ERRORRADAR: Benchmarking Complex Mathematical Reasoning of Multimodal Large Language Models Via Error Detection

Anonymous ACL submission

Abstract

As the field of Multimodal Large Language Models (MLLMs) continues to evolve, their potential to handle mathematical reasoning tasks is promising, as they can handle multimodal questions via cross-modal understanding capabilities compared to text-only LLMs. Current mathematical benchmarks *predominantly focus on evaluating MLLMs' problem-solving ability*, yet there is a crucial gap in addressing more complex scenarios such as error detection, for enhancing reasoning capability in complicated settings. To fill this gap, we formally formulate the new task — **multimodal error detection**, and introduce **ERRORRADAR**, the first benchmark designed to assess MLLMs' capabilities in such a task. ERRORRADAR evaluates two sub-tasks: *error step identification* and *error categorization*, providing a framework for evaluating MLLMs' complex mathematical reasoning ability. It consists of 2,500 high-quality multimodal K-12 mathematical problems, collected from real-world student interactions in an educational organization, with expert-based annotation and metadata such as problem type and error category. Through extensive experiments, we evaluated both open-source and closed-source representative MLLMs, benchmarking their performance against educational expert evaluators. Results indicate challenges still remain, as GPT-4o with best model performance is around 10% behind human evaluation.

1 Introduction

On the path to Artificial General Intelligence, Large Language Models (LLMs) such as GPT-4 (OpenAI, 2023) have emerged as a central focus in both industry and academia (Minaee et al., 2024; Zhao et al., 2023; Zhu et al., 2023). As the real world is inherently multimodal, the evolution of Multimodal Large Language Models (MLLMs) such as the latest GPT-4o (OpenAI, 2024b) and Gemini series (Reid et al., 2024), has become a growing area of interest, demonstrating remarkable effectiveness

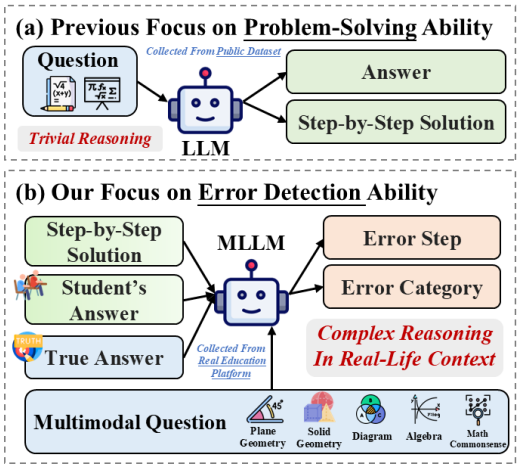


Figure 1: Comparison of research scope and task setting between previous works (a) and our proposed ERRORRADAR benchmark (b) on mathematical reasoning tasks.

in diverse applications (Xiao et al., 2024; He et al., 2024b; Yan et al., 2024b; Hao et al., 2024). In particular, multimodal reasoning stands to benefit education scenarios from the robust capabilities of MLLMs (Wang et al., 2024d; Li et al., 2024a), given its reliance on multimodal inputs to comprehensively grasp users' intentions.

Within the multimodal sphere, mathematical scenarios pose a significant challenge, demanding sophisticated reasoning abilities from MLLMs (Ahn et al., 2024; Yan et al., 2024a). These scenarios have attracted considerable research aimed at pushing the boundaries of MLLMs' reasoning capabilities (Hu et al., 2024b; Jia et al., 2024a; Lu et al., 2024c; Shi et al., 2024b; Zhuang et al., 2024). Besides, various representative benchmarks have been designed to measure MLLMs' performance in complex mathematical reasoning tasks, which involve multi-step reasoning and quantitative analysis within visual contexts (Lu et al., 2024b; Zhang et al., 2024b; Qiao et al., 2024; Peng et al., 2024).

Scrutinizing the off-the-shelf mathematical reasoning benchmarks, there is a predominant focus on evaluating the problem-solving capabilities

Benchmarks	Venue	Modality	Student Ans.	Error Det.
TheoremQA (Chen et al., 2023a)	EMNLP	T	-	-
MathBench (Liu et al., 2024b)	ACL	T	-	-
MR-GSM8K (Zeng et al., 2024b)	ICLR	T	-	-
SciEval (Sun et al., 2024)	AAAI	T	-	-
EIC (Li et al., 2024c)	ACL Finding	T	-	✓
CMMaTH (Li et al., 2024d)	COLING	T, I	-	-
MathScape (Zhou et al., 2024a)	arXiv	T, I	-	-
MATH-V (Wang et al., 2024c)	NeurIPS	T, I	-	-
QRData (Liu et al., 2024c)	ACL	T, I	-	-
IsoBench (Fu et al., 2024)	COLM	T, I	-	-
SciBench (Wang et al., 2024e)	ICML	T, I	-	-
MathVista (Lu et al., 2024b)	ICLR	T, I	-	-
MathVerse (Zhang et al., 2024b)	ECCV	T, I	-	-
ErrorRadar (Ours)	-	T, I	✓	✓

Table 1: Comparison between our proposed ErrorRadar vs. representative LLM-based mathematical reasoning benchmarks. T and I represent text and image. ***Student Ans.*** indicates if the dataset contains real student data (*i.e.*, students' incorrect answers); ***Error Det.*** represents if error detection task is included. See more comparison in Appendix A.

of MLLMs, prioritizing the accuracy with which MLLMs can solve mathematical problems (Wang et al., 2024c; Lu et al., 2024b), as depicted in Figure 1 (a). However, in educational contexts, it is even more crucial to consider user-oriented needs, such as **error detection**. As indicated in Figure 1 (b), this involves not only pinpointing the first incorrect step in a student's step-by-step solution but also categorizing the types of errors made, which is a multifaceted process that requires a deep understanding of both mathematical concepts and cognitive processes (Rabillas et al., 2023).

Towards this end, addressing the aforementioned research gap, we aim to formulate the new task of evaluating MLLMs in the context of error detection scenarios, and therefore introduce the corresponding benchmark termed **ErrorRadar**. We have designed two sub-tasks to comprehensively assess the performance: *error step identification* and *error categorization*. To construct a rich and reliable dataset, we initially sourced a collection of multimodal K-12 level math problems from an educational organization and subsequently refined the dataset through rigorous manual annotation to ensure quality. In particular, we also collect real students' answers for each multimodal question for a relatively robust experimental setting, compared to other relevant benchmarks (See comparisons in Figure 1). Furthermore, we categorized the dataset to better align with diverse needs as follows: **Problem types**: *plane geometry*, *solid geometry*, *diagram*, *algebra*, and *mathematical common sense*; and **Error categories**: *visual perception errors*, *calculation errors*, *reasoning errors*, *knowledge errors*, and *misinterpretation of the problem*. In summary, ErrorRadar comprises 2,500 high-quality instances derived from real-life problem-solving data, providing a foundational dataset to enhance the complex reasoning capabilities of MLLMs for the research community and industry.

For ErrorRadar, we carry out an extensive experimental analysis to determine the proficiency in complex mathematical reasoning of various MLLMs. The evaluation encompasses both the latest open-source MLLMs (*e.g.*, InternVL2 (Chen et al., 2023b), LLaVA-NEXT (Liu et al., 2024a), CogVLM2 (Wang et al., 2023)), and closed-source MLLMs (*e.g.*, GPT4-o (OpenAI, 2024b), Gemini Pro 1.5 (Reid et al., 2024), Claude 3.5 (Anthropic, 2024b)). Our focus was on *their error detection capabilities, specifically the identification of the erroneous step and the classification of the error type.* To establish a comparative human performance standard, we involved expert human educators who possess a graduate-level degree or higher qualifications. The results demonstrate that ErrorRadar, covering cutting-edge topics such as MLLMs' complex reasoning, poses a significant challenge, with human evaluation for two error detection tasks achieving less than 70%.

From in-depth evaluation of representative MLLMs, we obtain the following findings: ❶ Closed-source MLLMs, particularly GPT-4o, consistently outperform open-source MLLMs in both sub-tasks, and show more balanced performance across different error categories; ❷ Weaker MLLMs exhibit an over-reliance on simpler categories, while stronger models handle complex tasks better; ❸ Both MLLMs and humans perform better on error step identification compared to error categorization, as localizing specific errors is inherently simpler than categorizing errors. Our contributions can be summarized as follows[1]:

❶ We take the **first step to formulate the multimodal error detection task**, and introduce a multimodal benchmark termed ErrorRadar. This benchmark serves as a standard operator for assessing the complex mathematical reasoning capabilities of the latest MLLMs.

❷ We meticulously curate an extensive dataset comprising approximately 2,500 high-quality instances with rigorous annotation and rich metadata derived from real user interactions in an educational organization. To the best of our knowledge, this is the first attempt to use real-world student problem-solving data to evaluate MLLMs.

[1]Refer to Appendix B and C for the impact statement and more clarification of our proposed new task setting.

❸ Our comprehensive experimental evaluation of more than 20 MLLMs, both proprietary and open-source, highlight the substantial room for improvement (*i.e.*, 7%-15% in performance) in the complex mathematical reasoning capabilities, underscoring the necessity for further research.

2 Related Work

Benchmarks for Mathematical Reasoning. Recent advancements in mathematical reasoning benchmarks have led to the development of both pure text and multimodal assessments (Lu et al., 2022; Wang et al., 2024c; Zheng et al., 2024; Huo et al., 2024). While datasets like GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021), SuperCLUE-Math (Xu et al., 2024), and MathBench (Liu et al., 2024b) focus on text-based problems, the field has expanded to include multimodal benchmarks that introduce visual elements, pushing the boundaries of AI's mathematical understanding. For instance, MathVista (Lu et al., 2024b) evaluates AI's performance on visual math QA tasks; MATH-V (Wang et al., 2024c) focuses on multimodal mathematical understanding with competition-derived questions; MathVerse (Zhang et al., 2024b) assesses visual diagram comprehension using CoT strategies; CMMU (He et al., 2024c) tests multi-disciplinary, multimodal math understanding with a broad range of Chinese-language questions; MathScape (Zhou et al., 2024a) further advances the field by presenting longer, more complex, and open-ended multimodal problems; and MMMU (Yue et al., 2024) covers college-level knowledge including interleaved mathematical questions. The aforementioned benchmarks assess the mathematical reasoning capabilities of MLLMs by evaluating their problem-solving levels, but they overlook tasks based on the student's perspective, such as error detection. Therefore, we propose the ErrorRadar benchmark, entirely based on real student response data. We discuss more relevant reasoning benchmarks and specific MLLMs in Appendix D.

Multimodal Large Language Models. Generative foundation models such as GPT-4 (OpenAI, 2023), Claude (Anthropic, 2024b), and Gemini (Pal and Sankarasubbu, 2024) have significantly advanced various task solutions without fine-tuning (Cui et al., 2024; Yan and Lee, 2024; Zou et al., 2025; Zhong et al., 2024). Similarly, current open-source MLLMs, built on top of powerful LLMs, have also demonstrated promising potential in multimodal tasks such as image captioning (Yang et al., 2024a) and visual question answering (Fan et al., 2024). For instance, LLaVA-NEXT (Liu et al., 2024a) proposed projecting visual embeddings, extracted by a pretrained vision encoder, into the word space through a single MLP layer, where LLMs like LLaMA, Vicuna, and Mistral are fine-tuned to understand these post-projection tokens. In a similar fashion, Phi3 (Abdin et al., 2024), DeepSeek-VL (Lu et al., 2024a), MiniCPM-V (Yao et al., 2024), ChatGLM (GLM et al., 2024), CogVLM (Wang et al., 2023), Intern-VL (Chen et al., 2023b), Qwen-VL (Bai et al., 2023) and Yi-VL (Young et al., 2024) also utilize a projector (or adapter, shared compression layer, *etc.*) to align the visual embeddings extracted from a vision encoder with text embeddings, which are then concatenated and fed into LLM. Therefore, we propose ErrorRadar, a benchmark on a fine-grained evaluation of MLLMs' ability to detect errors based on students' answers and reasoning steps.

3 The ErrorRadar Dataset

3.1 Task Formulation

Basic Setting. In this task, we assess the model's ability to detect errors in mathematical problem-solving processes across multiple samples. Let N denote the total number of samples in the evaluation set. For each sample $i \in \{1, 2, \ldots, N\}$, the input set $\mathcal{I}_i$ is defined as:

$$\mathcal{I}_i = \{Q_{\text{text},i}, Q_{\text{image},i}, A_{\text{correct},i}, A_{\text{incorrect},i}, \{S_{k,i}\}_{k=1}^{n_i}\},$$

where $Q_{\text{text},i}$ represents the textual statement of the i-th problem, $Q_{\text{image},i}$ represents the image representation of the i-th problem, $A_{\text{correct},i}$ represents the correct solution for the i-th problem, $A_{\text{incorrect},i}$ represents the incorrect student solution for the i-th problem, and $\{S_{k,i}\}_{k=1}^{n_i}$ denotes the sequence of n_i steps in the i-th problem-solving process, with each $S_{k,i}$ representing a distinct step.

Subtask 1: Error Step Identification. The task is to identify the index x of the first incorrect step in the sequence $\{S_{k,i}\}_{k=1}^{n_i}$. The function $f_{\text{step},i}$ maps the input $\mathcal{I}_i$ to the index of the erroneous step:

$$f_{\text{step},i} : \mathcal{I}_i \rightarrow x_i,$$

$$\text{where } x_i = \arg\min_k \{S_{k,i} \text{ is incorrect}\}.$$

Subtask 2: Error Categorization. The task is to classify the type of error for the i-th problem into one of the following categories:

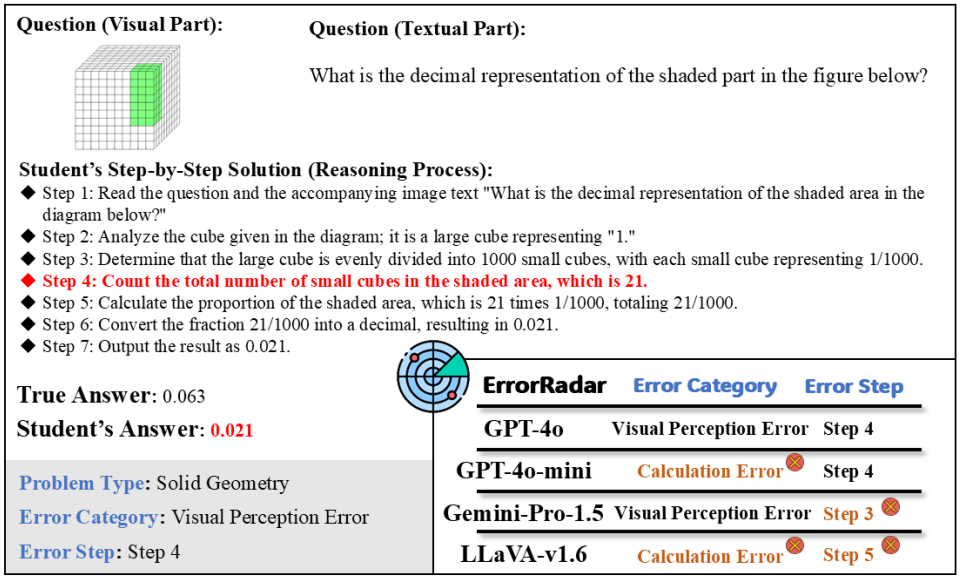


Figure 2: Example of our annotated multimodal mathematical reasoning dataset ERRORRADAR, and performance comparison on error categorization and error step localization tasks among representative MLLMs. It is evident even simple math problems can be mishandled by superior MLLMs in one or both tasks, highlighting the challenging nature of multimodal error detection.

