer a promising alternative, but naïvely applied they risk introducing bias and inconsistency (Ye et al., 2024). To address this, we adopt a rubric-guided LLM-as-a-Judge framework, inspired by BiGGenBench (Kim et al., 2025). The key idea is to anchor the evaluation

Category	Dimension	Evaluation Criteria (Condensed)
General	`Brevity`	Assesses whether the response is concise yet sufficient, avoiding redundancy while maintaining coverage comparable to the reference.
	`Coherence`	Assesses whether the response is logically consistent, factually accurate, and free of contradictions, relative to the reference.
Specific	`Insight Discovery`	Examines whether the response identifies the key structure or observation required at this stage, consistent with R_{insight}.
	`Operation Formulation`	Evaluates whether the response proposes the appropriate next conceptual operation, as indicated by R_{form}.
	`Operation Execution`	Evaluates whether the response correctly and transparently performs the intended operation, as defined in R_{exec}.
	`Solution Scope Control`	Checks whether the response remains focused on the current step, without advancing beyond R_{insight}, R_{form}, R_{exec}.

Table 2: Six-dimensional rubric for evaluating tutoring responses. Each dimension is operationalized relative to the step-specific reference answers R_{insight}, R_{form}, R_{exec}.

of each sample to a problem-specific rubric, rather than relying on generic criteria.

Reference Answer Curation. For each key-step sample we derive reference answers from the instructor's explanation in the post-key-step content of the video. We deliberately extract only the content relevant to the immediate next step to ensure the evaluation focuses on the current tutoring step rather than the full solution. Based on this focused content, we construct three reference answers by human annotation and denote them as R_{insight} for Insight Discovery, R_{form} for Operation Formulation, and R_{exec} for Operation Execution.

Rubric Generation. With the reference answers, we define six task-specific rubric dimensions. The first two, *Brevity* and *Coherence*, capture general qualities of effective instructional text. The next three, *Insight Discovery*, *Operation Formulation*, and *Operation Execution*, correspond directly to the structured tasks required by our benchmark. The final dimension, *Solution Scope Control*, penalizes responses that provide the full solution instead of stepwise tutoring. A detailed explanation of each dimension and its scoring criteria is provided in Table 2. The judge model evaluates candidate responses strictly against this rubric, rather than comparing them to references in a free-form way. This decomposition reduces the cognitive load on the judge model, improves consistency, and enables fine-grained assessment of both solution correctness and pedagogical effectiveness. The complete rubric is detailed in Table 2, with implementation details provided in Appendix A.3.

4 Experiments

We evaluate 12 leading MLLMs on our MMTutorBench to assess their capabilities in multimodal tutoring, study the impact of various input configurations through ablation, validate our LLM-as-a-Judge evaluation framework, and analyze the primary failure modes of the top-performing model.

4.1 Experimental Setup

To comprehensively evaluate our benchmark, we select 12 MLLMs, which span both proprietary and open-source models. Our evaluation suite includes 5 proprietary models: GPT-5 (OpenAI, 2025b), GPT-4o (OpenAI, 2024), Gemini-2.5-Pro (Comanici et al., 2025), Gemini-2.0-Flash, and GPT-o3-2025-04-16. Additionally, we assess 9 leading open-source models: Qwen2.5-VL (7B, 72B) (Bai et al., 2025), InternVL3.5 (8B, 38B) (Wang et al., 2025), Gemma3-27B-it (Kamath et al., 2025), GLM-4.1V-9B-thinking (Hong et al., 2025), and MiMo-VL-7B (Xia et al., 2025). For all experiments, we employ a standardized prompt (detailed in Appendix B) to ensure a fair comparison. Unless otherwise specified, our default experimental setting is zero-shot, where models are provided only with the task instruction and the relevant images, without any in-context examples or student query. We assess responses using the rubric in Table 2, where each of the six criteria receives a binary score (0 or 1) for a maximum total score of 6.

4.2 Main Results

Table 3 summarizes the comprehensive evaluation results for all 12 models on MMTutorBench under

Model	Tot.	Insight	OpForm.	OpExec.	Scope	Brevity	Coherence
Proprietary Models							
Gemini-2.5-Pro	**4.69**	**0.79**	**0.73**	**0.73**	**0.69**	0.78	**0.97**
GPT-5	4.32	0.76	0.70	0.70	0.36	**0.83**	**0.97**
GPT-o3	3.97	0.68	0.66	0.66	0.28	0.72	**0.97**
Gemini-2.0-Flash	3.77	0.54	0.56	0.61	0.44	0.75	0.87
GPT-4o	3.18	0.50	0.47	0.47	0.28	0.64	0.81
Open-Source Models							
Qwen2.5-VL-72B	**3.40**	**0.53**	**0.51**	**0.57**	0.32	0.60	0.86
InternVL3.5-38B	3.26	**0.53**	0.49	0.55	0.22	0.58	**0.88**
InternVL3.5-8B	3.17	0.43	0.44	0.49	0.26	**0.69**	0.85
Gemma-3-27B	2.87	0.38	0.37	0.39	**0.35**	0.66	0.72
MiMo-VL-7B	2.78	0.51	0.46	0.48	0.25	0.35	0.73
GLM-4V-9B	2.55	**0.53**	0.50	0.54	0.12	0.11	0.75
Qwen2.5-VL-7B	2.52	0.31	0.28	0.30	**0.35**	0.59	0.68

Table 3: Performance comparison of various models on our MMTutorBench. We report the total score (Tot.) and a detailed breakdown across six dimensions in our rubric. Column headers are abbreviations for: `Insight Discovery`, `Operation Formulation`, `Operation Execution`, `Solution Scope Control`, `Brevity`, and `Coherence`.

the default setting. Our analysis of this data yields several key insights into the current capabilities and limitations of MLLMs on this challenging task.

A clear performance gap remains between proprietary and open-source models. As shown in Table 3, there is a clear distinction in performance between the two model categories. The leading proprietary model, Gemini-2.5-Pro, achieves a total score of 4.69. In contrast, the top-performing open-source model, Qwen2.5-VL-72B, scored 3.40. This substantial gap of 1.29 points highlights that state-of-the-art proprietary systems still hold a considerable advantage in tackling the complex, multifaceted reasoning required by our benchmark.

All models show a clear gap from human level. To establish an upper bound for performance, we evaluated human expert responses on a subset of the data (detailed in Table 4). The total human score reached 5.85, demonstrating a high standard of pedagogical quality. Even the most capable model in our evaluation, Gemini-2.5-Pro (4.69), remains more than a full point below this human baseline. This gap underscores the profound difficulty of the task and indicates that current MLLMs have not yet mastered the nuanced skills required for effective multimodal tutoring.

Tutoring Mode struggles with scope control. Intriguingly, models designed for specific educational scenarios do not always excel on our benchmark. As shown in Table 4, the **GPT-4o Study Mode** (OpenAI, 2025a), tailored for learning applications, achieves a total score of 3.15, comparable to the standard GPT-4o's 3.21. A closer look at the score breakdown shows a key trade-off: while the study mode may show competence in identifying insights (Insight: 0.62 vs. 0.53), it exhibits a severe deficiency in managing the answer's boundaries, scoring only 0.11 in **Solution Scope Control**. This failure to adhere to the problem's scope while attempting to be more explanatory demonstrates the benchmark's capacity to test not just correctness, but also crucial pedagogical skills like conciseness and focus. The poor performance of this specialized mode further validates the challenging and comprehensive nature of MMTutorBench.

Model	Tot.	Insight	OpForm.	OpExec.	Scope	Brevity	Coh.
Human	5.85	0.97	0.97	0.97	0.97	0.98	0.98
GPT-4o	3.21	0.53	0.45	0.51	0.29	0.62	0.80
GPT-Study	3.15	0.62	0.47	0.53	0.11	0.51	0.91

Table 4: Performance comparison between human, GPT Study Mode, and GPT-4o on a 10% subset of MMTutorBench (66 samples). Coh. denotes Coherence.

4.3 Advanced Tutoring Capabilities

Beyond single-step correctness, effective tutoring requires sustained scaffolding and pedagogical flexibility. We evaluate models on multi-turn consistency and persona adaptability using GPT-5 as a representative case study.