$\{\text{VIS}, \text{CAL}, \text{REAS}, \text{KNOW}, \text{MIS}\}$. The error categorization function $f_{\text{error},i}$ maps the input $\mathcal{I}_i$ to the error category $C_{\text{error},i}$:

$$f_{\text{error},i} : \mathcal{I}_i \rightarrow C_{\text{error},i}.$$

More concrete examples can be seen in Figure 2 and Appendix E. The discrepancies within the five error categories are delineated as follows:

- ✯ **Visual Perception Errors (VIS)**: These errors arise when there is a failure to accurately interpret the information contained in images or diagrams presented in the question due to visual issues.
- ✯ **Calculation Error (CAL)**: These errors manifest during the calculation process, which may include arithmetic mistakes such as incorrect addition, subtraction, multiplication, or division, errors in unit conversion, or mistakes in the numerical signs between multiple steps.
- ✯ **Reasoning Error (REAS)**: These errors occur during problem-solving process when improper reasoning is applied, leading to incorrect application of logical relationships or conclusions.
- ✯ **Knowledge Error (KNOW)**: These errors result from incomplete or incorrect understanding of the knowledge base, leading to mistakes when applying relevant knowledge points.
- ✯ **Misinterpretation of the Question (MIS)**: These errors occur when there is a failure to correctly understand the requirements of the question or a misinterpretation of the question's intent, leading to responses that are irrelevant to the question's demands.

Performance Metrics. The evaluation of both subtasks is conducted separately:

- **Error Step Identification Performance.** Let $G_{\text{step},i}$ be the ground truth index of the first incorrect step for the i-th sample. We report the accuracy for this subtask:

$$\text{Acc}_{\text{step}} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(x_i = G_{\text{step},i}),$$

  where $\mathbb{I}(\cdot)$ is indicator function, returning 1 if prediction matches ground truth, and 0 otherwise.

- **Error Categorization Performance.** We report Precision, Recall, and F1-score for each error category, alongside their macro-averaged counterparts as overall performance metrics.

3.2 Data Source & Annotation

Following the roadmap shown in Figure 3, this section includes how we collect and annotate ERRORRADAR dataset to ensure the overall data quality. Different from the conventional benchmarks that rely on public datasets or modified textbook collections (Lu et al., 2024b; Zhou et al., 2024a), ERRORRADAR dataset is uniquely sourced from the question bank of a global educational organization. This repository encompasses a vast array

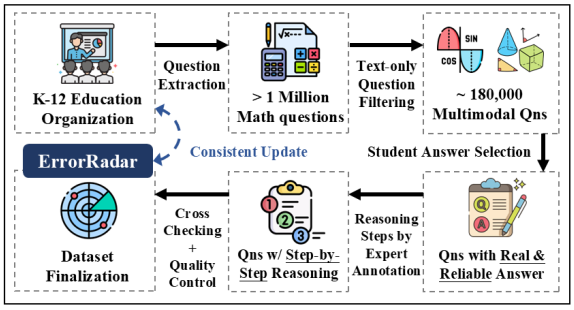


Figure 3: Roadmap of ERRORRADAR dataset collection, filtering, annotation, and consistent update.

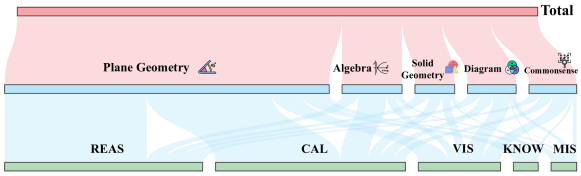


Figure 4: Dataset distribution of ERRORRADAR with respect to problem type and error category.

Statistic	Number
Total multimodal questions	2,500
Problem Type	
- Plane Geometry	1559 (62.4%)
- Solid Geometry	191 (7.6%)
- Diagram	233 (9.3%)
- Algebra	288 (11.5%)
- Math Commonsense	229 (9.2%)
Error Category	
- Visual Perception Error	395 (15.8%)
- Calculation Error	912 (36.5%)
- Reasoning Error	951 (38.0%)
- Knowledge Error	119 (4.8%)
- Misinterpretation of the Qns	123 (4.9%)
Average Reasoning Step	7.6
Maximum Reasoning Step	20
Minimum Reasoning Step	3
Average Question Length	168
Maximum Question Length	719
Minimum Question Length	13

Figure 5: Key statistics of ERRORRADAR.

of mathematical problems in K-12 levels, totaling over a million entries. Initially, we curated approximately 180,000 math problems with a single-image setting, aligning with multimodal setup.

Subsequently, we refined our selection by evaluating the universality and articulation of the problem content. For each problem, we identified multiple incorrect answers, and finally selected the most frequently given incorrect answer as the student's response. Additionally, we scrutinized cases where the most common incorrect answer was due to system input errors despite the answer being correct. In such instances, we amended the dataset by incorporating the next most frequently incorrect answer.

Furthermore, since error detection tasks necessitate a step-by-step reasoning process, we enriched our dataset with new content through manual annotation. Specifically, we provided professional annotators with the original multimodal QA data, student's incorrect answers, and the pedagogical team's analysis of correct answer process. Based on initial data, annotators delineated the erroneous steps leading to the incorrect answers (More details of annotation and inconsistent case handling in Appendix F.1 and F.2).

Our team of annotators, consisting of around ten educational experts with domain expertise, conducted two rounds of cross-checking to ensure the reliability of the annotations. In cases of inconsistency, the related data were presented to the annotation lead for final adjudication. The annotators' results were subject to review and quality control by the educational organization from which the data originated, ensuring *security*, *reliability*, and *consistent updates*.

3.3 Dataset Details

As illustrated in Figure 5, ERRORRADAR dataset comprises a substantial collection of 2,500 multimodal math questions designed for error detection tasks. It predominantly includes plane geometry problems, with solid geometry, diagram, algebra, and math commonsense questions making up the remainder, highlighting its focus on diverse mathematical problems. It also categorizes errors into visual perception, calculation, reasoning, knowledge, and question misinterpretation. Key statistics indicate a diverse dataset with an average reasoning step of 7.6, a variety of question lengths, and a wide range of reasoning steps. Detailed distribution of ERRORRADAR, problem type definition, and error category formulation can be seen in Figure 4, Appendix F.3 and F.4.

4 Experiments and Analysis

4.1 Evaluation Protocols

In ERRORRADAR benchmark, we propose an evaluation strategy using template matching rules. The evaluation process consists of three stages: *response generation*, *answer extraction*, and *performance calculation*. Initially, the MLLMs generate responses given the inputs, which incorporates the multimodal mathematical question, wrong answer, and its step-by-step reasoning, using the template from Appendix G.3. Subsequently, the short answer text can be extracted from the detailed response. Finally, the model performance is based on the detailed score calculation as shown in Section

3.1. The final score will be calculated by averaging the scores from three rounds of assessment.

4.2 Experimental Setup

In our experimental setup, we meticulously categorized and evaluated a diverse array of MLLMs into three distinct groups to assess their capabilities across error detection tasks. (i) The **Open-Source MLLMs** category encompassed models such as InternVL-2 (Chen et al., 2023b), Phi-3-vision (Abdin et al., 2024), Yi-VL (Young et al., 2024), DeepSeek-VL (Lu et al., 2024a), LLaVA-v1.6-Vicuna (Liu et al., 2024a), MiniCPM-LLaMA3-V2.5 (Yao et al., 2024), MiniCPM-V2.6 (Yao et al., 2024), Qwen-VL (Bai et al., 2023), GLM-4v (GLM et al., 2024), and LLaVA-NEXT (Liu et al., 2024a), each demonstrating their unique strengths and capabilities in handling different types of errors. (ii) The **Closed-Source MLLMs** featured proprietary models like Qwen-VL-Max (Bai et al., 2023), Claude-3-Haiku (Anthropic, 2024a), Claude-3.5-Sonnet (Anthropic, 2024b), Gemini-Pro-1.5 (Reid et al., 2024), GPT-4o-mini (OpenAI, 2024a), and GPT-4o (OpenAI, 2024b), providing a comparison point for the performance of models that are not publicly accessible. (iii) Lastly, the **Human Performance** category served as a benchmark for natural intelligence, allowing us to gauge how closely MLLMs can emulate human cognitive functions across tasks such as visual perception (More details in Appendix G.2). Prompts for MLLMs and sources of MLLMs are in Appendix G.3 and G.4, respectively.

4.3 Experimental Results

4.3.1 Main Results

Finding #1: Closed-source MLLMs generally outperform open-source MLLMs in both error detection tasks, with GPT-4o demonstrating the strongest performance. Figures 6 and 7 show that closed-source MLLMs generally outperform open-source MLLMs in both STEP and CATE tasks, and they also exhibit relatively more balanced performance across the five error categories. This superiority can likely be attributed to the proprietary datasets and advanced training resources available to closed-source models, which allow for more robust fine-tuning (Shi et al., 2023; Yu et al., 2024). Notably, GPT-4o stands out as the best model, achieving highest scores not only in STEP and CATE tasks, demonstrating its overall versatility. Given the performance gap, open-source MLLMs can further enhance themselves by distilling error detection capabilities of closed-source ones (Hsieh et al., 2023). See more actionable suggestions in Appendix G.5.

Finding #2: Open-source MLLMs tend to predict CAL category, leading to unusually high recall. Figure 8 indicates that MLLMs with relatively low performance in the CATE task tend to exhibit unusually high recall in the CAL category. Specifically, open-source models like MiniCPM-LLaMA3-v2.5 even achieve a 100% recall in CAL, while Phi-3-vision and InternVL-2-8B reach 99.6%. Upon analyzing the category prediction proportions of CAL from Figure 9 (See details of all MLLMs in Appendix G.6), it becomes clear that open-source MLLMs with the top five CAL recall predict over 80% of instances as CAL category, suggesting an over-reliance on this category. In contrast, closed-source MLLMs with top-five CAL recall do not exhibit this extreme trend of prediction bias. This phenomenon likely arises from weaker MLLMs attempting to overfit on the CAL category, a relatively simpler classification, to compensate for their inability to handle more complex scenarios (Tirumala et al., 2022; Xu et al., 2021). Models exhibiting this phenomenon can assign different weights to samples of different categories during training to reduce the model's preference for a particular category. This can be achieved by adjusting the weights in the loss (*e.g.*, Focal Loss & AdaFocal) (Li et al., 2022; Ghosh et al., 2022).

Finding #3: MLLMs with strong overall performance tend to handle STEP easier than CATE. From Figures 6 and 7, the best open-source MLLMs, such as InternVL2-76B, and the best closed-source MLLMs, like GPT-4o, exhibit a tendency where their STEP performance surpasses that of CATE. This trend holds even for human performance, where accuracy on STEP is higher (69.8%) compared to recall on CATE (60.7%). The reason for this disparity is likely that identifying the error step is inherently easier, as it involves localizing a specific point of failure. On the other hand, categorizing the error requires more complex reasoning and contextual understanding to classify the nature of the error, which adds difficulty. This mirrors the settings in object detection, where localization (*i.e.,* predicting where an object is) is relatively simpler than classification (*i.e.,* predicting what an object is) (Zou et al., 2023; Jiao et al., 2021). To improve the performance of error categorization tasks, MLLMs need to better understand the relationship between the problem itself and the

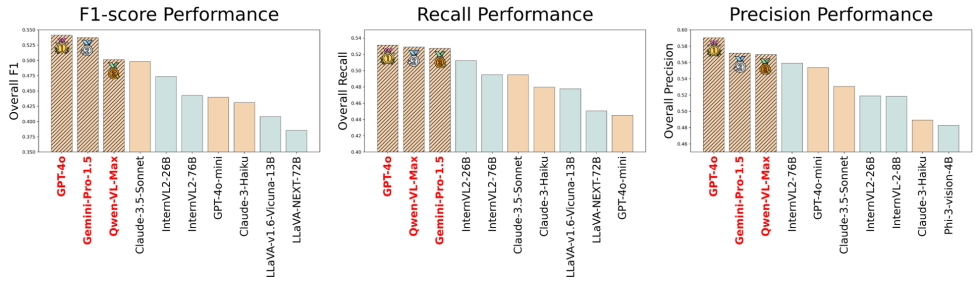


Figure 6: Error category performance of top 10 MLLMs for F1, recall, and precision, respectively. The orange bars represent closed-source MLLMs, while the blue ones represent open-source MLLMs. The masked bars represent the top 3. Due to page limit, we leave the bar charts of all models' performance in Appendix G.1.

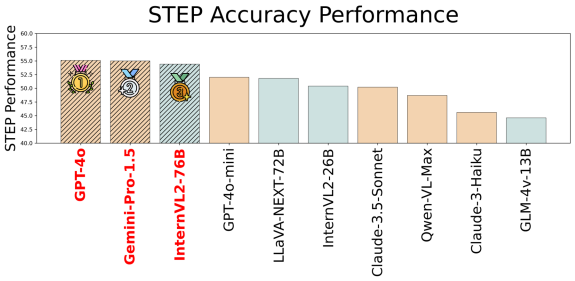


Figure 7: Error step performance of top 10 MLLMs. The orange bars represent closed-source MLLMs, while the blue ones represent open-source MLLMs. The masked bars represent the top 3. We leave the bar charts of all models' performance in Appendix G.1.

steps where errors occur. Thus, modeling this part of the relationship can be a focus in the design of training data (Shi et al., 2024a; Song et al., 2025).

Finding #4: CAL is the easiest category for MLLMs, while KNOW is the most difficult. CAL is the category with the highest F1 performance among most MLLMs, which could be attributed to the structured and deterministic nature of calculations, where errors often result in clear, quantifiable deviations from expected outcomes, making them more straightforward to detect (Lewkowycz et al., 2022; Kojima et al., 2022). Conversely, KNOW stands out as the most challenging category, suggesting that MLLMs struggle significantly with tasks requiring deep factual understanding and contextual reasoning. The complexity of knowledge errors likely stems from the need for comprehensive domain expertise, which current MLLMs may not fully encapsulate yet.

Finding #5: MLLMs still have a gap to close to reach human-level intelligence in error detection. Human performance significantly outperforms the best MLLMs in both the STEP and CATE tasks, with overall performance of 69.8% and 60.7% respectively, compared to the highest MLLM scores of 55.1% and 53.1%. Notably, the detection of VIS by humans is markedly superior to the best MLLMs, with a difference of nearly 20%. This substantial lead may be attributed to the sophisticated pattern recognition inherent to human visual processing (Doerig et al., 2022), which MLLMs, despite their advancements, have yet to fully emulate. Besides, it is interesting to note that human performance in REAS detection is lower than all closed-source MLLMs but higher than almost all open-source MLLMs.

Finding #6: Specialized multimodal reasoning and math models underperform generalist models like GPT-4o. As illustrated in Figure 10, we evaluate QVQ (Team, 2024) (a reasoning-enhanced variant of Qwen2-VL-72B) and find that it still lags behind GPT-4o. This suggests that specialized reasoning training alone does not guarantee superior performance in our task, which requires fine-grained multimodal error analysis beyond pure logical deduction. We also test three math-focused models (Zhang et al., 2024c; Gao et al., 2023; Shi et al., 2024c) to investigate whether problem-solving ability generalizes to our task. Their performance was weaker than general models, with G-LLaVA showing the lowest scores. We attribute this to its narrow geometric-focused training, which lacks diversity for our error categories.

4.3.2 Visual Perception Analysis

Due to the space limit, we discuss the visual perception analysis in Appendix G.7, and further analyze misclassification for each category and visual perception case study in Appendix G.8 and G.9.

4.3.3 Relation between Error Category and Error Step

Due to the space limit, we discuss the relation between error category and error step in Appendix G.10, and further analyze the phenomenon via cognitive load analysis in Appendix G.11.

4.3.4 Scaling Analysis

Finding #1: The performance of MLLMs on STEP task increases with the scale of parameters. We observe a phenomenon similar to the

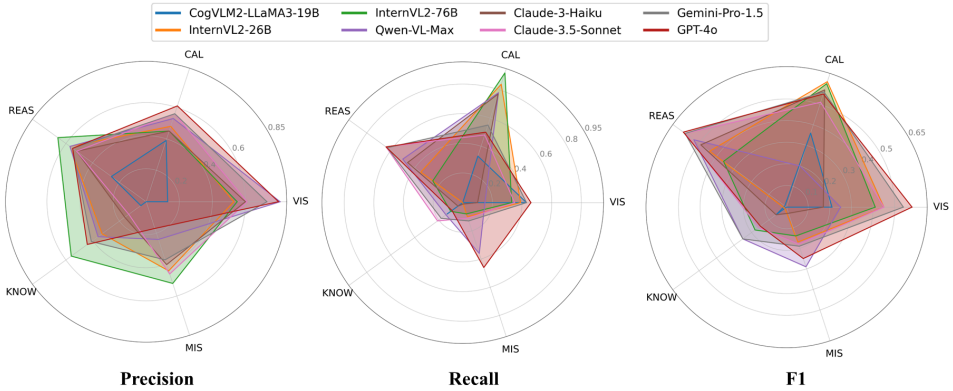


Figure 8: The radar charts of error category performance for the top eight MLLMs (each dimension indicates an error category). Considering visualization clarity, we leave the detailed values of all models in Appendix G.1.