Multi-turn Scenarios. The results reveal a significant inverse correlation between context length and scaffolding discipline. While GPT-5 maintains robust diagnostic accuracy across turns (*In-*

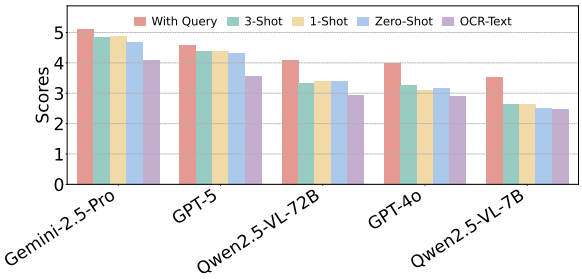


Figure 3: Performance comparison of various models across five distinct input variants. The variants include: (a) **Zero-Shot**, where only the images is provided; (b) **With Query**, which supplements the images with a corresponding textual student query; (c) **OCR-Text**, a pipeline approach where text is first extracted from the images via an OCR model and then fed to the language model; (d) **1-Shot** and (e) **3-Shot**, which provide one and three in-context examples, respectively. The results highlight the significant performance boost from including student queries and the critical limitations of the OCR-based pipeline approach.

sight score increases from 0.87 to 0.91), its ability to constrain the solution scope degrades sharply, with the *Solution Scope Control* score dropping from 0.18 in Turn 2 to 0.09 in Turn 3. This deficiency is most pronounced in introspective queries requiring conceptual justification; in such cases, the scope control score collapses to 0.00, indicating that the model fails to withhold the final answer when pressed for deeper explanations. (See Appendix H for the complete performance dynamics across interaction types.)

Student-Level Adaptability. The results highlight a substantial gap between problem-solving and tutoring capabilities: while GPT-5 achieves a high *Insight* score of 0.72, its *Adaptivity* score is disproportionately low at 0.30. This points to inherent behavioral rigidity, where models disregard prompted constraints on tone and granularity, reverting instead to their default, neutral training patterns regardless of the student's simulated needs. Full quantitative results and the rubric for adaptivity alignment are detailed in Appendix D.

4.4 Ablation Study on Input Variants.

Impact of Few-Shot Prompts We evaluate in-context learning across zero-, 1-, and 3-shot settings. As shown in Figure 3, few-shot prompting yields only marginal and model-dependent gains. While top-performing models, such as Gemini-2.5-Pro, improve slightly (4.69 to 4.86 in 3-shot), GPT-4o conversely exhibits performance degradation (3.18 to 3.09 in 1-shot). This variability suggests that while in-context learning can offer a slight advantage for state-of-the-art models, the foundational reasoning capabilities of the model remain the dominant factor for success on our benchmark.

Impact of Student Queries. To evaluate the impact of textual context, we compare performance in the image-only setting (Zero-Shot) versus the setting supplemented by textual student queries. The inclusion of student queries yields universal gains across all models, ranging from +0.42 for Gemini-2.5-Pro (4.69 to 5.11) to +1.01 for Qwen2.5-VL-7B (2.52 to 3.53). This trend suggests that textual queries serve as a powerful focusing mechanism to ground visual analysis, bypassing the more ambiguous and error-prone task of inferring students' confusion from visual context alone. This finding underscores the critical role of explicit, text-based cues in enabling effective multimodal tutoring.

Impact of Modality: Image vs. OCR-Text To investigate the importance of direct visual processing, we compare the end-to-end multimodal approach with a pipeline method. In the latter, the powerful OCR model, MiniCPM4.1-8B(Team, 2025), first extracts text from the image, which is then processed by the MLLM. This method causes significant performance degradation across most models—for instance, Gemini-2.5-Pro dropped from 4.69 to 4.16—confirming that direct visual analysis is indispensable. This indicates that semantically crucial visual cues—such as the spatial layout of equations, diagrams, and non-textual symbols—are inadequately captured in a text-only representation, validating that end-to-end visual understanding is essential for genuine multimodal comprehension in our benchmark.

4.5 Rubrics Effectiveness

To address the complex task of assessing tutoring effectiveness and correctness, we employ a rubric-based LLM-as-a-Judge. This section validates this methodology by examining two key aspects: its correlation with human judgment (validity) and consistency across evaluators (reliability).

To **validate** our methodology, we first benchmark it against human expert scores. As shown in Table 6, our rubric-based evaluation strongly correlates with human judgment (Pearson's $r = 0.725$), significantly outperforming all comparison baselines, thus confirming its **validity**.

We then establish the rubric's **reliability** through inter-judge analysis between `GPT-o4-mini` and

Model	Tot.	Insight	OpForm.	OpExec.	Scope	Brevity	Coherence
Gemini-2.5-Pro	4.77/4.87	0.80/0.80	0.74/0.74	0.74/0.74	0.71/0.71	0.80/0.93	0.98/0.94
GPT-5	4.41/4.40	0.77/0.72	0.71/0.68	0.72/0.68	0.38/0.44	0.85/0.93	0.98/0.94
InternVL3.5-8B	3.19/3.10	0.43/0.43	0.45/0.40	0.50/0.48	0.27/0.29	0.68/0.80	0.86/0.74
MiMo-VL-7B-RL	2.75/2.78	0.51/0.46	0.46/0.44	0.48/0.46	0.26/0.26	0.34/0.59	0.72/0.58

Table 5: Inter-judge reliability analysis on four representative models. A random 90% sample of the data is utilized for scoring comparison. Scores are presented in the format **GPT-o4-mini / Qwen3-30B-A3B-Instruct-2507**.

Category	Metric	Spearman (ρ)	Pearson (r)
Embedding-based	BERTScore	0.230	0.219
Rule-based	BLEU	0.233	0.267
	ROUGE-L	0.341	0.386
LLM-as-a-Judge	Standard Judge	0.563	0.625
	Ours	**0.652**	**0.725**

Table 6: Correlation with human expert judgments. Traditional rule-based and embedding-based metrics fail to capture pedagogical nuances, whereas our metric demonstrates superior alignment with expert scores.

`Qwen3-30B-A3B-Instruct-2507` (Yang et al., 2025). The scores demonstrated exceptionally high agreement, with a Pearson correlation exceeding 0.98 across the dataset (visualized in Appendix C). This high degree of concordance, exemplified by nearly identical scores for models like GPT-5 (4.41 vs. 4.40), confirms that our evaluation is robust and minimizes judge-specific bias, ensuring the objectivity of our reported results.

4.6 Error Analysis

To understand the primary failure modes of the model, we conduct a detailed error analysis on the top-performing model, Gemini-2.5-Pro, by categorizing the instances where the model scored zero across our six evaluation dimensions. Figure 4 presents the proportion of samples that failed in each dimension.

As illustrated, the most significant challenge for the model lies in Solution Scope Control, with nearly one-third (31.04%) of its responses failing to adhere to the required scope of the solution. This is closely followed by failures in Operation Execution (27.14%) and Operation Formulation (26.88%). These three dimensions collectively indicate that while the model may identify a path forward, it struggles profoundly with correctly executing the necessary steps and constraining its output to the appropriate level of detail, often providing overly complex or irrelevant information.

Furthermore, the model exhibits considerable weaknesses in Brevity (22.08%) and Insight Discovery (21.17%). This suggests that in about one-fifth of cases, the model either fails to produce a concise answer or misses the core insight required

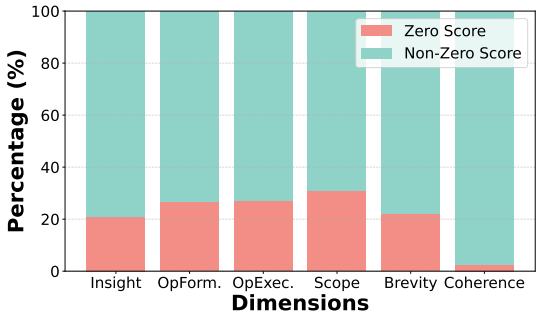


Figure 4: Error distribution for the top-performing model, Gemini-2.5-Pro. The chart displays the percentage of samples that received a score of zero in each of our six evaluation dimensions.

to solve the problem efficiently. These issues often compound the problems with operational control, leading to responses that are both incorrect and verbose.

A noteworthy finding, however, is the model's exceptional performance in Coherence. With a failure rate of only 2.73%, the model's outputs are almost always logically structured, fluent, and easy to follow. This reveals a critical disparity: the model has mastered linguistic and structural coherence, but still lacks the deeper reasoning and self-control capabilities required for precise operational execution and scope management. The outputs are often well-formed but substantively flawed.

5 Conclusion

We introduce MMTutorBench, a comprehensive benchmark for evaluating Multimodal Large Language Models (MLLMs) on mathematical tutoring tasks. Our evaluation of 12 leading models reveals a significant performance gap between proprietary and open-source systems, with all models falling substantially short of human expert performance. We also show that direct visual grounding is indispensable, as text-only inputs are insufficient for effective tutoring. Furthermore, our findings indicate that while most models possess foundational visual understanding and problem-solving capabilities, they struggle to grasp the pedagogical concept of tutoring and often fail to appropriately control the scope of their guidance.

Limitations

We acknowledge two primary limitations. First, although our questions are pedagogically anchored in authentic video timestamps, they are simulated and may not fully capture the linguistic ambiguity and spontaneity inherent in real-world learner interactions. Second, its scope is confined to English-language mathematics, which limits the generalizability of our findings across other subjects and languages. Future work could address these limitations by incorporating real-world classroom transcripts and expanding the dataset to more subjects and languages.

References

Mark H Ashcraft. 2002. Math anxiety: Personal, educational, and cognitive consequences. *Current directions in psychological science*, 11(5):181–185.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Ming-Hsuan Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025. Qwen2.5-vl technical report. *CoRR*, abs/2502.13923.

Connie Barroso, Colleen M Ganley, Amanda L McGraw, Elyssa A Geer, Sara A Hart, and Mia C Daucourt. 2021. A meta-analysis of the relation between math anxiety and math achievement. *Psychological bulletin*, 147(2):134.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit S. Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, Luke Marris, Sam Petulla, Colin Gaffney, Asaf Aharoni, Nathan Lintz, Tiago Cardal Pais, Henrik Jacobsson, Idan Szpektor, Nan-Jiang Jiang, and 81 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *CoRR*, abs/2507.06261.

Ling Fu, Zhebin Kuang, Jiajun Song, Mingxin Huang, Biao Yang, Yuzhe Li, Linghao Zhu, Qidi Luo, Xinyu Wang, Hao Lu, and 1 others. 2024. Ocrbench v2: An improved benchmark for evaluating large multimodal models on visual text localization and reasoning. *arXiv preprint arXiv:2501.00321*.

Philippe Gervais, Anastasiia Fadeeva, and Andrii Maksai. 2025. Mathwriting: A dataset for handwritten mathematical expression recognition. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 5459–5469.

Adit Gupta, Jennifer Reddig, Tommaso Calo, Daniel Weitekamp, and Christopher J. MacLellan. 2025. Beyond final answers: Evaluating large language models for math tutoring. *Preprint*, arXiv:2503.16460.

Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, Shuaiqi Duan, Weihan Wang, Yan Wang, Yean Cheng, Zehai He, Zhe Su, Zhen Yang, Ziyang Pan, Aohan Zeng, and 58 others. 2025. Glm-4.1v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. *CoRR*, abs/2507.01006.

Fareya Ikram, Alexander Scarlatos, and Andrew Lan. 2025. Exploring llms for predicting tutor strategy and student outcomes in dialogues. *Preprint*, arXiv:2507.06910.

Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière,

Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean-Bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, Gaël Liu, and 191 others. 2025. Gemma 3 technical report. *CoRR*, abs/2503.19786.

Debanjana Kar, Leopold Böss, Dacia Braca, Sebastian Maximilian Dennerlein, Nina Christine Hubig, Philipp Wintersberger, and Yufang Hou. 2025. Mathbuddy: A multimodal system for affective math tutoring. *arXiv preprint arXiv:2508.19993*.

Seungone Kim, Juyoung Suk, Ji Yong Cho, Shayne Longpre, Chaeeun Kim, Dongkeun Yoon, Guijin Son, Yejin Cho, Sheikh Shafayat, Jinheon Baek, Sue Hyun Park, Hyeonbin Hwang, Jinkyung Jo, Hyowon Cho, Haebin Shin, Seongyun Lee, Hanseok Oh, Noah Lee, Namgyu Ho, and 13 others. 2025. The biggen bench: A principled benchmark for fine-grained evaluation of language models with language models. *Preprint*, arXiv:2406.05761.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2024. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations (ICLR)*.

Jakub Macina, Nico Daheim, Ido Hakimi, Manu Kapur, Iryna Gurevych, and Mrinmaya Sachan. 2025. Mathtutorbench: A benchmark for measuring open-ended pedagogical capabilities of LLM tutors. *CoRR*, abs/2502.18940.

Dajung Diana Oh, Michael M Barger, and Eva M Pomerantz. 2022. Parents' math anxiety and their controlling and autonomy-supportive involvement in children's math learning: Implications for children's math achievement. *Developmental Psychology*, 58(11):2158.

OpenAI. 2024. Hello gpt-4o. News announcement by OpenAI.

OpenAI. 2025a. Chatgpt study mode. https://openai.com/zh-Hans-CN/index/chatgpt-study-mode/. Accessed: 2025-10-05.

OpenAI. 2025b. Gpt-5 is here. Technical report, OpenAI. News announcement by OpenAI.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Runqi Qiao, Qiuna Tan, Guanting Dong, Minhui Wu, Chong Sun, Xiaoshuai Song, Zhuoma GongQue, Shanglin Lei, Zhe Wei, Miaoxuan Zhang, and 1 others. 2024. We-math: Does your large multimodal model achieve human-like mathematical reasoning? *arXiv preprint arXiv:2407.01284*.

Alexander Scarlatos, Naiming Liu, Jaewook Lee, Richard Baraniuk, and Andrew Lan. 2025. *Training LLM-Based Tutors to Improve Student Learning Outcomes in Dialogues*, page 251–266. Springer Nature Switzerland.

Alan H. Schoenfeld. 1987. Pólya, problem solving, and education. *Mathematics Magazine*, 60(5):283–291.

Kai Sun, Yushi Bai, Ji Qi, Lei Hou, and Juanzi Li. 2024. Mm-math: Advancing multimodal math evaluation with process evaluation and fine-grained classification. *arXiv preprint arXiv:2404.05091*.

MiniCPM Team. 2025. MiniCPM4: Ultra-efficient llms on end devices.

Danielle R. Thomas, Conrad Borchers, Jionghao Lin, Sanjit Kakarla, Shambhavi Bhushan, Erin Gatz, Shivang Gupta, Ralph Abboud, and Kenneth R. Koedinger. 2025. Leveraging llms to assess tutor moves in real-life dialogues: A feasibility study. *Preprint*, arXiv:2506.17410.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. 2024a. Measuring multimodal mathematical reasoning with math-vision dataset. *Advances in Neural Information Processing Systems*, 37:95095–95169.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. 2024b. Measuring multimodal mathematical reasoning with math-vision dataset. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, Zhaokai Wang, Zhe Chen, Hongjie Zhang, Ganlin Yang, Haomin Wang, Qi Wei, Jinhui Yin, Wenhao Li, Erfei Cui, and 56 others. 2025. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *CoRR*, abs/2508.18265.

Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612.

Allan Wigfield and Judith L Meece. 1988. Math anxiety in elementary and secondary school students. *Journal of educational Psychology*, 80(2):210.

Bingquan Xia, Bowen Shen, Cici, Dawei Zhu, Di Zhang, Gang Wang, Hailin Zhang, Huaqiu Liu, Jiebao Xiao, Jinhao Dong, Liang Zhao, Peidian Li, Peng Wang, Shihua Yu, Shimao Chen, Weikun Wang, Wenhan Ma, Xiangwei Deng, Yi Huang, and 44 others. 2025. Mimo: Unlocking the reasoning potential of language model - from pretraining to posttraining. *CoRR*, abs/2505.07608.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Jiayi Ye, Yanbo Wang, Yue Huang, Dongping Chen, Qihui Zhang, Nuno Moniz, Tian Gao, Werner Geyer, Chao Huang, Pin-Yu Chen, Nitesh V Chawla, and Xiangliang Zhang. 2024. Justice or prejudice? quantifying biases in llm-as-a-judge. *Preprint*, arXiv:2410.02736.

Ye Yuan, Xiao Liu, Wondimu Dikubab, Hui Liu, Zhilong Ji, Zhongqin Wu, and Xiang Bai. 2022. Syntax-aware network for handwritten mathematical expression recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4553–4562.

Liang Zhang and Edith Aurora Graf. 2025. Mathematical computation and reasoning errors by large language models. *Preprint*, arXiv:2508.09932.

Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Peng Gao, and 1 others. 2024. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? *arXiv preprint arXiv:2403.14624*.