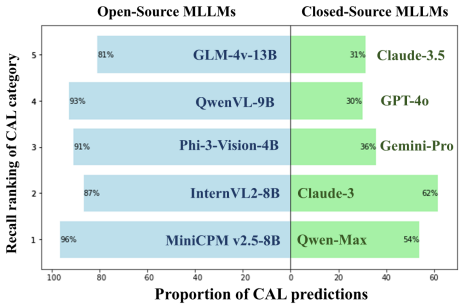


Figure 9: The proportion of CAL predictions of respective representative closed-source and open-source MLLMs.

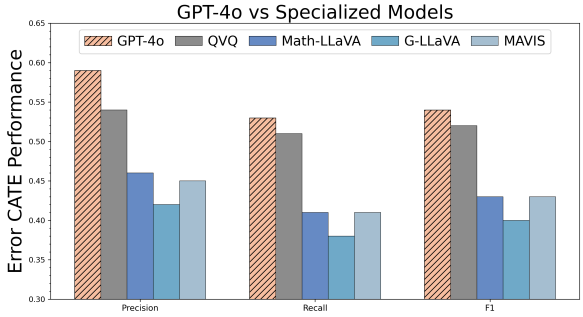


Figure 10: Performance comparison between GPT-4o vs multimodal reasoning-/math-specialized models.

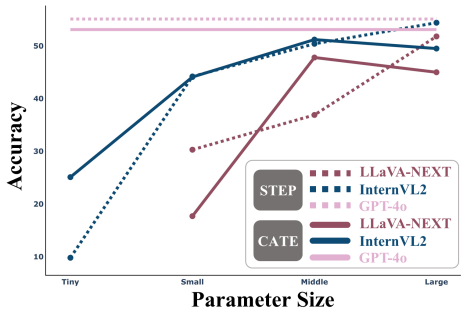


Figure 11: The accuracy of **STEP** and **CATE** of two representative MLLM series: LLaVA-NEXT and InternVL2. We denote *Tiny*, *Small*, *Middle*, *Large* as the 2B, 8B, 26B, 76B for InternVL2 and None, 7B, 13B, 72B for LLaVA-NEXT.

scaling law (Kaplan et al., 2020) in our experiments. As shown in Figure 11, when the size of the InternVL2 model increases from Tiny to Huge, the accuracy of STEP task rises from 9.8% to 54.4%, showing an improvement of 44.6%. Similarly, as the size of LLaVA-NEXT increases from Small to Large, its STEP accuracy also improves from 30.3% to 51.8%, indicating larger MLLMs exhibit greater reasoning ability in localizing error steps.

Finding #2: CATE task is relatively more difficult to improve through scaling. While the accuracy of CATE shows a trend of improvement for both the InternVL2 and LLaVA-NEXT models as their size increases from Tiny (Small) to Middle, a slight decrease is also observed when model size reaches Large. We presume this is because CATE is a more challenging task compared to STEP, and merely increasing the model size without fine-tuning is insufficient for sustained improvement and may even introduce bias (Aghajanyan et al., 2023; Muennighoff et al., 2024).

5 Conclusion

In conclusion, we introduce ErrorRadar, the first multimodal benchmark designed specifically for evaluating MLLMs's reasoning in mathematical error detection scenarios. By focusing on both *error step identification* and *error categorization*, ErrorRadar bridges a critical research gap in assessing MLLMs' capabilities in complex mathematical reasoning. The dataset's construction, based on real-world student interactions, ensures a robust evaluation framework that reflects genuine user needs. Extensive experimental analysis, comparing leading open-source and proprietary MLLMs, reveals significant challenges in error detection, highlighting the need for continued advancements in the multimodal reasoning domain towards Artificial General Intelligence.

Limitations

While ERRORRADAR provides a novel benchmark for multimodal mathematical error detection, we acknowledge certain limitations that also pave the way for future research:

- **Dataset Scale:** Although our dataset of 2,500 instances is substantial for an initial benchmark and sourced from real-world interactions, the vast domain of K-12 mathematics encompasses an even wider array of problem types, visual representations, and nuanced student errors. Future work will focus on continuously expanding ERRORRADAR with more instances and greater diversity in problem structures and visual elements to ensure broader coverage.
- **Scope of Evaluated MLLMs:** Our study benchmarked a comprehensive set of over 20 contemporary MLLMs. Nevertheless, the field of MLLMs is evolving at an extremely rapid pace, with new architectures and larger models being released frequently. We aim to periodically update our benchmark results by evaluating new state-of-the-art MLLMs as they become available, ensuring ERRORRADAR remains a relevant and current tool.
- **Static Error Evaluation:** The benchmark assesses MLLMs on their ability to identify and categorize errors in pre-existing, static student solutions. It does not currently evaluate the models' interactive capabilities, such as guiding a student through error correction or generating pedagogical explanations for the identified errors. We envision future research building upon ERRORRADAR to explore these more dynamic and interactive educational applications of MLLMs.

References

Amine Abbad-Andaloussi, Andrea Burattin, Tijs Slaats, Ekkart Kindler, and Barbara Weber. 2023. Complexity in declarative process models: Metrics and multi-modal assessment of cognitive load. *Expert Systems with Applications*, 233:120924.

Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, and 1 others. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Armen Aghajanyan, Lili Yu, Alexis Conneau, Wei-Ning Hsu, Karen Hambardzumyan, Susan Zhang, Stephen Roller, Naman Goyal, Omer Levy, and Luke Zettlemoyer. 2023. Scaling laws for generative mixed-modal language models. In *International Conference on Machine Learning*, pages 265–279. PMLR.

Janice Ahn, Rishu Verma, Renze Lou, Di Liu, Rui Zhang, and Wenpeng Yin. 2024. Large language models for mathematical reasoning: Progresses and challenges. *arXiv preprint arXiv:2402.00157*.

Aida Amini, Saadia Gabriel, Peter Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. *arXiv preprint arXiv:1905.13319*.

Anthropic. 2024a. Claude 3.

Anthropic. 2024b. Claude 3.5.

Muhammad Haseeb Aslam, Marco Pedersoli, Alessandro Lameiras Koerich, and Eric Granger. 2024. Multi teacher privileged knowledge distillation for multimodal expression recognition. *arXiv preprint arXiv:2408.09035*.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Jing Bi, Susan Liang, Xiaofei Zhou, Pinxin Liu, Junjia Guo, Yunlong Tang, Luchuan Song, Chao Huang, Guangyu Sun, Jinxi He, and 1 others. 2025. Why reasoning matters? a survey of advancements in multimodal reasoning (v1). *arXiv preprint arXiv:2504.03151*.

Marcel Binz and Eric Schulz. 2023. Turning large language models into cognitive models. *arXiv preprint arXiv:2306.03917*.

Jiaqi Chen, Jianheng Tang, Jinghui Qin, Xiaodan Liang, Lingbo Liu, Eric P Xing, and Liang Lin. 2021. Geoqa: A geometric question answering benchmark towards multimodal numerical reasoning. *arXiv preprint arXiv:2105.14517*.

Shijie Chen, Yu Zhang, and Qiang Yang. 2024. Multi-task learning in natural language processing: An overview. *ACM Computing Surveys*, 56(12):1–32.

Wenhu Chen, Ming Yin, Max Ku, Pan Lu, Yixin Wan, Xueguang Ma, Jianyu Xu, Xinyi Wang, and Tony Xia. 2023a. Theoremqa: A theorem-driven question answering dataset. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7889–7901.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. 2023b. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv preprint arXiv:2312.14238*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Can Cui, Yunsheng Ma, Xu Cao, Wenqian Ye, Yang Zhou, Kaizhao Liang, Jintai Chen, Juanwu Lu, Zichong Yang, Kuei-Da Liao, and 1 others. 2024. A survey on multimodal large language models for autonomous driving. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 958–979.

Yihe Deng, Pan Lu, Fan Yin, Ziniu Hu, Sheng Shen, Quanquan Gu, James Y Zou, Kai-Wei Chang, and Wei Wang. 2024. Enhancing large vision language models with self-training on image comprehension. *Advances in Neural Information Processing Systems*, 37:131369–131397.

Adrien Doerig, Tim C Kietzmann, Emily Allen, Yihan Wu, Thomas Naselaris, Kendrick Kay, and Ian Charest. 2022. Visual representations in the human brain are aligned with large language models. *arXiv preprint arXiv:2209.11737*.

Yue Fan, Jing Gu, Kaiwen Zhou, Qianqi Yan, Shan Jiang, Ching-Chen Kuo, Yang Zhao, Xinze Guan, and Xin Wang. 2024. Muffin or chihuahua? challenging multimodal large language models with multipanel vqa. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6845–6863.

Deqing Fu, Ghazal Khalighinejad, Ollie Liu, Bhuwan Dhingra, Dani Yogatama, Robin Jia, and Willie Neiswanger. 2024. Isobench: Benchmarking multimodal foundation models on isomorphic representations. *arXiv preprint arXiv:2404.01266*.

Bofei Gao, Feifan Song, Zhe Yang, Zefan Cai, Yibo Miao, Qingxiu Dong, Lei Li, Chenghao Ma, Liang Chen, Runxin Xu, and 1 others. 2024. Omni-math: A universal olympiad level mathematic benchmark for large language models. *arXiv preprint arXiv:2410.07985*.

Jiahui Gao, Renjie Pi, Jipeng Zhang, Jiacheng Ye, Wanjun Zhong, Yufei Wang, Lanqing Hong, Jianhua Han, Hang Xu, Zhenguo Li, and 1 others. 2023. G-llava: Solving geometric problem with multi-modal large language model. *arXiv preprint arXiv:2312.11370*.

Arindam Ghosh, Thomas Schaaf, and Matthew Gormley. 2022. Adafocal: Calibration-aware adaptive focal loss. *Advances in Neural Information Processing Systems*, 35:1583–1595.

Kushankur Ghosh, Colin Bellinger, Roberto Corizzo, Paula Branco, Bartosz Krawczyk, and Nathalie Japkowicz. 2024. The class imbalance problem in deep learning. *Machine Learning*, 113(7):4845–4901.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, and 1 others. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *arXiv preprint arXiv:2406.12793*.

Xiaotian Han, Yiren Jian, Xuefeng Hu, Haogeng Liu, Yiqi Wang, Qihang Fan, Yuang Ai, Huaibo Huang, Ran He, Zhenheng Yang, and 1 others. 2024. Infimm-webmath-40b: Advancing multimodal pre-training for enhanced mathematical reasoning. *arXiv preprint arXiv:2409.12568*.

Xixuan Hao, Wei Chen, Yibo Yan, Siru Zhong, Kun Wang, Qingsong Wen, and Yuxuan Liang. 2024. Urbanvlp: A multi-granularity vision-language pre-trained foundation model for urban indicator prediction. *arXiv preprint arXiv:2403.16831*.

Yunzhuo Hao, Jiawei Gu, Huichen Will Wang, Linjie Li, Zhengyuan Yang, Lijuan Wang, and Yu Cheng. 2025. Can mllms reason in multimodality? emma: An enhanced multimodal reasoning benchmark. *arXiv preprint arXiv:2501.05444*.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, and 1 others. 2024a. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3828–3850.

Jinlong He, Pengfei Li, Gang Liu, Zixu Zhao, and Shenjun Zhong. 2024b. Pefomed: Parameter efficient fine-tuning on multimodal large language models for medical visual question answering. *arXiv preprint arXiv:2401.02797*.

Zheqi He, Xinya Wu, Pengfei Zhou, Richeng Xuan, Guang Liu, Xi Yang, Qiannan Zhu, and Hua Huang. 2024c. Cmmu: A benchmark for chinese multimodal multi-type question understanding and reasoning. *arXiv preprint arXiv:2401.14011*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. *arXiv preprint arXiv:2305.02301*.

Haigen Hu, Xiaoyuan Wang, Yan Zhang, Qi Chen, and Qiu Guan. 2024a. A comprehensive survey on contrastive learning. *Neurocomputing*, page 128645.

Yushi Hu, Weijia Shi, Xingyu Fu, Dan Roth, Mari Ostendorf, Luke Zettlemoyer, Noah A Smith, and Ranjay Krishna. 2024b. Visual sketchpad: Sketching as a visual chain of thought for multimodal language models. *arXiv preprint arXiv:2406.09403*.

Xuhan Huang, Qingning Shen, Yan Hu, Anningzhe Gao, and Benyou Wang. 2024. Mamo: a mathematical modeling benchmark with solvers. *arXiv preprint arXiv:2405.13144*.

Jiahao Huo, Yibo Yan, Boren Hu, Yutao Yue, and Xuming Hu. 2024. Mmneuron: Discovering neuron-level domain-specific interpretation in multimodal large language model. *arXiv preprint arXiv:2406.11193*.

Farkhund Iqbal, Ahmed Abbasi, Abdul Rehman Javed, Ahmad Almadhor, Zunera Jalil, Sajid Anwar, and Imad Rida. 2024. Data augmentation-based novel deep learning method for deepfaked images detection. *ACM Transactions on Multimedia Computing, Communications and Applications*, 20(11):1–15.

Mengzhao Jia, Zhihan Zhang, Wenhao Yu, Fangkai Jiao, and Meng Jiang. 2024a. Describe-then-reason: Improving multimodal mathematical reasoning through visual comprehension training. *arXiv preprint arXiv:2404.14604*.

Mengzhao Jia, Zhihan Zhang, Wenhao Yu, Fangkai Jiao, and Meng Jiang. 2024b. Describe-then-reason: Improving multimodal mathematical reasoning through visual comprehension training. *arXiv preprint arXiv:2404.14604*.

Licheng Jiao, Ruohan Zhang, Fang Liu, Shuyuan Yang, Biao Hou, Lingling Li, and Xu Tang. 2021. New generation deep learning for video object detection: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 33(8):3195–3215.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Mehran Kazemi, Hamidreza Alvari, Ankit Anand, Jialin Wu, Xi Chen, and Radu Soricut. 2023. Geomverse: A systematic evaluation of large models for geometric reasoning. *arXiv preprint arXiv:2312.12241*.

Michael J Kennedy and John Elwood Romig. 2024. Cognitive load theory: An applied reintroduction for special and general educators. *TEACHING Exceptional Children*, 56(6):440–451.

Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. 2023. Understanding the effects of rlhf on llm generalisation and diversity. *arXiv preprint arXiv:2310.06452*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Eldar Kurtic, Amir Moeini, and Dan Alistarh. 2024. Mathador-lm: A dynamic benchmark for mathematical reasoning on large language models. *arXiv preprint arXiv:2406.12572*.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, and 1 others. 2022. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems*, 35:3843–3857.

Chengpeng Li, Zheng Yuan, Hongyi Yuan, Guanting Dong, Keming Lu, Jiancan Wu, Chuanqi Tan, Xiang Wang, and Chang Zhou. 2023. Query and response augmentation cannot help out-of-domain math reasoning generalization.

Hang Li, Tianlong Xu, Chaoli Zhang, Eason Chen, Jing Liang, Xing Fan, Haoyang Li, Jiliang Tang, and Qingsong Wen. 2024a. Bringing generative ai to adaptive learning in education. *arXiv preprint arXiv:2402.14601*.

Qintong Li, Leyang Cui, Xueliang Zhao, Lingpeng Kong, and Wei Bi. 2024b. Gsm-plus: A comprehensive benchmark for evaluating the robustness of llms as mathematical problem solvers. *arXiv preprint arXiv:2402.19255*.

Xiang Li, Chengqi Lv, Wenhai Wang, Gang Li, Lingfeng Yang, and Jian Yang. 2022. Generalized focal loss: Towards efficient representation learning for dense object detection. *IEEE transactions on pattern analysis and machine intelligence*, 45(3):3139–3153.