Stage	Stats	Rejection Criteria
Video Sel.	$\sim 25.6\%$	Videos are discarded due to poor handwriting legibility, low resolution, or lack of clear audio-visual alignment.
Key-Step Ext.	$\sim 63.4\%$	Annotators filtered out LLM-suggested steps that are redundant, lack handwritten context, or involve pure calculation without pedagogical value.
Rubric Ver.	100% Ver. 12.8% Corr.	Expert annotators manually verify all generated rubrics. Approximately 12.83% require correction to remove factual errors to ensure strict alignment with the QA pairs.

Table 7: Statistics of the data construction pipeline. The rigorous filtering and verification process ensures high data quality. (Sel.: Selection, Ext.: Extraction, Ver.: Verification, Corr.: Corrected)

A Detailed Data Construction

A.1 Key-step Frame Extraction

Our key-step frame extraction protocol is designed to systematically identify the most pedagogically valuable moments from each source video. The process is as follows:

1. **Automated Candidate Generation:** We employ Gemini-2.5-Pro (Comanici et al., 2025), a powerful multimodal model, to analyze the full content of each video. Guided by a carefully crafted prompt, the model is instructed to identify pivotal steps in the problem-solving process.

2. **Timestamp Pair Generation:** For each pivotal moment, the model outputs a **pair of precise timestamps** (in `HH:MM:SS` format) and a brief justification. This pair consists of the timestamp for the **critical step** itself and the timestamp for the **immediately preceding step**. To ensure our benchmark contains a diverse set of problems, we limit the extraction to a maximum of five such pairs per video.

3. **Frame Pair Extraction:** The generated timestamp pairs are then used with the FFmpeg to extract the corresponding two static image frames for each identified moment.

4. **Human-in-the-Loop Verification and Curation:** This phase is the core of our quality control. As detailed in Table 7, we applied strict rejection criteria at multiple stages:
   - **Video Selection:** Prior to extraction, approximately **25.6%** of raw videos are discarded due to poor legibility or audio-visual misalignment.
   - **Step Filtration:** During the manual review of extracted frames, annotators filter out **63.4%** of the LLM-suggested candidates. Steps are rejected if they are redundant, lack handwritten context, or involve pure calculation without pedagogical value.

5. **Reliability Assessment:** To validate our annotation standards, we conduct a dual-blind annotation on 10% of the data prior to the full-scale process. The experts achieve an **Inter-Annotator Agreement (IAA) of >90%** on key-step identification, establishing a solid gold standard for the dataset creation.

Prompt for Key-step Frame Extraction from Tutoring Videos

Persona
You are a multimodal AI assistant analyzing a math tutoring video. Extract timestamps of key instructional frames that satisfy all of the following.

Task
Capture each **critical step** in the math tutoring video **and** the **immediately preceding step** before that critical step.
NOTE
1. **Critical Step**: a pivotal stage in problem solving (e.g., equation transformation, formula introduction, concept explanation, etc.).
2. You must also capture the step just before the critical step, where the key equation is **about to appear but has not yet appeared**. The **previous step** and the **critical step** should be tightly connected; the only difference is whether the key equation is present.
3. Limit the total to **at most 5** pairs—choose the most representative moments.

Requirement
1. **Clear Handwriting**: The handwriting relevant to the step (equations, diagrams, etc.) is fully written and legible.
2. **Peak Clarity**: The handwriting is stable and complete—not mid-writing, blurry, or partially shown.
3. **Audio-Visual Match**: The narration clearly refers to the handwriting visible on screen.

Exclude
- Writing only the problem number/title.

- Frames with blurry or incomplete handwriting.
- Transition frames (e.g., board erasing).
- Final boxed answers without explanation.

Output Requirement
Return a **chronological list** of timestamp dictionaries with concise justifications.

Output Format

```
{
  keyframe_timestamp: [MM:SS],
  prev_step_timestamp: [MM:SS],
  reason: 'xxx',
}
```

*Note: These three fields denote the timestamp of the **critical step**, the timestamp of its **immediately preceding** step, and a brief justification for the key instructional moment, respectively.*

A.2 Context Reconstruction

Our context reconstruction protocol is designed to provide a comprehensive visual narrative leading up to each extracted key-step frame. Since a single frame often lacks the preceding information necessary for full comprehension (e.g., the original problem statement), this process segments the source video into a sequence of visually coherent images. The process is as follows:

1. **Automated Scene Segmentation:** The process begins by programmatically identifying significant visual shifts in the video. We compute the **Structural Similarity Index (SSIM)** (Wang et al., 2004) score between every pair of consecutive frames. A potential scene boundary is flagged wherever this score drops below a threshold. To ensure that only meaningful transitions are retained and to filter out noise from transient motions (e.g., camera jitter), we apply a temporal filter that enforces a **minimum time interval** between consecutive boundaries.

   The formal algorithm is a two-step process. First, a set of candidate timestamps, T_{cand}, is identified:

   $$T_{\text{cand}} = \{t_n \mid \text{SSIM}(I_{n-1}, I_n) < \tau\}, \quad (\text{we use } \tau = 0.8)$$

   Second, this candidate set is filtered iteratively to produce the final set of boundaries, T_{sb}, ensuring each boundary is separated by a minimum duration, Δt_{min}:

   $$t_{s_1} = t'_1 \quad (\text{where } t'_1 \text{ is the earliest candidate})$$

   $$t_{s_j} = \min\{t'_i \in T_{\text{cand}} \mid |t'_i - t_{s_{j-1}}| \geq \Delta t_{\text{min}}\}, \quad \text{for } j > 1$$

2. **Representative Frame Extraction:** Once the final set of scene boundaries is established, a single, clear **representative frame** is extracted from each resulting video segment. This transforms the video into an initial, condensed sequence of static images that summarizes the visual progression of the solution.

3. **Manual Verification and Curation:** Similar to our key-step frame extraction, this phase is crucial for data quality. Our annotation team meticulously reviews the automatically generated sequence of representative frames. Their task is to refine this sequence by removing redundant images and supplementing any missing frames to repair logical or visual discontinuities. This meticulous curation ensures that the context provided for each key-step is a coherent and complete narrative.

A.3 Rubric Generation

To enhance the accuracy and reliability of our evaluation, we developed problem-specific rubrics. The generation process for each sample's corresponding rubric is as follows:

1. **Question-Answer Pair Extraction:** Our process begins by analyzing the video transcripts. We first employ Gemini-2.0-Flash to process the subtitles and isolate the core mathematical problem-solving steps relevant to each key-step frame. Based on these extracted, concise solution steps, we then generate corresponding question-answer pairs. This output subsequently undergoes manual refinement, where annotators polish the questions to be clear and self-contained, and trim the answers to represent pedagogically meaningful steps.

2. **Automated Rubric Generation:** Based on each sample's question-answer pair and its full set of contextual images, we use Gemini-2.5-Pro with a structured prompt to generate scoring criteria for four specific dimensions:

Insight Discovery, **Operation Formulation**, **Operation Execution**, and **Solution Scope Control**. These criteria are then combined with the requirements for two general dimensions (**Brevity** and **Coherence**) to create a preliminary six-dimensional rubric.

3. **Manual Verification and Curation:** The auto-generated rubrics undergo a rigorous manual verification process to ensure their precision and fairness. Our annotators meticulously review each scoring criterion, performing corrections, additions, or deletions as needed. The primary task is to ensure that the rubric's specific dimensions (**Insight Discovery**, **Operation Formulation**, **Operation Execution**) precisely and exclusively map to the corresponding components of the reference answer. This involves rephrasing ambiguous descriptions, clarifying conditions for earning points, and removing any criteria that do not directly pertain to the specific problem, thereby creating a highly reliable, problem-specific evaluation standard. While 100% of the rubrics are manually reviewed, approximately 12.8% required correction to fix factual errors and ensure strict alignment with the QA pairs.

Prompt for Transcript Processing

You are helping clean up a noisy transcript from a math teaching video. Your task is to extract and return exactly **one** clear, concise sentence that conveys the core math explanation.

- Remove all filler words, background chatter, repetitions, or unrelated talk.
- Keep only mathematical explanation or reasoning.
- Do not include greetings, pauses, or commentary like 'so yeah', 'okay', 'um', etc.
- Make sure the sentence is standalone, complete, and easy to understand.

[Noisy Transcript]
`{text}`

[Denoised Math Explanation (One Sentence)]
...

Prompt for Generating Q&A Pairs

Role-Playing
You are a precise and logical tutor who guides students step by step through problem-solving. You do not need to solve the entire problem; you only need to instruct the student's question on the next key step and the reason for doing so. Your responses are purely **analytical and instructional**, with **no emotional tone** and **no conversational language**.