Xiaoyuan Li, Wenjie Wang, Moxin Li, Junrong Guo, Yang Zhang, and Fuli Feng. 2024c. Evaluating mathematical reasoning of large language models: A focus on error identification and correction. *arXiv preprint arXiv:2406.00755*.

Zhong-Zhi Li, Ming-Liang Zhang, Fei Yin, Zhi-Long Ji, Jin-Feng Bai, Zhen-Ru Pan, Fan-Hu Zeng, Jian Xu, Jia-Xin Zhang, and Cheng-Lin Liu. 2024d. Cmmath: A chinese multi-modal math skill evaluation benchmark for foundation models. *arXiv preprint arXiv:2407.12023*.

Chen Liang, Jiahui Yu, Ming-Hsuan Yang, Matthew Brown, Yin Cui, Tuo Zhao, Boqing Gong, and Tianyi Zhou. 2024a. Module-wise adaptive distillation for multimodality foundation models. *Advances in Neural Information Processing Systems*, 36.

Zhenwen Liang, Dian Yu, Wenhao Yu, Wenlin Yao, Zhihan Zhang, Xiangliang Zhang, and Dong Yu. 2024b. Mathchat: Benchmarking mathematical reasoning and instruction following in multi-turn interactions. *arXiv preprint arXiv:2405.19444*.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024a. Llava-next: Improved reasoning, ocr, and world knowledge.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.

Hongwei Liu, Zilong Zheng, Yuxuan Qiao, Haodong Duan, Zhiwei Fei, Fengzhe Zhou, Wenwei Zhang, Songyang Zhang, Dahua Lin, and Kai Chen. 2024b. Mathbench: Evaluating the theory and application proficiency of llms with a hierarchical mathematics benchmark. *arXiv preprint arXiv:2405.12209*.

Wentao Liu, Qianjun Pan, Yi Zhang, Zhuo Liu, Ji Wu, Jie Zhou, Aimin Zhou, Qin Chen, Bo Jiang, and Liang He. 2025. Cmm-math: A chinese multimodal math dataset to evaluate and enhance the mathematics reasoning of large multimodal models. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 12585–12591.

Xiao Liu, Zirui Wu, Xueqing Wu, Pan Lu, Kai-Wei Chang, and Yansong Feng. 2024c. Are llms capable of data-based statistical and causal reasoning? benchmarking advanced quantitative reasoning with data. *arXiv preprint arXiv:2402.17644*.

Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Yaofeng Sun, and 1 others. 2024a. Deepseek-vl: towards real-world vision-language understanding. *arXiv preprint arXiv:2403.05525*.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2024b. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*.

Pan Lu, Baolin Peng, Hao Cheng, Michel Galley, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, and Jianfeng Gao. 2024c. Chameleon: Plug-and-play compositional reasoning with large language models. *Advances in Neural Information Processing Systems*, 36.

Pan Lu, Liang Qiu, Wenhao Yu, Sean Welleck, and Kai-Wei Chang. 2022. A survey of deep learning for mathematical reasoning. *arXiv preprint arXiv:2212.10535*.

Huan Ma, Changqing Zhang, Yatao Bian, Lemao Liu, Zhirui Zhang, Peilin Zhao, Shu Zhang, Huazhu Fu, Qinghua Hu, and Bingzhe Wu. 2023a. Fairness-guided few-shot prompting for large language models. *Advances in Neural Information Processing Systems*, 36:43136–43155.

Yubo Ma, Yixin Cao, YongChing Hong, and Aixin Sun. 2023b. Large language model is not a good few-shot information extractor, but a good reranker for hard samples! *arXiv preprint arXiv:2303.08559*.

Yujun Mao, Yoon Kim, and Yilun Zhou. 2024. Champ: A competition-level dataset for fine-grained analyses of llms' mathematical reasoning capabilities. *arXiv preprint arXiv:2401.06961*.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2024. Large language models: A survey. *arXiv preprint arXiv:2402.06196*.

Swaroop Mishra, Matthew Finlayson, Pan Lu, Leonard Tang, Sean Welleck, Chitta Baral, Tanmay Rajpurohit, Oyvind Tafjord, Ashish Sabharwal, Peter Clark, and 1 others. 2022. Lila: A unified benchmark for mathematical reasoning. *arXiv preprint arXiv:2210.17517*.

Niklas Muennighoff, Alexander Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin A Raffel. 2024. Scaling data-constrained language models. *Advances in Neural Information Processing Systems*, 36.

OpenAI. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

OpenAI. 2024a. Gpt-4o mini: advancing cost-efficient intelligence.

OpenAI. 2024b. GPT-4o system card.

Fred Paas, Alexander Renkl, and John Sweller. 2010. Cognitive load theory and instructional design: Recent developments. *Educational Psychologist*.

Ankit Pal and Malaikannan Sankarasubbu. 2024. Gemini goes to med school: exploring the capabilities of multimodal large language models on medical challenge problems & hallucinations. *arXiv preprint arXiv:2402.07023*.

Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. 2024. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*.

Shuai Peng, Di Fu, Liangcai Gao, Xiuqin Zhong, Hongguang Fu, and Zhi Tang. 2024. Multimath: Bridging visual and mathematical reasoning for large language models. *arXiv preprint arXiv:2409.00147*.

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*.

Annabelle Rabillas, Osias Kit Kilag, Neil Cañete, Maria Trazona, Mery Lou Calope, and Jacqueline Kilag. 2023. Elementary math learning through piaget's cognitive development stages. *Excellencia: International Multi-disciplinary Journal of Education (2994-9521)*, 1(4):128–142.

AM Rahman, Junyi Ye, Wei Yao, Sierra S Liu, Jesse Yu, Jonathan Yu, Wenpeng Yin, and Guiling Wang. 2024. From blind solvers to logical thinkers: Benchmarking llms' logical integrity on faulty mathematical problems. *arXiv preprint arXiv:2410.18921*.

Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, and 1 others. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Junhong Shen, Liam Li, Lucio M Dery, Corey Staten, Mikhail Khodak, Graham Neubig, and Ameet Talwalkar. 2023. Cross-modal fine-tuning: Align then refine. In *International Conference on Machine Learning*, pages 31030–31056. PMLR.

Haizhou Shi, Zihao Xu, Hengyi Wang, Weiyi Qin, Wenyuan Wang, Yibin Wang, Zifeng Wang, Sayna Ebrahimi, and Hao Wang. 2024a. Continual learning of large language models: A comprehensive survey. *arXiv preprint arXiv:2404.16789*.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. 2023. Detecting pretraining data from large language models. *arXiv preprint arXiv:2310.16789*.

Wenhao Shi, Zhiqiang Hu, Yi Bin, Junhua Liu, Yang Yang, See-Kiong Ng, Lidong Bing, and Roy Ka-Wei Lee. 2024b. Math-llava: Bootstrapping mathematical reasoning for multimodal large language models. *arXiv preprint arXiv:2406.17294*.

Wenhao Shi, Zhiqiang Hu, Yi Bin, Junhua Liu, Yang Yang, See-Kiong Ng, Lidong Bing, and Roy Ka-Wei Lee. 2024c. Math-llava: Bootstrapping mathematical reasoning for multimodal large language models. *arXiv preprint arXiv:2406.17294*.

Mingyang Song, Zhaochen Su, Xiaoye Qu, Jiawei Zhou, and Yu Cheng. 2025. Prmbench: A fine-grained and challenging benchmark for process-level reward models. *arXiv preprint arXiv:2501.03124*.

Shezheng Song, Xiaopeng Li, and Shasha Li. 2023. How to bridge the gap between modalities: A comprehensive survey on multimodal large language model. *arXiv preprint arXiv:2311.07594*.

Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Heung-Yeung Shum, and Jian Guo. 2023. Think-on-graph: Deep and responsible reasoning of large language model with knowledge graph. *arXiv preprint arXiv:2307.07697*.

Liangtai Sun, Yang Han, Zihan Zhao, Da Ma, Zhennan Shen, Baocai Chen, Lu Chen, and Kai Yu. 2024. Scieval: A multi-level large language model evaluation benchmark for scientific research. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19053–19061.

Qwen Team. 2024. Qvq: To see the world with wisdom.

Kushal Tirumala, Aram Markosyan, Luke Zettlemoyer, and Armen Aghajanyan. 2022. Memorization without overfitting: Analyzing the training dynamics of large language models. *Advances in Neural Information Processing Systems*, 35:38274–38290.

Trieu H Trinh, Yuhuai Wu, Quoc V Le, He He, and Thang Luong. 2024. Solving olympiad geometry without human demonstrations. *Nature*, 625(7995):476–482.

George Tsoukalas, Jasper Lee, John Jennings, Jimmy Xin, Michelle Ding, Michael Jennings, Amitayush Thakur, and Swarat Chaudhuri. 2024. Putnambench: Evaluating neural theorem-provers on the putnam mathematical competition. *arXiv preprint arXiv:2407.11214*.

Anna Vettoruzzo, Mohamed-Rafik Bouguelia, Joaquin Vanschoren, Thorsteinn Rognvaldsson, and KC Santosh. 2024. Advances and challenges in meta-learning: A technical review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Binghai Wang, Rui Zheng, Lu Chen, Yan Liu, Shihan Dou, Caishuang Huang, Wei Shen, Senjie Jin, Enyu Zhou, Chenyu Shi, and 1 others. 2024a. Secrets of rlhf in large language models part ii: Reward modeling. *arXiv preprint arXiv:2401.06080*.

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. *Advances in Neural Information Processing Systems*, 37:95095–95169.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Mingjie Zhan, and Hongsheng Li. 2024c. Measuring multimodal mathematical reasoning with math-vision dataset. *arXiv preprint arXiv:2402.14804*.

Peijie Wang, Zhong-Zhi Li, Fei Yin, Dekang Ran, and Cheng-Lin Liu. 2025. Mv-math: Evaluating multimodal math reasoning in multi-visual contexts. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19541–19551.

Shen Wang, Tianlong Xu, Hang Li, Chaoli Zhang, Joleen Liang, Jiliang Tang, Philip S Yu, and Qingsong Wen. 2024d. Large language models for education: A survey and outlook. *arXiv preprint arXiv:2403.18105*.

Weihan Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang, Lei Zhao, Xixuan Song, and 1 others. 2023. Cogvlm: Visual expert for pretrained language models. *arXiv preprint arXiv:2311.03079*.

Xiaoxuan Wang, Ziniu Hu, Pan Lu, Yanqiao Zhu, Jieyu Zhang, Satyen Subramaniam, Arjun R Loomba, Shichang Zhang, Yizhou Sun, and Wei Wang. 2024e. Scibench: Evaluating college-level scientific problem-solving abilities of large language models. *arXiv preprint arXiv:2307.10635*.

Felix A Wichmann and Robert Geirhos. 2023. Are deep neural networks adequate behavioral models of human visual perception? *Annual Review of Vision Science*, 9(1):501–524.

Hanguang Xiao, Feizhong Zhou, Xingyue Liu, Tianqi Liu, Zhipeng Li, Xin Liu, and Xiaoxuan Huang. 2024. A comprehensive survey of large language models and multimodal large language models in medicine. *arXiv preprint arXiv:2405.08603*.

Yi Xin, Junlong Du, Qiang Wang, Ke Yan, and Shouhong Ding. 2024. Mmap: Multi-modal alignment prompt for cross-domain multi-task learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 16076–16084.

Fengli Xu, Qianyue Hao, Zefang Zong, Jingwei Wang, Yunke Zhang, Jingyi Wang, Xiaochong Lan, Jiahui Gong, Tianjian Ouyang, Fanjin Meng, and 1 others. 2025. Towards large reasoning models: A survey of reinforced reasoning with large language models. *arXiv preprint arXiv:2501.09686*.

Liang Xu, Hang Xue, Lei Zhu, and Kangkang Zhao. 2024. Superclue-math6: Graded multi-step math reasoning benchmark for llms in chinese. *arXiv preprint arXiv:2401.11819*.

Runxin Xu, Fuli Luo, Zhiyuan Zhang, Chuanqi Tan, Baobao Chang, Songfang Huang, and Fei Huang. 2021. Raise a child in large language model: Towards effective and generalizable fine-tuning. *arXiv preprint arXiv:2109.05687*.

Yibo Yan and Joey Lee. 2024. Georeasoner: Reasoning on geospatially grounded context for natural language understanding. *arXiv preprint arXiv:2408.11366*.

Yibo Yan, Jiamin Su, Jianxiang He, Fangteng Fu, Xu Zheng, Yuanhuiyi Lyu, Kun Wang, Shen Wang, Qingsong Wen, and Xuming Hu. 2024a. A survey of mathematical reasoning in the era of multimodal large language model: Benchmark, method & challenges. *arXiv preprint arXiv:2412.11936*.

Yibo Yan, Shen Wang, Jiahao Huo, Jingheng Ye, Zhendong Chu, Xuming Hu, Philip S Yu, Carla Gomes, Bart Selman, and Qingsong Wen. 2025a. Position: Multimodal large language models can significantly advance scientific reasoning. *arXiv preprint arXiv:2502.02871*.

Yibo Yan, Shen Wang, Jiahao Huo, Philip S Yu, Xuming Hu, and Qingsong Wen. 2025b. Mathagent: Leveraging a mixture-of-math-agent framework for real-world multimodal mathematical error detection. *arXiv preprint arXiv:2503.18132*.

Yibo Yan, Haomin Wen, Siru Zhong, Wei Chen, Haodong Chen, Qingsong Wen, Roger Zimmermann, and Yuxuan Liang. 2024b. Urbanclip: Learning text-enhanced urban region profiling with contrastive language-image pretraining from the web. In *Proceedings of the ACM on Web Conference 2024*, pages 4006–4017.

Xu Yang, Yongliang Wu, Mingzhuo Yang, Haokun Chen, and Xin Geng. 2024a. Exploring diverse in-context configurations for image captioning. *Advances in Neural Information Processing Systems*, 36.

Zhen Yang, Jinhao Chen, Zhengxiao Du, Wenmeng Yu, Weihan Wang, Wenyi Hong, Zhihuan Jiang, Bin Xu, and Jie Tang. 2024b. Mathglm-vision: Solving mathematical problems with multi-modal large language model. *arXiv preprint arXiv:2409.13729*.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, and 1 others. 2024. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint arXiv:2408.01800*.

Shuo Yin, Weihao You, Zhilong Ji, Guoqiang Zhong, and Jinfeng Bai. 2024. Mumath-code: Combining tool-use large language models with multi-perspective data augmentation for mathematical reasoning. *arXiv preprint arXiv:2405.07551*.

Weihao You, Shuo Yin, Xudong Zhao, Zhilong Ji, Guoqiang Zhong, and Jinfeng Bai. 2024. Mumath: Multi-perspective data augmentation for mathematical reasoning in large language models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2932–2958.

Alex Young, Bei Chen, Chao Li, Chengen Huang, Ge Zhang, Guanwei Zhang, Heng Li, Jiangcheng Zhu, Jianqun Chen, Jing Chang, and 1 others. 2024. Yi: Open foundation models by 01. ai. *arXiv preprint arXiv:2403.04652*.

Yue Yu, Yuchen Zhuang, Jieyu Zhang, Yu Meng, Alexander J Ratner, Ranjay Krishna, Jiaming Shen, and Chao Zhang. 2024. Large language model as attributed training data generator: A tale of diversity and bias. *Advances in Neural Information Processing Systems*, 36.

Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, and 1 others. 2024. Mmmu-pro: A more robust multi-discipline multimodal understanding benchmark. *arXiv preprint arXiv:2409.02813*.

Liang Zeng, Liangjun Zhong, Liang Zhao, Tianwen Wei, Liu Yang, Jujie He, Cheng Cheng, Rui Hu, Yang Liu, Shuicheng Yan, and 1 others. 2024a. Skywork-math: Data scaling laws for mathematical reasoning in large language models–the story goes on. *arXiv preprint arXiv:2407.08348*.

Zhongshen Zeng, Yinhong Liu, Yingjia Wan, Jingyao Li, Pengguang Chen, Jianbo Dai, Yuxuan Yao, Rongwu Xu, Zehan Qi, Wanru Zhao, and 1 others. 2024b. Mr-ben: A comprehensive meta-reasoning benchmark for large language models. *arXiv preprint arXiv:2406.13975*.

Jiaxin Zhang, Zhongzhi Li, Mingliang Zhang, Fei Yin, Chenglin Liu, and Yashar Moshfeghi. 2024a. Geoeval: benchmark for evaluating llms and multi-modal models on geometry problem-solving. *arXiv preprint arXiv:2402.10104*.

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

Renrui Zhang, Xinyu Wei, Dongzhi Jiang, Yichi Zhang, Ziyu Guo, Chengzhuo Tong, Jiaming Liu, Aojun Zhou, Bin Wei, Shanghang Zhang, and 1 others. 2024c. Mavis: Mathematical visual instruction tuning. *arXiv e-prints*, pages arXiv–2407.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, and 1 others. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Kening Zheng, Junkai Chen, Yibo Yan, Xin Zou, and Xuming Hu. 2024. Reefknot: A comprehensive benchmark for relation hallucination evaluation, analysis and mitigation in multimodal large language models. *arXiv preprint arXiv:2408.09429*.

Siru Zhong, Xixuan Hao, Yibo Yan, Ying Zhang, Yangqiu Song, and Yuxuan Liang. 2024. Urbancross: Enhancing satellite image-text retrieval with cross-domain adaptation. *arXiv preprint arXiv:2404.14241*.

Minxuan Zhou, Hao Liang, Tianpeng Li, Zhiyu Wu, Mingan Lin, Linzhuang Sun, Yaqi Zhou, Yan Zhang, Xiaoqin Huang, Yicong Chen, and 1 others. 2024a. Mathscape: Evaluating mllms in multimodal math scenarios through a hierarchical benchmark. *arXiv preprint arXiv:2408.07543*.

Zihao Zhou, Shudong Liu, Maizhen Ning, Wei Liu, Jindong Wang, Derek F. Wong, Xiaowei Huang, Qiufeng Wang, and Kaizhu Huang. 2024b. Is your model really a good math reasoner? evaluating mathematical reasoning with checklist.

Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Haonan Chen, Zhicheng Dou, and Ji-Rong Wen. 2023. Large language models for information retrieval: A survey. *arXiv preprint arXiv:2308.07107*.

Wenwen Zhuang, Xin Huang, Xiantao Zhang, and Jin Zeng. 2024. Math-puma: Progressive upward multimodal alignment to enhance mathematical reasoning. *arXiv preprint arXiv:2408.08640*.

Wenwen Zhuang, Xin Huang, Xiantao Zhang, and Jin Zeng. 2025. Math-puma: Progressive upward multimodal alignment to enhance mathematical reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 26183–26191.

Chengke Zou, Xingang Guo, Rui Yang, Junyu Zhang, Bin Hu, and Huan Zhang. 2024. Dynamath: A dynamic visual benchmark for evaluating mathematical reasoning robustness of vision language models. *arXiv preprint arXiv:2411.00836*.

Xingchen Zou, Yibo Yan, Xixuan Hao, Yuehong Hu, Haomin Wen, Erdong Liu, Junbo Zhang, Yong Li, Tianrui Li, Yu Zheng, and 1 others. 2025. Deep learning for cross-domain data fusion in urban computing: Taxonomy, advances, and outlook. *Information Fusion*, 113:102606.

Zhengxia Zou, Keyan Chen, Zhenwei Shi, Yuhong Guo, and Jieping Ye. 2023. Object detection in 20 years: A survey. *Proceedings of the IEEE*, 111(3):257–276.

Contents of Technical Appendices

Technical Appendices and Supplements

A Comparison with Relevant Benchmarks

The field of mathematical reasoning evaluation has seen a significant proliferation of benchmarks, each designed to probe the capabilities of LLMs and MLLMs. As illustrated in Table 1, these benchmarks can be broadly categorized.

A substantial portion of existing work has concentrated on text-based mathematical problems. Benchmarks such as TheoremQA (Chen et al., 2023a), MathBench (Liu et al., 2024b), MR-GSM8K (Zeng et al., 2024b), and SciEval (Sun et al., 2024) provide valuable resources for assessing the pure linguistic and logical reasoning abilities of LLMs on mathematical tasks. Recognizing the inherently multimodal nature of many real-world math problems, the community has also developed numerous benchmarks that integrate visual information. Datasets like CMMaTH (Li et al., 2024d), MathScape (Zhou et al., 2024a), MATH-V (Wang et al., 2024c), MathVista (Lu et al., 2024b), and MathVerse (Zhang et al., 2024b) challenge MLLMs to solve problems by jointly interpreting textual and visual inputs (See more discussion on these benchmarks in Appendix D.1). However, a common thread among these benchmarks is their primary focus on evaluating a model's **problem-solving ability**—that is, their capacity to generate a correct final answer. More recently, the task of error analysis has begun to emerge. For instance, the EIC benchmark (Li et al., 2024c) made a notable contribution by introducing tasks for error identification and correction. While this represents a critical step toward evaluating deeper reasoning, EIC is limited to a *text-only* modality.

In this context, ErrorRadar is uniquely positioned to fill two critical gaps in the current evaluation landscape. First, it is the **first benchmark designed to specifically evaluate multimodal error detection**. By requiring models to analyze reasoning steps in a context that includes both text and images, it presents a more complex and realistic challenge that moves beyond simple problem-solving. Second, and most distinctively, ErrorRadar is built upon **real student answers**. Unlike benchmarks that rely on synthetically generated data or correct solutions, our dataset captures the authentic, often nuanced errors that human learners make. This provides a more robust and educationally relevant setting to test the fine-grained diagnostic capabilities of MLLMs.

Benchmarks	Venue	Modality	Student Ans.	Error Det.
TheoremQA (Chen et al., 2023a)	EMNLP	T	-	-
MathBench (Liu et al., 2024b)	ACL	T	-	-
MR-GSM8K (Zeng et al., 2024b)	ICLR	T	-	-
SciEval (Sun et al., 2024)	AAAI	T	-	-
EIC (Li et al., 2024c)	ACL Finding	T	-	✓
CMMaTH (Li et al., 2024d)	COLING	T, I	-	-
MathScape (Zhou et al., 2024a)	arXiv	T, I	-	-
MATH-V (Wang et al., 2024c)	NeurIPS	T, I	-	-
Olympiadbench (He et al., 2024a)	ACL	T, I	-	-
QRData (Liu et al., 2024c)	ACL	T, I	-	-
IsoBench (Fu et al., 2024)	COLM	T, I	-	-
SciBench (Wang et al., 2024e)	ICML	T, I	-	-
EMMA (Hao et al., 2025)	ICML	T, I	-	-
MathCheck (Zhou et al., 2024b)	ICLR	T, I	-	-
DynaMath (Zou et al., 2024)	ICLR	T, I	-	-
MathVista (Lu et al., 2024b)	ICLR	T, I	-	-
Math-Vision (Wang et al., 2024b)	NeurIPS	T, I	-	-
MV-MATH (Wang et al., 2025)	CVPR	T, I	-	-
CMM-MATH (Liu et al., 2025)	ACM MM	T, I	-	-
MathVerse (Zhang et al., 2024b)	ECCV	T, I	-	-
ErrorRadar (Ours)	-	T, I	✓	✓

Table 2: Comparison between our proposed ErrorRadar benchmark vs. its relevant LLM-based mathematical reasoning benchmarks or datasets. Under the column of ***Modality***, the letters T and I represent text and image, respectively. The column labeled as ***Student Ans.*** indicates whether the dataset contains real student data (*i.e.*, students' incorrect answers); the column labeled as ***Error Det.*** represents whether error detection task is included.

In summary, while previous benchmarks have laid essential groundwork for evaluating mathematical problem-solving, ErrorRadar introduces a novel, more demanding paradigm focused on multimodal error diagnostics with real-world data, thereby offering a more comprehensive assessment of complex mathematical reasoning.

B Impact Statement

The introduction of the ErrorRadar benchmark and the formalization of the multimodal error detection task represent significant strides in enhancing the complex reasoning abilities of MLLMs. We will release the whole dataset to the community upon acceptance, and we will be committed to refining and scaling up the dateset for further research. Its broader impact can be seen in several key areas:

First, the ability to detect and categorize errors in mathematical reasoning is critical for improving the efficacy of AI in educational settings. By using real-world data sourced from student interactions, ErrorRadar provides invaluable insights into the cognitive and error patterns of learners. These insights can inform personalized learning interventions, helping to identify not only where students struggle but also why they do so.

Second, while the use of AI in educational settings presents substantial opportunities, it also necessitates a careful consideration of fairness, bias, and transparency. By incorporating human expert

evaluation and real-world student data, ERRORRADAR helps to mitigate risks of biased AI assessments, ensuring that MLLMs are held to high standards of accuracy across diverse error categories. However, as these models continue to evolve, it is crucial that they remain accessible and equitable across different educational contexts.

Last, the success of multimodal AI in addressing complex mathematical reasoning tasks could have a transformative effect on education at large, extending beyond traditional K-12 settings into higher education and lifelong learning. The ERRORRADAR benchmark sets the stage for further research into error detection across diverse disciplines, such as physics, economics, and even coding, where MLLMs could be used to enhance learning outcomes through improved diagnostic and instructional capabilities.

C Task Clarification

C.1 How ERRORRADAR Contribute to Complex Multimodal Math Reasoning

While our primary focus in this paper was to formally introduce the task, construct the benchmark, and establish baseline performances (as is typical for a benchmark paper), we argue that evaluating and understanding error detection directly contributes to the goal of enhancing complex mathematical reasoning in several crucial ways:

- **Error Detection as a Diagnostic Tool for Deeper Reasoning Failures:** Standard problem-solving benchmarks typically evaluate only final answer accuracy, with limited insight into why a model fails. ERRORRADAR offers a more granular diagnostic capability, via identifying where the reasoning breaks down and what kind of mistake was made.

- **Highlighting Specific Weaknesses for Targeted Improvement:** Our extensive experiments (See Section 4) reveal distinct error patterns across MLLMs:
  - Weaker open-source models often exhibit a bias towards predicting CAL, suggesting they oversimplify complex issues.
  - KNOW is challenging for most models.
  - Top models like GPT-4o struggle with visual perception compared to humans. These findings pinpoint specific areas (*e.g.,* visual grounding, knowledge integration) where models need improvement.
  - We explicitly provide **Actionable Suggestions** in Appendix G.5, directly linking the benchmark's insights to pathways for enhancing MLLM capabilities.

- **Error Detection as a Proxy for Robust Reasoning:** The ability to correctly identify an error in a given reasoning chain requires a deep understanding of the correct reasoning process itself. A model that can reliably perform error detection inherently possesses a more sophisticated grasp of mathematical logic, visual interpretation, and knowledge application. Therefore, success in error categorization signals a higher level of metacognitive-like reasoning.

C.2 Comparison between ERRORRADAR Setting and General LLM-as-Judge

To clearly illustrate the distinctions, we present the following comparison:

Therefore, while identifying wrong steps is a facet of some LLM-as-Judge applications, ErrorRadar defines a significantly **more specific**, **complex**, and **context-grounded** set of tasks.

C.3 Relationship between ERRORRADAR Setting and Multimodal Math Problem-Solving

As shown in Section 4.3.1 Finding #6, results offer compelling evidence regarding this relationship:

Evidence of Relationship: The fact that these math-specialized models can perform the ERRORRADAR tasks demonstrates a fundamental relationship. Both problem-solving and error detection operate within the multimodal mathematics domain. They require core capabilities like understanding mathematical notation and terminology presented textually.

Evidence of Distinction: The performance of these specialized models on ERRORRADAR was only moderate. This performance gap provides evidence that the capabilities required by ERRORRADAR are distinct from, and not merely a subset of, problem-solving ability. The math-specialized models' training optimizes for generating correct solution paths but may not sufficiently equip them to diagnose errors in incorrect paths.

Table 3: Comparison between ERRORRADAR and general LLM-as-Judge paradigm.

Dimension	ERRORRADAR	General LLM-as-Judge
Input Data	Real student incorrect step-by-step solutions & problem context	Often model-generated outputs
Core Task	Specific & Fine-grained	General & Variable
Output Granularity	Precise & Structured	Score, ranking, etc.
Evaluation Focus	Diagnostic Accuracy	Overall correctness, coherence, etc.
Context/Purpose	Educational Diagnostics	Model evaluation
Error Taxonomy	Predefined & Educationally Relevant	Usually no fixed, fine-grained error taxonomy

While our benchmark primarily focuses on evaluating MLLMs' ability to detect errors in student-provided reasoning chains (a critical task for educational applications), we acknowledge the importance of assessing *whether MLLMs can independently solve the same problems correctly*. To address this, we conducted additional experiments: GPT-4o achieved 82% accuracy in solving problems independently, significantly higher than its error detection performance. This aligns with prior work (*e.g.,* MathVista (Lu et al., 2024b)) showing that MLLMs excel in direct problem-solving but struggle with error analysis.

D More Related Work

D.1 Mathematical Reasoning Benchmarks

The landscape of mathematical reasoning benchmarks has evolved significantly, **initially focusing on textual problem-solving and gradually incorporating multimodal inputs and more complex reasoning tasks.** Early efforts in benchmarking, such as MathQA (Amini et al., 2019) and GSM8K (Cobbe et al., 2021), along with the widely used MATH dataset (Hendrycks et al., 2021), primarily assessed models on arithmetic and word problems presented in a text-only format. These were instrumental in the pre-LLM and early LLM era. As models grew more sophisticated, benchmarks like LILA (Mishra et al., 2022) continued to expand the scope of text-based mathematical reasoning. However, recognizing that many real-world mathematical problems inherently involve visual components, the research community has increasingly developed multimodal benchmarks. Prominent examples include MathVista (Lu et al., 2024b), which evaluates reasoning in visual contexts, MathVerse (Zhang et al., 2024b) focusing on whether MLLMs truly understand diagrams, and MMMU (Yue et al., 2024) for massive multi-discipline multimodal understanding. Specific multimodal domains like geometry have also seen dedicated benchmarks such as GeoQA (Chen et al., 2021), GeomVerse (Kazemi et al., 2023), GeoEval (Zhang et al., 2024a), MATH-Vision (Wang et al., 2024c), and DynaMath (Zou et al., 2024) which introduces dynamic visual elements. Efforts to diversify language coverage in multimodal settings are also emerging, with datasets like CMM-Math (Li et al., 2024d) for Chinese.

Beyond the transition to multimodality, benchmark tasks have expanded **from straightforward problem-solving to encompass more intricate aspects of mathematical reasoning**. While problem-solving remains a dominant task, as seen in competition-level benchmarks like Omni-MATH (Gao et al., 2024), CHAMP (Mao et al., 2024), and PutnamBench (Tsoukalas et al., 2024), there's a growing focus on higher-order thinking. This includes mathematical proving, evaluated by datasets such as TheoremQA (Chen et al., 2023a) and the IMO-AG-30 subset used with AlphaGeometry (Trinh et al., 2024). A particularly significant development is the increased attention to error analysis. Benchmarks like EIC-Math (Li et al., 2024c) and Mathador-LM (Kurtic et al., 2024) focus on error identification and, in some cases, correction within textual solutions. FaultyMath (Rahman et al., 2024) evaluates logical integrity on faulty problems. The complexity of evaluation is further pushed by benchmarks addressing multi-turn interactions like MathChat (Liang et al., 2024b), model robustness through adversarial examples

(GSM-PLUS by (Li et al., 2024b)), and specialized quantitative reasoning with data (QRData by (Liu et al., 2024c)) or mathematical modeling (Mamo by (Huang et al., 2024)). These advancements reflect a drive towards more comprehensive and challenging evaluations of (M)LLMs' mathematical reasoning capabilities (Yan et al., 2024a, 2025b).