Task Description
Based on the learning context provided below, you need to generate two parts:

1. **Student's Question:** Simulate a student who is trying to move forward in the problem but is confused about **what to do next**, based on the current step. The question should sound natural and authentic, using first-person language like "What should I do now?" or "How do I continue from here?" Notice: the question should NOT mention any specific mathematical operations or concepts when asking for the next step, but rather focus on the general direction or operation to take.
2. **Tutor's Answer:** Provide a **precise and impersonal response** following the format:
   - **[key detail]**: extract the key detail in the student's current state of the solution including the image and text, and explain the rationale for paying attention to this key detail.
   - **[key operation]**: Based on the key detail, provide the very next critical operation the student should perform.
   - **[next step]**: Perform the key operation in detail and determine the result.

Key Example (Few-Shot)
Image of current step: This is an example image of the current step of student's problem-solving process. `'img:{few_shot_img_path}'`
Context:

- **Initial entire problem and student's current problem-solving stage:** "one plus a thousand to the power four plus a thousand and one to the power four divided by one plus a thousand squared plus a thousand and one squared. to simplify this fraction i'm going to let 1000 = x then this fraction becomes one plus x to power four and a thousand and one equals x plus one so it's x plus one all to the power four and this sum is divided by 1 plus x squared plus x plus 1 all squared. now i can simplify this algebraic expressionlet's do the easier one first which is 1 plus x squared plus x squared plus two x plus one. we can use the binomial theorem and we have x to the four and then it's 4 choose 1 x cubed 4 choose 2 x squared 4 choose 3 x and then it's just 1 at the end so i will not write 4 choose 4 but i'll just write 1 at the end. and we simplify we'll have 2 x to the power 4 plus 4 x cubed plus 6 x squared plus 4 x plus 2 at the top. we have 2 x squared plus 2x and then plus 2. and we realize that for the both the top and the bottom we have two so we'll cancel them out and we'll have x to the power four plus two x cubed plus three x squared plus two x plus one."
- **Student's current point of confusion:** "Three x squared plus two x plus one is the result."
- **Tutor's answer:** "The best solution is to factorize the numerator. If you cannot see how what do we get if we fracture this fraction if not then we can just use long division and you realize that with x

squared on top. So it's 2x here not x and then we remove that we'll have x squared plus x plus one at the end so it turns out that the top is actually x squared plus this plus one all squared and then we can have a simplified expression and since x equals a thousand then the rest is very easy one million one thousand and one so this."

Required Output: Student: Okay, so I've simplified the top and bottom to get $x^4 + 2x^3 + 3x^2 + 2x + 1$ over $x^2 + x + 1$. I'm not really sure how to simplify this further. What should I do now?

Tutor:

[key detail]: We can find that the coefficients of the numerator polynomial $(x^4 + 2x^3 + 3x^2 + 2x + 1)$ are palindromic (or symmetric) and the numerator might be a multiple of the denominator, specifically its square.

[key operation]: Thus we can consider factoring the numerator to attempt canceling the denominator and simplifying the fraction.

[next step]: Factorize the numerator into $(x^2 + x + 1)(x^2 + x + 1)$. Then cancel the denominator $(x^2 + x + 1)$ from the numerator, resulting in a simplified expression of $(x^2 + x + 1)$.

Your Task
Image of current step: `'img:{img_path}'`
Context:

- **Entire problem and student's current problem-solving stage:**

  `{before}`

- **Student's current point of confusion:**

  `{sentence}`

- **Tutor's answer:**

  `{after}`

Required Output: Student: ...
Tutor: ...

Prompt for Generating Rubrics

Persona: You are a meticulous AI assistant specializing in educational assessment. Your purpose is to craft precise, objective, and detailed evaluation rubrics. These rubrics are used to judge the quality of AI-generated hints for students solving math problems.

Goal
Your task is to generate a detailed, 4-point scoring rubric. This rubric will be used to evaluate a model's response to a specific student's question. You will create the rubric referenced on the provided question and its answer. Based on the output format example, you will fill in the "xxx" parts.

Note
When `condition_for_1` and `condition_for_0` parts of your rubric refer to specific information about the problem, it MUST be **decisive**; words like "such as" or "for example" that could mislead the scoring process are *not* allowed.

Output format

Notice, the criteria and `id` in the `evaluation_criteria` should be identical to the few-shot examples. The rubric should be in JSON format, with the following structure:

```
{
    "task_description": "You are an AI
     evaluator. Please assess an AI's
   response to a student's math question
     about xxx, based on the following
     `evaluation_criteria`. For each
    criterion, assign a score of 0 or 1,
    and summarize all scores in a single
     JSON object.",
     "evaluation_criteria": [
         {
             "id": "insight_discovery",
         "criterion": "Does the response
            identify and point out a key
             structure or feature in the
         expression that aids in solving
             the problem?",
             "condition_for_1": "xxx",
             "condition_for_0": "xxx"
         },
         {
         "id": "operation_formulation",
         "criterion": "Does the response
          explicitly propose a specific,
          valid and optimal mathematical
        operation to solve the student's
             problem?",
             "condition_for_1": "xxx",
             "condition_for_0": "xxx"
         },
         {
           "id": "operation_execution",
         "criterion": "Does the response
             provide sufficiently clear
           execution guidance or initial
             steps for the proposed key
            operation, rather than just
             giving its name?",
             "condition_for_1": "xxx",
             "condition_for_0": "xxx"
         },
         {
        "id": "solution_scope_control",
         "criterion": "Is the response a
        focused hint that only guides the
         current step, rather than giving
        a lengthy explanation or the full
         answer? Note: If the response is
        not a focused hint for the correct
         step and is completely incorrect
             or unhelpful,
             the solution_scope_control
         score should be 0, regardless of
 whether the condition_for_1 below is met.
 If the response is a focused hint for the
  correct step, you still need to check if
   it meets the condition_for_1 below.",
             "condition_for_1": "xxx",
             "condition_for_0": "xxx"
         }
    ],
   "output_format_instruction": "Please
   strictly adhere to this JSON format for
     the output:
```

```
    {\"insight_discovery\": <0|1>,
    \"operation_formulation\": <0|1>,
    \"operation_execution\":
  <0|1>, \"solution_scope_control\": <0|1>}"
   }
```

Few-shot examples:
`{rubric_few_shot}`

Now, please generate a rubric for the following question and answer, using JSON format:
Question: question
Reference Answer: answer
Images: 'img:img_path', **Previous Images:** ', '.join(f"'img:p'" for p in prev_img_paths)

`Rubric:`

B Prompts for Structured Output Generation

This section presents the exact prompts used to guide the model's generation process, ensuring full reproducibility of our experiments. We designed two prompt variants to handle distinct input conditions. The primary prompt, **Task Instruction without Student Query**, is used for tasks where the model must reason solely from the visual context of the provided images. The second, **Task Instruction with Student Query**, is an extension that incorporates an explicit student's question via the `{question}` placeholder.

Both prompts are structured to elicit a three-part response: '[Insight Discovery]', '[Operation Formulation]', and '[Operation Execution]'. They also include placeholders like `{few_shots}` for injecting few-shot examples and `{prev_imgs_str}`/`{kf_img_path}` for the image inputs.

Task Instruction w/o Student Query

You are a precise and logical tutor who guides students step by step through problem-solving.

Task Description
A student is working on his math homework but got stuck after completing a few steps and does not know how to proceed. You will be given: (1) a series of previous images in chronological order that show the original problem (the first image) and the student's step-by-step problem-solving process; (2) a single current image that shows the student's current solution process; (3) a question that the student is asking about the next step; based on the images and question, you need to identify the student's point of confusion and provide guidance on **the next key step** and **the detailed rationale for executing the key step**.

Note: the next key step should be the single, most logical next step required to continue solving the problem. Your responses should be purely analytical and instructional, with no emotional tone and no conversational language. Your responses **MUST** be precise and concise. Do **NOT** include any unnecessary, overly long, or multi-step subsequent steps.

Your response must include the following three parts:

- **Insight Discovery**: extract the key detail in the student's current state of the solution including the image and text, and explain the rationale for paying attention to this key detail.
- **Operation Formulation**: Based on the key detail, provide the very next critical operation the student should perform.
- **Operation Execution**: Perform the key operation in detail and determine the result.

`{few_shots}`

Image

- Previous images (chronological): `{prev_imgs_str}`
- Current image: `'img:{kf_img_path}'`

Output
Use standard Latex notation for mathematical expressions. Your response **MUST** follow this format:

- **[Insight Discovery]**: xxx
- **[Operation Formulation]**: xxx
- **[Operation Execution]**: xxx

Task Instruction w/ Student Query

Role
You are a precise and logical tutor who guides students step by step through problem-solving.

Task Description
A student is working on his math homework but got stuck after completing a few steps and does not know how to proceed. You will be given: (1) a series of previous images in chronological order that show the original problem (the first image) and the student's step-by-step problem-solving process; (2) a single current image that shows the student's current solution process; (3) a question that the student is asking about the next step; based on the images and question, you need to identify the student's point of confusion and provide guidance on **the next key step** and **the detailed rationale for executing the key step**.

Note: the next key step should be the single, most logical next step required to continue solving the problem. Your responses should be purely analytical and instructional, with no emotional tone and no conversational language. Your responses **MUST** be precise and concise. Do **NOT** include any unnecessary, overly long, or multi-step subsequent steps.

Your response must include the following three parts:

- **Insight Discovery**: extract the key detail in the student's current state of the solution including the image and text, and explain the rationale for paying attention to this key detail.
- **Operation Formulation**: Based on the key detail, provide the very next critical operation the student should perform.

- **Operation Execution**: Perform the key operation in detail and determine the result.

`{few_shots}`

Image

- Previous images (chronological): `{prev_imgs_str}`
- Current image: `'img:{kf_img_path}'`

Student's question
`{question}`

Output
Use standard Latex notation for mathematical expressions. Your response **MUST** follow this format:

- **[Insight Discovery]**: xxx
- **[Operation Formulation]**: xxx
- **[Operation Execution]**: xxx

C Evaluation Validation

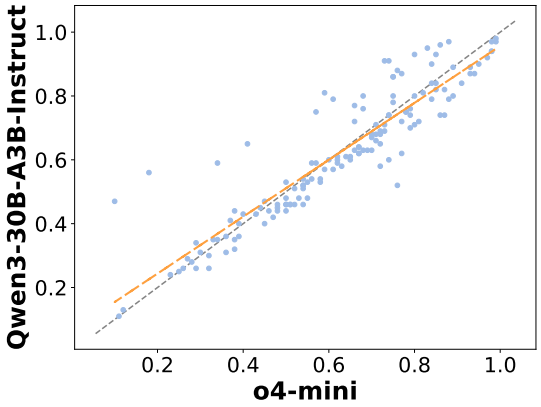


Figure 5: Inter-judge reliability of our evaluation rubric. The plot shows the correlation between average scores assigned to all 12 MLLMs by two independent judge models: GPT-o4-mini and Qwen3-30B-A3B-Instruct-2507.

To supplement our inter-judge reliability analysis in the main text, Figure 5 provides a scatter plot of the average scores assigned to all 12 MLLMs by the two independent judge models, demonstrating a high alignment between evaluation scores from smaller, open-source reasoning models and their more powerful, proprietary counterparts.

Furthermore, Table 5 reveals that our meticulously crafted, problem-specific criteria result in higher scoring consistency for the four specific dimensions across models of varying capabilities. In contrast, the two general dimensions (Brevity and Coherence), which use uniform criteria for all samples, show a greater but still acceptable variance in scores. This highlights the effectiveness of our problem-specific rubric design and the robustness of our overall evaluation framework.

D Evaluation of Student-Level Adaptability

To further assess the pedagogical capabilities of MLLMs beyond mere problem-solving, we conduct an additional experiment focusing on **Student Adaptivity**. This experiment evaluates whether models can dynamically adjust their explanatory tone and granularity based on specific student personas defined in the system prompt.

D.1 Experimental Setup

We introduce a **Persona-Based Injection** protocol. For each problem in the benchmark, we condition the model with one of two distinct student metadata profiles via the system prompt. The detailed instructions and constraints for each persona are contrasted in Table 8.

D.2 Evaluation Metric

We employ an LLM-as-a-Judge approach using `o4-mini` to quantify adaptability. Unlike standard correctness metrics, this metric focuses purely on pedagogical fit. We also introduce a binary rubric dimension, **Adaptivity Alignment**, as defined in Table 9.

We compare this adaptability score against the standard **Insight Score**, which measures the mathematical correctness and visual understanding of the solution.

D.3 Results and Analysis

We evaluate two representative models, GPT-5 and Qwen2.5-VL-72B-Instruct. The results are summarized in Table 10.

Solving $\neq$ Tutoring. A critical finding from this experiment is the divergence between problem-solving capability and tutoring adaptability. As shown in Table 10, while **GPT-5** demonstrates superior mathematical competence (Insight of 0.72), it performs poorly in Adaptivity (0.30).

Qualitative analysis reveals that despite explicit system instructions, stronger models often suffer from *behavioral rigidity*, prioritizing their default training preference for "standard solutions" over the specific pedagogical needs of the user. This result underscores the unique value of our benchmark: it highlights that a strong "Math Solver" is not necessarily a capable "Adaptive Tutor", pointing to a

Dimension	Persona A: Novice/Anxious	Persona B: Advanced/Focused
Student Metadata	• **Proficiency:** Novice • **State:** Anxious/Frustrated • **Goal:** Confidence building	• **Proficiency:** Advanced • **State:** Neutral/Hurried • **Goal:** Efficiency & Key Tricks
Tone Instruction	Be **warm, encouraging, and supportive**. Use phrases like "You're doing great" or "Don't worry" to lower anxiety.	Be **direct, professional, and concise**. Avoid filler words or emotional support; treat the student as a peer.
Granularity	Break down the execution into **very simple, explicit micro-steps**. Do **NOT** skip any intermediate calculation.	**Skip trivial arithmetic** or algebraic manipulations. Focus only on non-trivial transformations.
Strategy	Explain the basic concept behind the insight **patiently**, assuming no prior intuition.	Highlight the core insight or **mathematical trick immediately**. Provide a quick hint rather than a lecture.

Table 8: System Prompt Instructions for Student Personas. We inject these specific constraints into the system prompt to evaluate the model's ability to adapt its pedagogical style.

Metric: Adaptivity Alignment (0 or 1)
Criterion: Does the tutor's response explicitly adapt its tone, granularity, and strategy to align with the specific STUDENT PERSONA constraints provided in the instructions?
Score 1 (Strict Alignment): The response successfully embodies the required persona: • *For Novice:* The tone is encouraging AND the explanation is detailed without skipping steps. • *For Advanced:* The tone is concise/direct AND trivial steps are omitted to focus on the core insight. The response follows the specific formatting or stylistic constraints requested.
Score 0 (Generic/Mismatched): The response fails to adapt or contradicts the persona: • Uses a generic, robotic tone regardless of the student's state. • Is mismatched (e.g., overly verbose/pedantic for an Advanced student, or too abstract for an Anxious Novice). • Ignores specific instructions regarding granularity (e.g., skipping steps when asked not to).

Table 9: Rubric for Adaptivity Alignment. This binary metric evaluates style and pedagogical fit, independent of mathematical correctness.

crucial direction for future alignment research in educational AI.

E Weighted Evaluation Framework

In our main evaluation, we report unweighted averages across all rubric dimensions. However, not all dimensions are equally critical for effective tutoring. For instance, correctly diagnosing a student's misconception (Insight) is arguably more fundamental than the conciseness of the response (Brevity).

Model	Insight Score (Math Comp.)	Adaptivity Score (Pedagogical Fit)
GPT-5	0.72	0.30
Qwen2.5-VL-72B	0.51	0.27

Table 10: Evaluation of student-level adaptivity. The significant gap between Insight and Adaptivity scores reveals that strong mathematical competence does not inherently translate into effective pedagogical flexibility.

To validate the robustness of our benchmark, we introduce a **Weighted Evaluation Framework**. As detailed in Table 11, we assign distinct weights to each dimension to reflect their pedagogical priority.

E.1 Rationale for Weight Assignment

We categorize the dimensions into four priority levels based on educational taxonomy:

1. **Critical (25%):** *Insight Identification*. This is the "soul" of tutoring. Without accurate diagnosis of the mathematical structure, tutoring is impossible.

2. **High (20%):** *Solution Scope Control*. This distinguishes a "tutor" from a "solver." It ensures the model scaffolds the learning rather than revealing the final answer.

3. **Standard (15% each):** *Coherence, Operation Prescription, Execution*. These represent the baseline correctness and methodological soundness.