D.2 Math-Specific MLLMs

The evolution of AI in mathematical reasoning has significantly advanced with the advent of Math-Specific MLLMs (Math-MLLMs), which are engineered to interpret and resolve mathematical problems incorporating both textual and crucial visual elements like diagrams, graphs, and geometric figures (Yan et al., 2024a). Among the notable developments in this domain is MathGLM-Vision (Yang et al., 2024b), a model explicitly designed to integrate visual information for solving mathematical problems. Similarly, Math-LLaVA (Shi et al., 2024c) leverages fine-tuning on an extensive dataset of 360K high-quality multimodal math question-answer pairs (MathV360K) to directly enhance its multimodal mathematical reasoning capabilities. Addressing the challenge of data scarcity, MAVIS (Zhang et al., 2024c) features an automatic data generation engine and employs instruction fine-tuning to teach models problem decomposition. While primarily text-based, models such as Skywork-Math (Zeng et al., 2024a) and Qwen2.5-Math (Bai et al., 2025) are also adapting to support multimodal mathematical settings. Further contributions include Math-PUMA (Zhuang et al., 2025) with its progressive upward multimodal alignment strategy, and InfiMM-Math (Han et al., 2024), which achieves strong performance through training on a large-scale, LLM-validated multimodal dataset. Methodological innovations are also prominent, such as Visual Sketchpad (Hu et al., 2024b) enabling MLLMs to generate intermediate sketches, G-LLaVA (Gao et al., 2023) focusing on geometry, STIC (Deng et al., 2024) employing self-training for visual comprehension, and VCAR (Jia et al., 2024b) emphasizing visual-centric supervision. These collective efforts highlight a strong push towards integrating visual perception with textual understanding, though challenges in advanced visual reasoning and the need for diverse, large-scale datasets remain critical for future progress (Yan et al., 2024a, 2025a).

E More Multimodal Question Examples

In this section, you can refer to Figures 12, 13, 14, 15, and 16 for more concrete examples from our ErrorRadar dataset.

F Additional Dataset Details

F.1 Annotation Details

To ensure the quality and relevance of the ErrorRadar dataset for error detection tasks, we employed a rigorous manual annotation process, involving professional educational experts as annotators. This section outlines the details of the annotation procedure, focusing on how the data was enriched with step-by-step reasoning processes, identification of erroneous steps, and error categorization.

Annotator Selection and Training. Given the complexity of the task, we recruited a group of ten annotators with specialized knowledge in educational theory and mathematics, particularly in K-12 pedagogy. These annotators were trained extensively to familiarize themselves with the structure and expectations of the task. The training covered the specifics of multimodal problem-solving in mathematics, typical student error patterns, and the need for precise identification of reasoning steps that led to incorrect answers. The annotators were also briefed on using the provided tools and the quality assurance process.

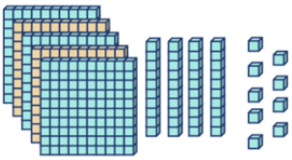
Annotation Process. Each mathematical problem in the dataset was annotated with a step-by-step reasoning process, capturing both correct and incorrect approaches to problem-solving. Annotators were provided with:

1. The original question stem (comprising both text and image components).
2. The student's most frequent incorrect answer.
3. The correct answer to the question.
4. The pedagogical analysis of the correct reasoning process, prepared by educational experts.

Based on these inputs, annotators were tasked with:

1. **Step-by-Step Reasoning Annotation**: For each problem, annotators mapped out the logical steps that students should ideally follow to arrive at the correct answer. This involved identifying key stages in the problem-solving

Question (Visual Part):



Question (Textual Part):

The diagram has _____ small squares.

Student's Step-by-Step Solution (Reasoning Process):

- Step 1: Read the question and the text in the diagram "The diagram below has _____ small blocks."
- Step 2: Identify that there are 5 groups of small blocks in the diagram, each containing 100, representing 5 hundreds.
- **Step 3: Identify that there are 2 strips of small blocks in the diagram, each containing 10, representing 2 tens.**
- Step 4: Identify that there are 9 individual small blocks in the diagram, representing 9 ones.
- Step 5: Combine the 5 hundreds, 2 tens, and 9 ones to write: 529.
- Step 6: Output the result as 529.

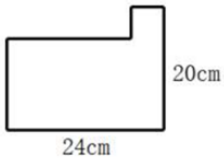
True Answer: 549
Student's Answer: 529

Problem Type: Counting
Error Category: Visual Perception Error
Error Step: Step 3

ErrorRadar	Error Category	Error Step
GPT-4o	Visual Perception Error	Step 3
GPT-4o-mini	Knowledge Error	Step 2
Gemini-Pro-1.5	Visual Perception Error	Step 2
LLaVA-v1.6	Reasoning Error	Step 3

Figure 12: Multimodal mathematical example one (type: counting) from ERRORRADAR dataset.

Question (Visual Part):



Question (Textual Part):

The perimeter of the figure below is _____ cm.

Student's Step-by-Step Solution (Reasoning Process):

- Step 1: Read the text in the question and diagram "The perimeter of the figure below is _____cm."
- Step 2: Determine the width of the figure as 20 cm.
- Step 3: Determine the length of the figure as 24 cm.
- **Step 4: Convert the perimeter of the figure into the perimeter of a rectangle with a length of 24 cm and a width of 10 cm.**
- Step 5: Calculate the perimeter of the rectangle as $2 \times (24 + 10)$.
- Step 6: Calculate the result as $2 \times 34 = 68$ (cm).
- Step 7: Output the result as 68.

True Answer: 88
Student's Answer: 68

Problem Type: Plane Geometry
Error Category: Reasoning Error
Error Step: Step 4

ErrorRadar	Error Category	Error Step
GPT-4o	Reasoning Error	Step 4
GPT-4o-mini	Visual Perception Error	Step 4
Gemini-Pro-1.5	Knowledge Error	Step 4
LLaVA-v1.6	Calculation Error	Step 5

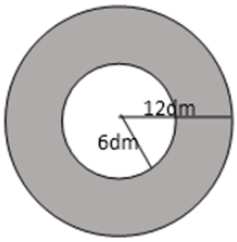
Figure 13: Multimodal mathematical example two (type: plane geometry) from ERRORRADAR dataset.

process, such as formula application, arithmetic operations, or logical deductions.

2. **Error Step Identification**: For problems where students provided incorrect answers, annotators identified the exact steps where the reasoning went wrong. These error steps were explicitly marked and linked to the incorrect responses, ensuring that they could be traced back to specific problem-solving mistakes.

3. **Error Categorization**: Once the erroneous step was identified, annotators assigned an appropriate error category based on a predefined

Question (Visual Part):



Question (Textual Part):

The area of the shaded region in the figure is _____ square decimeters.

Student's Step-by-Step Solution (Reasoning Process):

- Step 1: Read the problem description and the text in the figure: "The area of the shaded part is _____ square decimeters."
- Step 2: Determine that the outer diameter of the ring in the figure is 24 decimeters (twice 12 decimeters).
- Step 3: Calculate the area of the outer circle: 3.14 × (12)^2 = 3.14 × 144 = 452.16 square decimeters.
- Step 4: Determine that the inner diameter of the ring in the figure is 24 decimeters.
- Step 5: Calculate the area of the inner circle: 3.14 × (12)^2 = 3.14 × 144 = 452.16 square decimeters.
- Step 6: Calculate the area of the shaded part as the product of the inner circle area and the outer circle area, i.e., 452.16 × 452.16 = 204276.1856 square decimeters.
- Step 7: Output the result as 2599.92.

True Answer: 339.12

Student's Answer: 2599.92

Problem Type: Plane Geometry

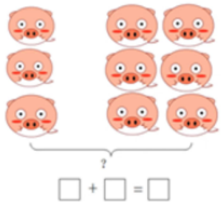
Error Category: Visual Perception Error

Error Step: Step 4

ErrorRadar	Error Category	Error Step
GPT-4o	Reasoning Error	Step 6
Qwen-VL	Reasoning Error	Step4
MiniCPM-V2.5	Calculation Error	Step 6
Human	Visual Perception Error	Step 6

Figure 14: Multimodal mathematical example three (type: plane geometry) from ERRORRADAR dataset.

Question (Visual Part):



Question (Textual Part):

Look at the picture and write the equation. The complete equation is _____ .

Student's Step-by-Step Solution (Reasoning Process):

- Step 1: Read the problem and the text in the image "Look at the picture and write the equation."
- Step 2: Identify there are 9 piglets on the left.
- Step 3: Identify there are 3 piglets on the right.
- Step 4: Subtract the 3 piglets on the right from the 9 piglets on the left, i.e., 9 - 3.
- Step 5: Calculate the result as 6.
- Step 6: Output the result as 9 - 3 = 6.

True Answer: 3+6=9

Student's Answer: 9-3=6

Problem Type: Counting

Error Category: Visual Perception Error

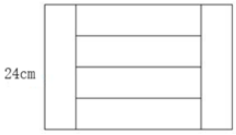
Error Step: Step 2

ErrorRadar	Error Category	Error Step
GPT-4o	Reasoning Error	Step 4
GPT-4o-mini	Misinterpretation of the Question	Step 3
LLaVA-v1.6	Calculation Error	Step 4
Human	Visual Perception Error	Step 2

Figure 15: Multimodal mathematical example four (type: counting) from ERRORRADAR dataset.

schema. These categories included common types of errors such as misinterpretation of the question (More details can be seen in Section 3.1). The categorization was designed

Question (Visual Part):



Question (Textual Part):

As shown in the figure, a large rectangle is divided into 6 identical smaller rectangles. If the length of a smaller rectangle is 24 centimeters, then the area of the large rectangle is _____ square centimeters.

Student's Step-by-Step Solution (Reasoning Process):

- Step 1: Read the problem statement and the text in the diagram, "Divide a large rectangle into 6 identical smaller rectangles."
- Step 2: Determine that the length of each smaller rectangle is 24 centimeters.
- Step 3: From the diagram, it is known that the length of the smaller rectangle is 4 times its width.
- Step 4: Calculate the width of the smaller rectangle, 24 ÷ 4 = 6 (centimeters).
- Step 5: Calculate the area of one smaller rectangle, 24 × 6 = 144 (square centimeters).
- **Step 6: Calculate the area of the large rectangle, which is the sum of the areas of 6 smaller rectangles, 144 × 6 = 764 (square centimeters).**
- Step 7: The output result is 764.

True Answer: 864

Student's Answer: 764

Problem Type: Plane Geometry

Error Category: Calculation Error

Error Step: Step 6

ErrorRadar	Error Category	Error Step
GPT-4o	Calculation Error	Step 6
GPT-4o-mini	Reasoning Error	Step 6
Gemini-Pro-1.5	Calculation Error	Step 3
LLaVA-v1.6	Reasoning Error	Step 3

Figure 16: Multimodal mathematical example five (type: plane geometry) from ERRORRADAR dataset.

to align with known student error patterns in mathematical learning.

Quality Control and Cross-Validation. To ensure annotation accuracy and consistency, each problem underwent two rounds of cross-checking:

1. **First Round of Cross-Validation**: After the initial annotation, another annotator independently reviewed the annotations. Any discrepancies between the first and second annotators were flagged for further analysis.

2. **Second Round of Cross-Validation**: In the second round, if the errors or discrepancies persisted, the problem was escalated to a senior educational expert who acted as the annotation lead. The annotation lead adjudicated these contentious cases, ensuring that the final decision was both pedagogically sound and aligned with the dataset's goals.

Dataset Refinement. Throughout the annotation process, we worked closely with the educational organization from which the dataset originated. This collaboration ensured that the annotations were not only reliable but also adhered to the standards of the organization's question bank. Additionally, ongoing feedback and updates from the organization helped refine the dataset, making it more accurate and relevant for multimodal error detection tasks.

Annotation Duration and Effort. The annotation process for the ERRORRADAR dataset spanned over a period of at least two months. During this time, the annotators, comprised of both professional educational experts and domain specialists, worked meticulously through several stages of preparation, annotation, and validation. Each annotator dedicated significant time to understanding the dataset, reviewing the provided pedagogical analyses, and applying their domain knowledge to identify and categorize errors. The first phase, involving step-by-step reasoning annotation, took approximately six weeks, while the subsequent cross-validation and quality control efforts accounted for the remaining two weeks. Given the complexity of the tasks and the necessity for high precision, the team's sustained efforts ensured that the final dataset was of the highest quality.

By incorporating these annotations, ERRORRADAR provides a robust foundation for studying student errors in mathematical reasoning and enables the development of advanced models for error detection and correction.

F.2 Details of Handling Inconsistent Annotations

To ensure the quality and reliability of our dataset for the multimodal mathematical error detection task, we established a systematic approach to resolve annotation inconsistencies. This process balances annotator independence with rigorous quality control, ensuring that the dataset is both accurate and representative.

F.2.1 Annotation Agreement Principles

1. **Guided Consensus**: Annotations must align with clear, predefined guidelines covering the five error categories. Annotators are trained extensively to reduce subjective biases.

2. **Cross-Checking and Agreement Threshold**: Each instance is annotated by at least three annotators. Disagreements are flagged for further review.

3. **Systematic Review Process**: For inconsistent cases, a multi-step resolution process is applied:

   (a) **Initial Review**: Annotators discuss disagreements, referencing annotation guidelines and the specific problem context.

   (b) **Expert Arbitration**: For unresolved cases, a domain expert (e.g., an educational professional) reviews and finalizes the annotation.

   (c) **Consensus-Driven Decisions**: When possible, annotations are harmonized based on majority opinion or shared agreement after discussions.

F.2.2 Case Resolution Framework

Case Example 1: Visual Perception vs. Reasoning Error

- **Example**: A problem presents a bar chart requiring students to determine the highest value. A student misidentifies the tallest bar and selects the wrong answer.
  - Annotator A labels this as a Visual Perception Error, arguing the mistake stems from misreading the chart.
  - Annotator B classifies it as a Reasoning Error, interpreting the mistake as a failure to compare values logically.
- **Resolution: Annotators revisit the problem:**
  - If evidence shows the student misunderstood the chart format (e.g., interpreting height as quantity but misjudging due to poor visualization), it is classified as a Visual Perception Error.
  - If the student correctly interprets the chart but misapplies logical comparisons (e.g., failing to compare values explicitly), it is categorized as a Reasoning Error.

  For persistent disagreement, an expert examines the student's work, including any notes or intermediate steps, to determine the correct annotation.

Case Example 2: Knowledge vs. Misinterpretation of the Question

- **Example**: A problem asks for the perimeter of a rectangle, but the student calculates the area instead.
  - Annotator A identifies this as a Knowledge Error, attributing the mistake to a lack of understanding of perimeter concepts.
  - Annotator B labels it as a Misinterpretation of the Question, asserting that the student misunderstood what was being asked.
- **Resolution**:
  - Did the student's work demonstrate understanding of the concept but apply it incorrectly (Misinterpretation of the Question)?
  - Did the mistake reveal a fundamental gap in knowledge about perimeter (Knowledge Error)?

  If disagreement persists, the annotators consult the expert, who may analyze additional context (e.g., previous responses or annotations).

F.2.3 Handling Irreconcilable Disagreements

If discrepancies persist despite review and arbitration, the affected data points are excluded from the dataset. This strict policy prioritizes the overall quality and consistency of the dataset, ensuring that retained samples maintain high reliability.

F.2.4 Monitoring and Feedback

Periodic feedback sessions are conducted to recalibrate annotators and refine guidelines based on observed patterns of disagreement. This iterative approach minimizes future inconsistencies and enhances annotator alignment over time.

F.3 Definition of Problem Type Category

The ERRORRADAR dataset distinguishes five primary types of multimodal mathematical problems, each characterized by unique features:

- ✯ **Plane Geometry Problems**: These involve two-dimensional shapes and figures, requiring knowledge of properties such as angles, lines, and polygons. Solving these problems often depends on understanding basic geometric principles and theorems about plane figures.

- ✯ **Solid Geometry Problems**: In contrast to plane geometry, solid geometry involves three-dimensional objects, such as cubes, cylinders, and spheres. These problems require spatial visualization and understanding of volume, surface area, and the relationships between different three-dimensional shapes.

- ✯ **Diagram-Based Problems**: These require analysis of provided visual information, such as graphs, charts, or diagrams, to solve mathematical queries. Interpreting visual data correctly is crucial, as these problems test the ability to extract and analyze quantitative information from visual aids.

- ✯ **Algebra Problems**: Algebra problems focus on abstract symbols and variables to represent numbers and relationships. These include tasks like solving equations, manipulating algebraic expressions, and understanding functions. The problem-solving process typically involves logical reasoning and manipulation of mathematical symbols.