4. **Secondary (10%):** *Brevity*. While concise language reduces cognitive load, it is secondary to factual accuracy and pedagogical strategy.

Dimension	Weight	Pedagogical Rationale
Insight Identification	**0.25**	*Diagnosis Capability.* It serves as the foundation of scaffolding, requiring the model to identify the deep mathematical structure rather than just calculating numbers.
Solution Scope Control	**0.20**	*Pedagogical Pacing.* Critical for preventing "spoilers." It forces the model to guide the student step-by-step rather than outputting the final result immediately.
Coherence	0.15	*Reliability Baseline.* In math tutoring, tolerance for hallucinations or contradictions is near zero. Factual errors negate all educational value.
Op. Prescription	0.15	*Methodology.* Bridging "Insight" and "Execution" by explicitly stating the correct strategic path (e.g., "Use factorization").
Op. Execution	0.15	*Demonstration.* While important, "pointing the way" (Prescription) is often more pedagogically valuable than "doing the math" (Execution) for the student.
Brevity	0.10	*User Experience.* Concise responses lower cognitive load, but this is a "nice-to-have" quality compared to correctness and pedagogical validity.

Table 11: **Pedagogical weight assignment.** Weights are distributed to prioritize diagnostic insight and scaffolding control over stylistic attributes.

E.2 Results and Consistency

We re-evaluate the top-performing models using this weighted scheme. As shown in Table 12, the relative ranking of the models remains entirely stable compared to the unweighted averages.

Model	Weighted Score	Ranking Stability
Gemini-2.5-Pro	**4.75**	Remains 1st
GPT-5	4.30	Remains 2nd
o3-2025	3.92	Remains 3rd
Qwen2.5-VL-7B	2.54	Stable

Table 12: Performance under weighted evaluation. The consistency in ranking confirms that performance gaps stem from core tutoring capabilities rather than trivial metrics.

E.3 Conclusion

The stability of the rankings confirms that the performance superiority of leading models (e.g., Gemini-2.5-Pro) stems from their robust capabilities in core tutoring dimensions—specifically *Insight* and *Scope Control*—rather than marginal advantages in lower-weighted metrics like Brevity.

F Benchmark Statistics

Statistics of MMTutorBench are summarized in Table 1. The benchmark comprises 770 problems incorporating 1,414 images. In this benchmark, nearly half of the problems (46.1%) contain two or more images.

The textual components of the benchmark are comprehensive. Questions are detailed, averaging 429.17 tokens. Similarly, the reference answers are substantial, averaging 144.3 tokens per problem, and are structured into three distinct tasks: **Insight Discovery**, **Operation Formulation**, and **Operation Execution**.

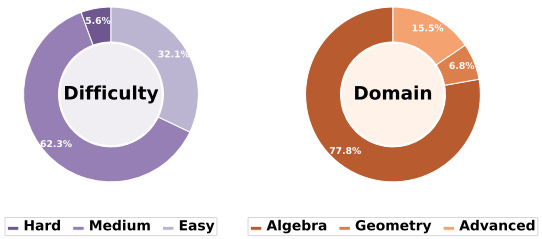


Figure 6: Distribution of benchmark statistics. The dataset features a tiered difficulty structure dominated by medium-level problems (Left) and categorizes mathematical domains into three meta-types (Right): Algebra, Geometry, and Advanced (which encompasses Calculus, Statistics, and Number Theory).

We further analyze the benchmark's composition by mathematical domain and difficulty level, as illustrated in Figure 6. The problems predominantly cover Algebra (77.8%), followed by Advanced topics (15.5%, encompassing Number Theory and Calculus) and Geometry (6.8%). In terms of difficulty, the benchmark is designed to be challenging, with a vast majority of problems classified as Hard (62.3%) or Medium (32.1%), ensuring a rigorous test of advanced reasoning capabilities.

G Detailed Analysis: Domain and Difficulty

To investigate the boundaries of model capabilities, we further dissect model performance across distinct mathematical domains and difficulty levels.

Domain Performance. As presented in Table 13, all evaluated models achieve their peak performance in the **Algebra** domain (e.g., Gemini-2.5-Pro achieves 4.81). This proficiency is likely attributed to the abundance of symbolic derivation data in pre-training corpora. In stark contrast, **Geometry** represents a significant bottleneck. Even the state-of-the-art model exhibits a substantial performance drop in this domain (Gemini-2.5-Pro drops to 3.67, and Qwen2.5-VL-72B-Instruct to 2.23). This stratification underscores the persistent challenge MLLMs face in tasks requiring intricate *visual-spatial reasoning*, as opposed to pure symbolic manipulation. The **Advanced** category sits between these extremes, indicating moderate difficulty in handling higher-order abstract concepts.

Domain	Algebra (N=599)	Geometry (N=52)	Advanced (N=119)
Model Performance			
Gemini-2.5-Pro	4.81	3.67	4.49
GPT-5	4.46	3.28	3.87
Qwen2.5-VL-72B-Instruct	3.52	2.23	3.29

Table 13: Model Performance across Different Mathematical Domains

Difficulty Analysis. Table 14 elucidates the correlation between problem complexity and tutoring quality. We observe a consistent performance degradation across all models as difficulty scales from **Easy** to **Hard**. Notably, proprietary models demonstrate superior *robustness* in high-complexity scenarios. specifically, Gemini-2.5-Pro maintains a commendable score of 4.02 on **Hard** problems, whereas Qwen2.5-VL-72B-Instruct suffers a sharp decline to 2.30. This disparity highlights that while open-source models show promise in handling fundamental tasks, they lack the deep reasoning capabilities required to effectively tutor students through complex, multi-step problems.

Difficulty	Easy (N=247)	Medium (N=480)	Hard (N=43)
Model Performance			
Gemini-2.5-Pro	4.90	4.64	4.02
GPT-5	4.48	4.29	3.81
Qwen2.5-VL-72B-Instruct	3.68	3.35	2.30

Table 14: Model Performance across Different Difficulty Levels

H Evaluation in Multi-turn Scenarios

While the main experiments focus on single-turn interactions to establish a baseline for core tutoring capabilities, MMTutorBench is designed with a modular architecture that naturally extends to multi-turn dialogues. We posit that a single-turn response serves as the **"atomic unit"** of tutoring; if a model fails to demonstrate Insight, Formulation, or Scope Control in an individual turn, the entire pedagogical chain collapses.

To rigorously evaluate these capabilities in dynamic contexts, we extend the dialogue context for a representative subset of the single-turn samples, analyzing subsequent turns (Turn 2 and Turn 3) and introducing a taxonomy of student query types.

H.1 Performance Dynamics Across Turns

First, we analyze the temporal evolution of model performance. As shown in Table 15, the evaluation results of GPT-5 reveal distinct dynamics as the conversation deepens:

1. **Persistence of Insight:** Interestingly, the *Insight Identification* score improves slightly in Turn 3 (0.91) compared to Turn 2 (0.87). This suggests that as the context accumulates, strong models are capable of maintaining (or even refining) their mathematical understanding of the student's problem.

2. **Degradation of Control:** However, a critical failure mode emerges in *Solution Scope Control*. The score drops significantly from 0.18 in Turn 2 to 0.09 in Turn 3. This indicates that while the model understands the math (high Insight), it struggles to maintain pedagogical discipline over longer interactions, becoming prone to "spoiling" the answer rather than continuing to scaffold.

H.2 Analysis by Interaction Type

To further dissect the model's adaptability, we categorize multi-turn interactions into three distinct reasoning types based on the student's intent:

- **Progressive:** The student asks a question that builds upon or deepens the understanding from the previous turn, moving the solution process forward linearly.

- **Exploratory:** The student asks for clarification on the current level or explores a different aspect of the problem, representing a lateral movement in reasoning.

- **Introspective:** The student asks a question regarding the same concept as the previous turn but demands a deeper conceptual justification. This requires the tutor to demonstrate metacognitive understanding rather than just procedural execution.

As presented in Table 16, evaluating GPT-5 against these categories reveals a clear performance hierarchy that mirrors pedagogical complexity:

- **Linear Proficiency:** The model excels in **Progressive** tasks (Total Score: 4.49), demonstrating strong baseline capabilities in Insight (0.90) and Execution (0.85). This aligns with

the model's training on step-by-step reasoning chains.

- **Complexity Gap:** Performance declines in **Exploratory** scenarios (4.35) and drops significantly in **Introspective** tasks (4.00). notably, the *Solution Scope Control* score hits **0.00** for Introspective tasks. This critical finding indicates that when students demand deep conceptual explanations, models struggle to withhold the final answer, failing to balance "explaining why" with "scaffolding the how."

These findings demonstrate that our rubric is highly sensitive to the nuances of multi-turn dynamics, effective at distinguishing between a linear solver and a capable, adaptive tutor.

Metric	Turn 2 (N=168)	Turn 3 (N=82)
Overall Performance		
Average Score	4.47	4.40
Detailed Component Scores		
Insight Identification	0.87	**0.91**
Operation Prescription	0.84	0.88
Operation Execution	0.82	0.84
Solution Scope Control	0.18	**0.09**
Brevity	0.76	0.72
Coherence	1.00	0.96

Table 15: **Performance dynamics across subsequent turns.** While mathematical insight improves with deeper context (Turn 3), pedagogical control (Scope) degrades, highlighting the difficulty of maintaining scaffolding over multi-turn interactions.

Type	Progressive (N=182)	Exploratory (N=63)	Introspective (N=5)
Overall Performance			
Total Score	**4.49**	4.35	4.00
Detailed Scores			
Insight	0.90	0.84	0.80
OpForm.	0.87	0.81	0.80
OpExec.	0.85	0.78	0.80
Scope	0.15	0.16	**0.00**
Brevity	0.74	0.78	0.60
Coh.	0.99	0.98	1.00

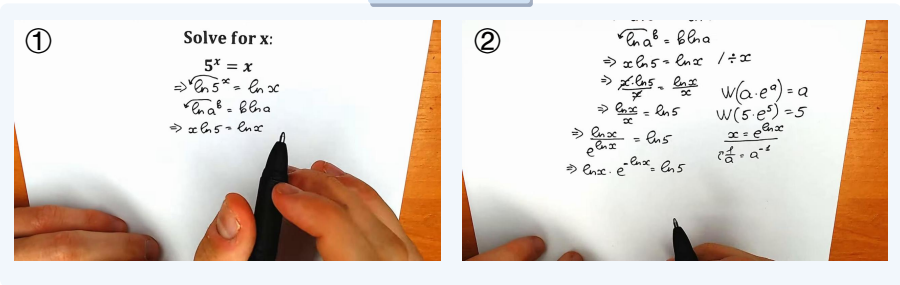
Table 16: **Performance across reasoning complexity levels.** The degradation in scores from Progressive to Introspective tasks confirms that the benchmark effectively differentiates between linear solving and complex, metacognitive reasoning—the "atomic" skills required for multi-turn tutoring.

I Case Study

Figures 7–8 illustrate our pipeline from response generation to evaluation, showcasing a high-scoring response from Gemini-2.5-Pro and a low-scoring one from Qwen2.5-VL-72B-Instruct. Gemini-2.5-Pro demonstrates strong tutoring capabilities, correctly inferring the student's confusion from the visual input alone and providing a pedagogically sound response. In contrast, Qwen2.5-VL-72B-Instruct adheres to the required three-part output format but fails to offer correct guidance.

Notably, the general dimensions of **Brevity** and **Coherence** are scored independently of a response's pedagogical value. Consequently, while the Qwen2.5-VL-72B-Instruct response lacks instructional merit, it still receives points on these dimensions for its accurate interpretation of the handwritten content and its conciseness. This case highlights the objectivity of our evaluation process and the rational design of our rubric.

Input Images



Student's Question

I've manipulated the equation to get $\ln(x) \cdot e^{-\ln(x)} = \ln(5)$. This looks very close to the form needed to use the Lambert W function. How do I proceed from here?

Reference Answer

[key detail]: The expression on the left side, $\ln(x) \cdot e^{-\ln(x)}$, does not match the required form $a \cdot e^a$ for the Lambert W function because the coefficient $\ln(x)$ is not identical to the exponent $-\ln(x)$.
[key operation]: To make the coefficient and the exponent identical, multiply both sides of the equation by -1.
[next step]: Performing the multiplication results in the equation $-\ln(x) \cdot e^{-\ln(x)} = -\ln(5)$. The left-hand side is now in the form $a \cdot e^a$, where $a = -\ln(x)$.

Figure 7: An example of input images, student question and refrence answer.

Rubric

Task Description
You are an AI evaluator. Please assess an AI's response to a student's math question about manipulating an equation for the Lambert W function, based on the following evaluation criteria. For each criterion, assign a score of 0 or 1, and summarize all scores in a single JSON object.

Evaluation Criteria

Insight Discovery
Does the response identify and point out a key structure or feature in the expression that aids in solving the problem?
Success Criterion: The response must **point out that the coefficient $ln(x)$ and the exponent $-ln(x)$ are not identical**, which prevents the direct application of **the Lambert W function**.
Failure Criterion: The response does not point out the mismatch between the coefficient $ln(x)$ and the exponent $-ln(x)$.

Operation Formulation
Does the response explicitly propose a specific, valid and optimal mathematical operation to solve the student's problem?
Success Criterion: The response must **propose multiplying the equation by -1**.
Failure Criterion: The response does not propose multiplying the equation by -1.

Operation Execution
Does the response provide sufficiently clear execution guidance or initial steps for the proposed key operation, rather than just giving its name?
Success Criterion: The response must show the result of multiplying the equation by -1, **resulting in the equation $-ln(x)\, e^{-ln(x)} = -ln(5)$**.
Failure Criterion: The response only suggests multiplying by -1 but does not show the resulting equation $-ln(x) * e^{-ln(x)} = -ln(5)$.

Solution Scope Control
Is the response a focused hint that only guides the current step, rather than giving a lengthy explanation or the full answer?
Success Criterion: The response must only guide the student on how to manipulate the equation into **the form $a * e^a = b$** and must not proceed to apply the Lambert W function or solve for x.
Failure Criterion: The response provides the full solution, including applying the Lambert W function to **find the value of $-ln(x)$** or **solving for x**.

Brevity
Is the response concise, avoiding unnecessary detail or repetition while still effectively addressing the task?
Success Criterion: The response delivers the essential idea or guidance in a brief and efficient manner, without including redundant phrases, excessive elaboration, or tangential information.
Failure Criterion: The response includes wordy explanations, repeated information, or elaborates beyond what is needed to fulfill the purpose of the response.

Gemini-2.5-Pro

Insight Discovery
The current equation is $ln(x) * e^{-ln(x)} = ln(5)$. To apply the Lambert W function, **which is defined by $W(Z) * e^{W(Z)} = Z$, the equation must be in the form $Y * e^Y = K$**. In the current state, the term multiplying the exponential, $ln(x)$, is not identical to the term in the exponent, $-ln(x)$.
Operation Formulation
Multiply both sides of the equation by -1. This will make the coefficient of the exponential term match its exponent, conforming to the structure required for the Lambert W function.
Operation Execution
$(-1) * [ln(x) * e^{-ln(x)}] = (-1) * ln(5)$,
$-ln(x) * e^{-ln(x)} = -ln(5)$,
The equation is now in the form $Y * e^Y = K$, where $Y = -ln(x)$ and $K = -ln(5)$.

Insight Discovery	1
Operation Formulation	1
Operation Execution	1
Solution Scope Control	1
Brevity	1
Coherence	1

Qwen2.5-VL-72B-Instruct

Insight Discovery
The key detail is the equation $ln(x) * e^{-ln(x)} = ln(5)$. This expression involves the application of the product of $ln(x)$ and its exponential. Paying attention to this is crucial because recognizing forms involving $ln(x)$ and $e^{-ln(x)}$ can often simplify to expressions involving x and its reciprocal.
Operation Formulation
Apply the property $e^{-ln(x)} = 1/x$.
Operation Execution
Using $e^{-ln(x)} = 1/x$ transforms the equation as follows:
$ln(x) * 1/x = ln(5)$,
simplifying to
$ln(x)/x = ln(5)$.

Insight Discovery	0
Operation Formulation	0
Operation Execution	0
Solution Scope Control	0
Brevity	1
Coherence	1

Coherence
Is the response internally coherent, factually accurate, and free of contradictions or disconnected statements?
Success Criterion: The response maintains logical flow, uses consistent language, and presents information that is factually correct without contradictions or shifts in meaning.
Failure Criterion: The response includes any of the following issues: (a) factual inaccuracies; (b) contradictions within the explanation; (c) disjointed or irrelevant statements that break the logical flow.

Figure 8: An evaluation rubric comparing a high-scoring response from Gemini-2.5-Pro with a low-scoring response from Qwen2.5-VL-72B-Instruct.