- ✯ **Math Commonsense Questions**: These encompass a variety of problem types, including time judgment, direction judgment, counting, and pattern recognition. Unlike the other categories, math commonsense challenges rely on everyday mathematical reasoning and problem-solving strategies that do not necessarily require formal mathematical knowledge, testing intuitive understanding rather than procedural skills.

These problem types highlight the ERRORRADAR dataset's diverse nature, with each category presenting distinct challenges and requiring specific reasoning abilities.

F.4 Development and Validation Process of Error Category

1. Cross-Team Collaboration to Align Task Needs

The process began with close collaboration between the research team and the education team to ensure that the error categories aligned with the unique requirements of the multimodal math error detection task. The research team provided insights into the task's technical objectives, focusing on precision and comprehensive error coverage. Simultaneously, the education team contributed their understanding of real-world educational scenarios, emphasizing the practical relevance and applicability of the error taxonomy to students' and teachers' needs.

Key Outcomes:

- Initial consensus that the categories must address both multimodal challenges and real-life classroom scenarios.

- Recognition of the need to balance academic rigor with user-friendly categorization.

2. Benchmark Survey and Focus Analysis

The research team conducted an extensive survey of representative benchmarks, focusing on error analysis frameworks in existing datasets. Examples included studies on problem-solving steps in educational AI and cognitive error modeling in multimodal tasks. The aim was to identify gaps in current frameworks and understand how existing taxonomies handle errors specific to visual, textual, and logical reasoning elements.

Key Outcomes:

- Identification of inadequacies in current benchmarks, particularly in addressing multimodal interactions like visual misinterpretations and reasoning errors tied to diagram-based tasks.

- Validation of the necessity for distinct categories to capture errors unique to multimodal math problems.

3. Collection of Feedback from Students and Teachers

The education team collected qualitative and quantitative feedback from students and teachers to ensure that the proposed error categories were grounded in real-world educational needs. Focus groups, surveys, and interviews were used to gather perspectives on common error patterns encountered during classroom activities and assessments.

Key Insights:

- Teachers highlighted frequent calculation errors (**CAL**) and reasoning errors (**REAS**) as significant roadblocks to effective problem-solving.
- Students often reported confusion stemming from visual misinterpretations (**VIS**) and misunderstanding the question intent (**MIS**).
- Feedback emphasized the importance of separating reasoning-based errors from knowledge-based errors (**KNOW**) for better diagnostic support.

4. Second Round of Discussion and Alignment

Following the feedback collection, the research and education teams reconvened to refine and align the error taxonomy. This phase involved iterative discussions to ensure that each category was distinct, comprehensive, and intuitive for annotators and end-users.

Adjustments Made:

- Clarified the scope of **Reasoning Errors (REAS)** to focus on improper logical application rather than factual knowledge gaps.
- Strengthened the definition of **Visual Perception Errors (VIS)** to address multimodal-specific challenges, such as interpreting diagrams or image-based data.
- Enhanced examples for each category to support annotation clarity.

5. Initial Finalization and Feedback from Educational Organization

The refined error categories were presented to a partner educational organization for feedback. This organization, which specializes in global education assessments, conducted an independent review and provided expert input.

Key Outcomes:

- Positive validation of the categories' relevance and comprehensiveness.
- Minor recommendations, such as specifying units and signs in the **Calculation Errors (CAL)** category, were integrated.

6. Final Validation and Alignment with Annotation Team

After incorporating feedback, the final set of error categories was finalized. The annotation team, comprising educational experts, received detailed guidelines and training to ensure consistent application of the taxonomy during the annotation process. Mock annotations were conducted to test the clarity and usability of the categories.

Final Adjustments:

- Annotators highlighted the need for clearer boundaries between **Reasoning Errors (REAS)** and **Knowledge Errors (KNOW)**, leading to additional examples and decision rules in the annotation guidelines.
- Alignment meetings ensured that all discrepancies and ambiguities were resolved before the dataset's official annotation began.

The aforementioned development process ensured that the five categories are comprehensive, robust, and applicable to both multimodal tasks and real-world educational scenarios.

G Additional Experimental Details

G.1 More Main Results

In this section, Figure 17 shows the **overall error category** performance of all models for F1, recall, and precision, respectively. Figure 18 shows the **overall error step** performance of all models. Tables 4, 5, and 6 show the **category-level error category** performance for recall, precision, and F1, respectively.

G.2 Human Performance Evaluation

In the Human Performance section, the evaluation involved three educational expert evaluators, each independently performing the two subtasks — error step identification and error categorization — on a set of multimodal math problems. To ensure the validity of their assessments, a rigorous cross-checking procedure was implemented. After the initial independent evaluations, the results from all

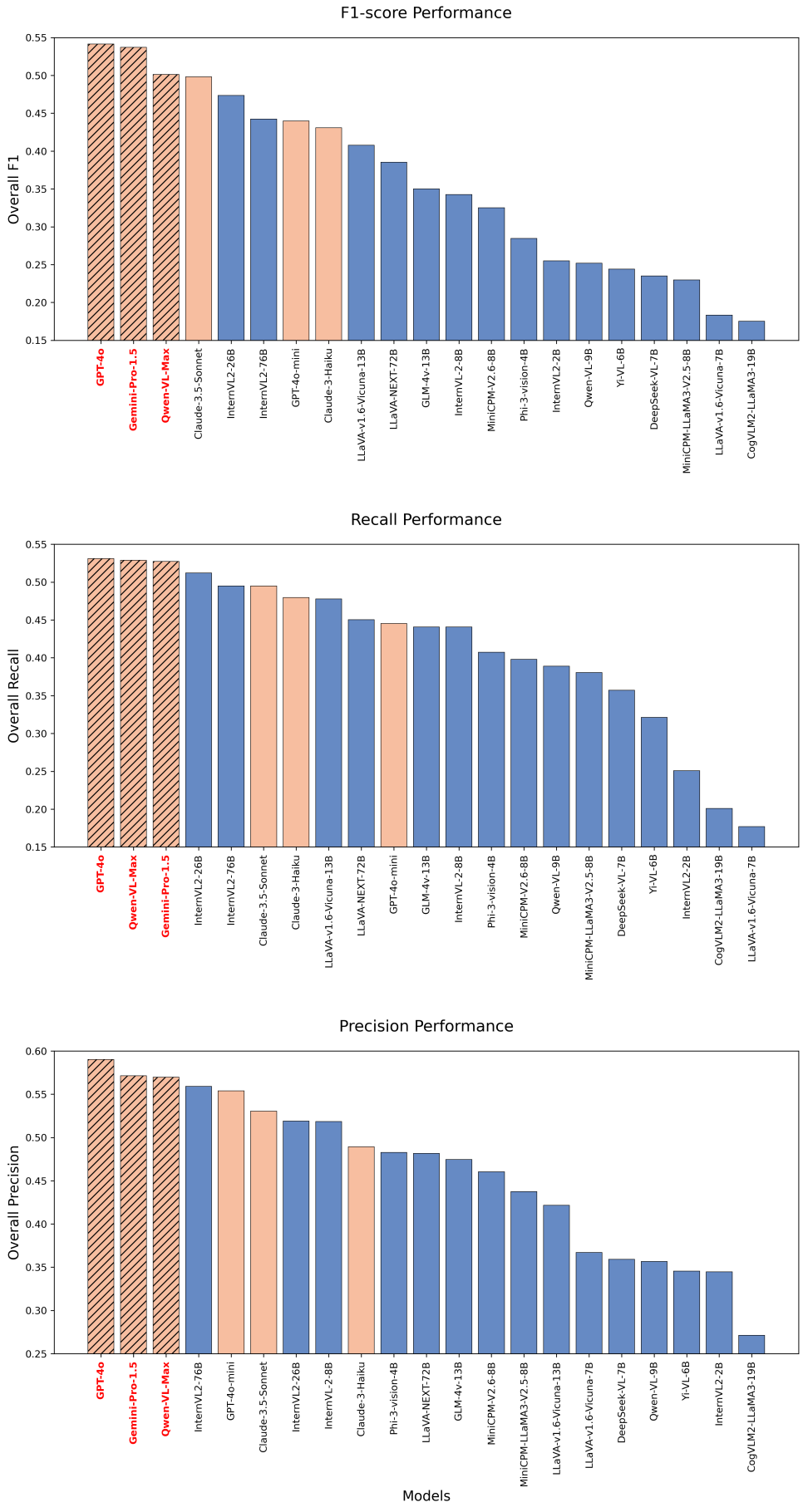


Figure 17: Error category performance of all models for F1, recall, and precision, respectively.

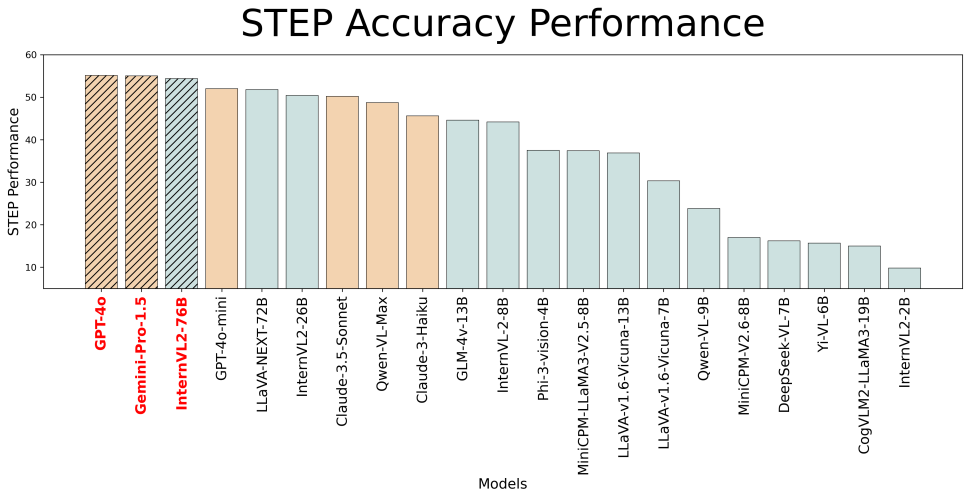


Figure 18: Error step performance of all models.

three experts were compared for both the identification of error steps and the categorization of those errors. When discrepancies arose, particularly in cases where the experts disagreed on which step contained an error or how an error should be classified, a structured conflict resolution process was followed.

The cross-check process began with identifying areas of disagreement between the evaluators. These conflicts were discussed in a series of consensus meetings, where the evaluators would review the conflicting steps or categorizations in detail. Each expert provided their rationale, referencing the mathematical principles involved as well as the multimodal representations of the problems. Through open dialogue, the evaluators aimed to reach a consensus on the correct interpretation of the error.

In cases where consensus could not be easily achieved, a majority-vote system was employed. However, for particularly complex or ambiguous cases, an additional adjudicator — who did not participate in the initial evaluations but had equivalent expertise — was consulted to provide a final judgment. This adjudicator reviewed the contentious cases along with the evaluators' justifications, ensuring an unbiased final decision. The outcome of this process was the creation of a refined ground truth dataset that balanced expert knowledge with the goal of consistent and reliable error identification and categorization.

G.3 Prompt for MLLM Evaluation

You can refer to Figures 19 and 20 for prompt details.

G.4 Model Sources

Table 7 details specific sources for the various MLLMs we evaluated. The hyperparameters for the experiments are set to their default values unless specified otherwise.

Multimodal Large Language Models	Parameters	VIS	CAL	REAS	KNOW	MIS
Open-Source MLLMs						
InternVL2 (Chen et al., 2023b)	2B	0.32	0.38	0.12	0.00	0.24
Phi-3-vision (Abdin et al., 2024)	4B	0.09	0.99	0.06	0.03	0.04
Yi-VL (Young et al., 2024)	6B	0.09	0.77	0.04	0.14	0.00
DeepSeek-VL (Lu et al., 2024a)	7B	0.04	0.90	0.00	0.28	0.06
LLaVA-v1.6-Vicuna (Liu et al., 2024a)	7B	0.40	0.14	0.08	0.00	0.55
InternVL-2 (Chen et al., 2023b)	8B	0.12	0.99	0.13	0.10	0.02
MiniCPM-LLaMA3-V2.5 (Yao et al., 2024)	8B	0.04	1.00	0.02	0.02	0.00
MiniCPM-V2.6 (Yao et al., 2024)	8B	0.11	0.87	0.12	0.10	0.17
Qwen-VL (Bai et al., 2023)	9B	0.08	0.99	0.03	0.00	0.00
GLM-4v (GLM et al., 2024)	13B	0.02	0.92	0.25	0.00	0.00
LLaVA-v1.6-Vicuna (Liu et al., 2024a)	13B	0.00	0.74	0.53	0.00	0.02
CogVLM2-LLaMA3 (Wang et al., 2023)	19B	0.43	0.33	0.00	0.13	0.00
InternVL2 (Chen et al., 2023b)	26B	0.39	0.84	0.35	0.00	0.10
LLaVA-NEXT (Liu et al., 2024a)	72B	0.07	0.85	0.31	0.07	0.00
InternVL2 (Chen et al., 2023b)	76B	0.33	0.92	0.25	0.10	0.08
Closed-Source MLLMs						
Qwen-VL-Max (Bai et al., 2023)	-	0.15	0.78	0.50	0.14	0.36
Claude-3-Haiku (Anthropic, 2024a)	-	0.10	0.77	0.46	0.04	0.01
Claude-3.5-Sonnet (Anthropic, 2024b)	-	0.35	0.48	0.64	0.21	0.11
Gemini-Pro-1.5 (Reid et al., 2024)	-	0.43	0.55	0.63	0.18	0.13
GPT-4o-mini (OpenAI, 2024a)	-	0.09	0.46	0.62	0.31	0.13
GPT-4o (OpenAI, 2024b)	-	0.46	0.50	0.64	0.09	0.46
Human Performance						
Human	-	0.67	0.76	0.48	0.35	0.54

Table 4: Comparison of open-source and closed-source MLLM performance (**recall** in percentage) across error detection tasks. We also denote **VIS**, **CAL**, **REAS**, **KNOW**, and **MIS** for visual perception error, calculation error, reasoning error, knowledge error, and misinterpretation of the question. The highest and second highest scores among MLLMs in each column are highlighted in red and blue, respectively.

Task Definition: You are an education expert proficient in K-12 mathematics. Your task is to identify the first step where the mistake occurred in the incorrect answer reasoning steps based on the following mathematical question (including the textual and visual parts), reference answer, and incorrect answer.

Output format:
Error Step: Step X

Below is the reference content you need to identify the error step:
Question Image: {image}
Question Text: {content}
Correct Answer: {answer}
Incorrect Answer: {user_answer}
Incorrect Answer Reasoning Steps:{user_answer_steps}

Instruction: Please provide the corresponding error step identification in the format "Error Step: Step X", without any additional content.

Figure 19: Prompt for error step identification task.

Multimodal Large Language Models	Parameters	VIS	CAL	REAS	KNOW	MIS
Open-Source MLLMs						
InternVL2 (Chen et al., 2023b)	2B	0.11	0.43	0.43	0.00	0.11
Phi-3-vision (Abdin et al., 2024)	4B	0.40	0.40	0.65	0.26	0.22
Yi-VL (Young et al., 2024)	6B	0.23	0.37	0.43	0.05	0.00
DeepSeek-VL (Lu et al., 2024a)	7B	0.16	0.41	0.44	0.10	0.11
LLaVA-v1.6-Vicuna (Liu et al., 2024a)	7B	0.21	0.42	0.45	0.00	0.05
InternVL-2 (Chen et al., 2023b)	8B	0.62	0.41	0.61	0.33	0.33
MiniCPM-LLaMA3-V2.5 (Yao et al., 2024)	8B	0.61	0.37	0.51	0.15	0.00
MiniCPM-V2.6 (Yao et al., 2024)	8B	0.54	0.42	0.55	0.08	0.13
Qwen-VL (Bai et al., 2023)	9B	0.39	0.39	0.36	0.00	0.25
GLM-4v (GLM et al., 2024)	13B	0.76	0.41	0.52	0.00	0.00
LLaVA-v1.6-Vicuna (Liu et al., 2024a)	13B	0.00	0.48	0.47	0.00	0.37
CogVLM2-LLaMA3 (Wang et al., 2023)	19B	0.13	0.39	0.26	0.04	0.00
InternVL2 (Chen et al., 2023b)	26B	0.53	0.48	0.57	0.33	0.44
LLaVA-NEXT (Liu et al., 2024a)	72B	0.66	0.42	0.51	0.45	0.08
InternVL2 (Chen et al., 2023b)	76B	0.55	0.45	0.66	0.56	0.52
Closed-Source MLLMs						
Qwen-VL-Max (Bai et al., 2023)	-	0.81	0.53	0.57	0.36	0.24
Claude-3-Haiku (Anthropic, 2024a)	-	0.60	0.45	0.52	0.16	0.40
Claude-3.5-Sonnet (Anthropic, 2024b)	-	0.61	0.56	0.52	0.13	0.46
Gemini-Pro-1.5 (Reid et al., 2024)	-	0.73	0.56	0.55	0.41	0.37
GPT-4o-mini (OpenAI, 2024a)	-	0.90	0.54	0.51	0.08	0.27
GPT-4o (OpenAI, 2024b)	-	0.80	0.61	0.55	0.44	0.17
Human Performance						
Human	-	0.64	0.68	0.64	0.23	0.38

Table 5: Comparison of open-source and closed-source MLLM performance (**precision** in percentage) across error detection tasks. We also denote **VIS**, **CAL**, **REAS**, **KNOW**, and **MIS** for visual perception error, calculation error, reasoning error, knowledge error, and misinterpretation of the question. The highest and second highest score among MLLMs in each column are highlighted in red and blue, respectively.

Multimodal Large Language Models	Parameters	VIS	CAL	REAS	KNOW	MIS
Open-Source MLLMs						
InternVL2 (Chen et al., 2023b)	2B	0.16	0.40	0.19	0.00	0.15
Phi-3-vision (Abdin et al., 2024)	4B	0.15	0.57	0.12	0.05	0.06
Yi-VL (Young et al., 2024)	6B	0.13	0.50	0.08	0.08	0.00
DeepSeek-VL (Lu et al., 2024a)	7B	0.07	0.57	0.00	0.15	0.08
LLaVA-v1.6-Vicuna (Liu et al., 2024a)	7B	0.28	0.22	0.14	0.00	0.09
InternVL-2 (Chen et al., 2023b)	8B	0.20	0.59	0.22	0.16	0.04
MiniCPM-LLaMA3-V2.5 (Yao et al., 2024)	8B	0.07	0.54	0.04	0.04	0.00
MiniCPM-V2.6 (Yao et al., 2024)	8B	0.18	0.56	0.19	0.08	0.15
Qwen-VL (Bai et al., 2023)	9B	0.14	0.56	0.06	0.00	0.01
GLM-4v (GLM et al., 2024)	13B	0.04	0.57	0.34	0.00	0.00
LLaVA-v1.6-Vicuna (Liu et al., 2024a)	13B	0.00	0.58	0.50	0.00	0.04
CogVLM2-LLaMA3 (Wang et al., 2023)	19B	0.21	0.36	0.01	0.06	0.00
InternVL2 (Chen et al., 2023b)	26B	0.45	0.61	0.44	0.01	0.17
LLaVA-NEXT (Liu et al., 2024a)	72B	0.12	0.57	0.39	0.12	0.01
InternVL2 (Chen et al., 2023b)	76B	0.41	0.60	0.36	0.18	0.14
Closed-Source MLLMs						
Qwen-VL-Max (Bai et al., 2023)	-	0.25	0.20	0.53	0.25	0.29
Claude-3-Haiku (Anthropic, 2024a)	-	0.17	0.57	0.49	0.06	0.03
Claude-3.5-Sonnet (Anthropic, 2024b)	-	0.45	0.51	0.58	0.16	0.18
Gemini-Pro-1.5 (Reid et al., 2024)	-	0.54	0.56	0.58	0.25	0.19
GPT-4o-mini (OpenAI, 2024a)	-	0.16	0.50	0.56	0.13	0.17
GPT-4o (OpenAI, 2024b)	-	0.58	0.55	0.59	0.15	0.25
Human Performance						
Human	-	0.65	0.71	0.55	0.28	0.44

Table 6: Comparison of open-source and closed-source MLLM performance (**F1** in percentage) across error detection tasks. We also denote **VIS**, **CAL**, **REAS**, **KNOW**, and **MIS** for visual perception error, calculation error, reasoning error, knowledge error, and misinterpretation of the question. The highest and second highest score among MLLMs in each column are highlighted in red and blue, respectively.

Task Definition: You are an education expert proficient in K-12 mathematics. Your task is to identify the category of error for the incorrect answer based on the following question (including the textual and visual parts), reference answer, and incorrect answer. The error should belong to one of the following categories: Visual Perception Error, Reasoning Error, Knowledge Error, Calculation Error, or Misinterpretation of the Question.

Output format:
Error Category: Clearly indicate which error category it belongs to.

The definitions of the error categories are as follows:
⋆Visual Perception Error: Failure to accurately obtain information from the images or charts in the question due to visual issues, leading to errors.
⋆Reasoning Error: Improper reasoning during the problem-solving process, failure to correctly apply logical relationships or draw conclusions, leading to errors
⋆Knowledge Error: Errors occur when applying relevant knowledge points due to incomplete or incorrect understanding of knowledge.
⋆Calculation Error: Errors occur in the calculation process, such as addition, subtraction, multiplication, division mistakes, or unit conversion errors, or errors in numerical symbols between multiple steps.
⋆Misinterpretation of the Question: Failure to correctly understand the requirements of the question or misinterpreting the meaning of the question stem, leading to an irrelevant answer, such as answering with numbers when letters are required, and vice versa.

Below is the reference content you need to identify the error step:
Question Image: {image}
Question Text: {content}
Correct Answer: {answer}
Incorrect Answer: {user_answer}
Incorrect Answer Reasoning Steps:{user_answer_steps}

Instruction: Please provide the corresponding error category in the format "Error Category: X", without any additional content.

Figure 20: Prompt for error categorization task.

MLLMs	Source	URL
InternVL2-2B	local checkpoint	https://huggingface.co/OpenGVLab/InternVL2-2B
InternVL2-8B	local checkpoint	https://huggingface.co/OpenGVLab/InternVL2-8B
InternVL2-26B	local checkpoint	https://huggingface.co/OpenGVLab/InternVL2-26B
InternVL2-76B	local checkpoint	https://huggingface.co/OpenGVLab/InternVL2-Llama3-76B
Phi-3-vision-4B	local checkpoint	https://huggingface.co/microsoft/Phi-3-vision-128k-instruct
Yi-VL-6B	local checkpoint	https://huggingface.co/01-ai/Yi-VL-6B
DeepSeek-VL-7B	local checkpoint	https://huggingface.co/deepseek-ai/deepseek-vl-7b-chat
LLaVA-v1.6-Vicuna-7B	local checkpoint	https://huggingface.co/llava-hf/llava-v1.6-vicuna-7b-hf
LLaVA-v1.6-Vicuna-13B	local checkpoint	https://huggingface.co/llava-hf/llava-v1.6-vicuna-13b-hf
LLaVA-NEXT-72B	local checkpoint	https://huggingface.co/llava-hf/llava-next-72b-hf
MiniCPM-V2.5-8B	local checkpoint	https://huggingface.co/openbmb/MiniCPM-Llama3-V-2_5
MiniCPM-V2.6-8B	local checkpoint	https://huggingface.co/openbmb/MiniCPM-V-2_6
Qwen-VL-9B	local checkpoint	https://huggingface.co/Qwen/Qwen-VL-Chat
GLM-4v-13B	local checkpoint	https://huggingface.co/THUDM/glm-4v-9b
CogVLM2-19B	local checkpoint	https://huggingface.co/THUDM/cogvlm2-llama3-chat-19B
Qwen-VL-Max	qwen-vl-max-0809	https://modelscope.cn/studios/qwen/Qwen-VL-Max
Claude-3-Haiku	claude-3-haiku	https://www.anthropic.com/api
Claude-3.5-Sonnet	claude-3-5-sonnet	https://www.anthropic.com/api
Gemini-Pro-1.5	gemini-1.5-pro-latest	https://deepmind.google/technologies/gemini/pro/
GPT-4o-mini	gpt-4o-mini-2024-07-18	https://platform.openai.com/docs/models/gpt-4o-mini
GPT-4o	gpt-4o-2024-08-06	https://platform.openai.com/docs/models/gpt-4o

Table 7: Sources of our evaluated MLLMs.

G.5 Detailed Actionable Suggestions

Finding #1: Closed-source MLLMs outperform open-source MLLMs
Actionable Suggestion: In order to improve the performance of open-source MLLMs, it is crucial to focus on distilling the error detection capabilities of closed-source models (Hsieh et al., 2023; Liang et al., 2024a). One effective approach is to use a *teacher-student framework* in which the more powerful closed-source model acts as the teacher and the open-source model as the student. This method allows the open-source models to learn from the strengths of closed-source models, particularly in error detection tasks including error step identification and error categorization. Moreover, open-source models should leverage datasets used by closed-source models and augment their training process to better mimic the proprietary training regimens of these models (Liang et al., 2024a; Aslam et al., 2024). To optimize performance, fine-tuning should be guided by the insights from closed-source models, including how these models handle complex error categories and balance their performance across various error types. This approach will ensure a more robust open-source model, capable of addressing the current performance gaps.

Finding #2: Open-source MLLMs over-predict CAL category
Actionable Suggestion: The tendency for open-source MLLMs to over-predict the CAL (Calculation Error) category is an issue that arises from their bias towards easier tasks. To address this, models should be regularized during training to reduce their preference for simpler categories like CAL. This can be achieved through the use of *weighted loss functions*, such as *Focal Loss* (Li et al., 2022) or *AdaFocal* (Ghosh et al., 2022), which down-weight easier categories and force the model to focus on more challenging ones. Additionally, introducing *class balancing techniques* such as oversampling the underrepresented categories (e.g., REAS, KNOW) or undersampling the CAL category can further help in addressing this bias (Ghosh et al., 2024). Another key approach is data augmentation, where the variety and complexity of error cases are increased, particularly for the more challenging categories. This will ensure that the model learns to classify all types of errors more evenly, avoiding an over-reliance on CAL (Iqbal et al., 2024). Finally, *meta-learning* techniques can be employed to dynamically adjust the model's bias toward different categories during training, allowing the model to better adapt to different error types (Vettoruzzo et al., 2024).

Finding #3: STEP tasks are easier than CATE tasks
Actionable Suggestion: The disparity in perfor-

mance between STEP and CATE tasks suggests that MLLMs need to be trained to better handle the complexity of error categorization. Since STEP tasks primarily involve localizing specific errors, which is conceptually simpler than categorizing them, it is crucial to build a stronger relationship between the two tasks in the training process. One effective method is to use *multi-task learning*, where the model is simultaneously trained on both STEP and CATE tasks, allowing it to learn not only how to localize errors but also how to classify them accurately (Chen et al., 2024; Xin et al., 2024). Additionally, *contrastive learning* can be used to distinguish between similar error steps and categories, improving the model's ability to reason about the relationship between error localization and categorization. Training data should also be designed to emphasize this relationship, ensuring that the model can learn the necessary contextual understanding to categorize errors correctly. By focusing on these aspects, MLLMs will be better equipped to handle the more complex task of error categorization (Hu et al., 2024a).

Finding #4: CAL is the easiest category, while KNOW is the hardest
Actionable Suggestion: The difficulty gap between CAL and KNOW errors highlights the need for specialized strategies to handle knowledge errors. Since CAL errors are generally more deterministic and easy to identify, while KNOW errors require deeper understanding and reasoning, MLLMs should incorporate domain-specific knowledge to improve their performance in this category. One approach is to integrate external knowledge bases or knowledge graphs into the training dataset, providing the model with richer, contextually relevant information. This will help the model recognize errors related to factual inaccuracies or incomplete reasoning (Sun et al., 2023; Pan et al., 2024). Additionally, knowledge-intensive reasoning models can be introduced to simulate more advanced human-like problem-solving capabilities. Techniques such as external validation of logical consistency can also be applied to better identify and rectify knowledge errors. Furthermore, few-shot learning methods can be used to allow MLLMs to generalize from limited examples, especially for rare or complex knowledge errors that are more difficult to detect (Ma et al., 2023a,b). By improving the model's access to domain-specific knowledge and reasoning tools, its ability to handle KNOW errors will be significantly enhanced.

Finding #5: Gap to human-level performance in error detection
Actionable Suggestion: To bridge the gap between human-level performance and MLLM performance, particularly in tasks such as Visual Perception Errors (VIS) and Reasoning Errors (REAS), MLLMs need to be trained to better mimic human cognitive processes. One promising approach is to employ *Reinforcement Learning from Human Feedback (RLHF)*, where human evaluators guide the model by providing corrective feedback and insights into error causes (Wang et al., 2024a; Kirk et al., 2023). This will help the model align more closely with human reasoning mechanisms, particularly in tasks that require higher-level cognitive functions. In addition, models can be trained to simulate human visual perception by integrating attention mechanisms and vision-language models that enable more sophisticated visual error detection. Incorporating logical reasoning modules into the model will also improve its performance in REAS, allowing it to understand the logical flow of the problem and detect reasoning errors more effectively. Finally, cross-modal alignment between text and image modalities will ensure that MLLMs process visual and textual inputs in a more integrated and human-like manner, thereby improving performance in VIS error detection (Shen et al., 2023). By aligning MLLMs more closely with human cognitive processes, it is possible to achieve significant improvements in error detection tasks and approach human-level performance.

Finding #6: Best Generalist models outperform specialized ones
Actionable Suggestion: To enhance the performance of specialized reasoning and math models on complex multimodal error analysis tasks, developers should prioritize strategies that broaden their contextual understanding and error analysis capabilities beyond their narrow domain. One key action is to augment their training datasets with a more diverse range of multimodal examples that explicitly feature varied error types and require nuanced, cross-modal reasoning for identification (Li et al., 2023; Yin et al., 2024; You et al., 2024). This could involve curating or synthetically generating data that forces models to not just solve a problem,

but to also analyze and explain potential errors in presented solutions, mirroring the capabilities of generalist models. Furthermore, incorporating instruction tuning with fine-grained error analysis prompts, similar to how general-purpose visual language models are trained (Liu et al., 2023), can help specialized models develop a more robust understanding of error patterns. Finally, exploring hybrid architectures or ensemble methods, where a specialized model's deep domain knowledge is guided or supplemented by a generalist model's broader contextual awareness and error-spotting acumen, could offer a practical path to improved performance without sacrificing specialization entirely (Xu et al., 2025; Bi et al., 2025; Yan et al., 2025b).

G.6 CAL and non-CAL Distribution of MLLMs

In this section, we indicate the distribution of CAL and non-CAL category predictions of 21 representative MLLMs, as shown in Figure 21. It can be seen that there is a bias towards CAL category among most open-source MLLMs, while closed-source ones except for Claude-3-Haiku and Qwen-VL-Max do not have such a bias for error categorization task.

G.7 Visual Perception Analysis

Finding #1: Closed-source MLLMs are most likely to misjudge VIS as REAS in error categorization task. Taking GPT-4o model as an example, as shown in the Figure 22, 48% of VIS are misclassified as REAS, followed by 30% being misjudged as MIS. When MLLM needs to handle information involving both visual and linguistic elements simultaneously, if an erroneous response to a math query originates from VIS, it mistakenly attributes this to a flaw in logical reasoning that occurs subsequent to initial visual misinterpretation.

Finding #2: Open-source MLLMs are more likely to misclassify VIS as CAL. Taking the open-source model CogVLM2-LLaMA3, which performs best in identifying VIS, as an example, CAL accounts for 64% of misclassified category, as illustrated in the Figure 22. When handling complex visual information, especially in geometry problems, the MLLM often struggles to accurately extract key features. Due to the open-source MLLM's weaker multimodal integration capabilities, it simplifies visual issues into numerical calculation problems. The lack of sufficient training and data for visual-related errors is also a key reason behind this phenomenon (Wichmann and Geirhos, 2023). See more analysis on misclassification for each category and visual perception case study in Appendix G.8 and G.9.

G.8 Analysis of Confusion Matrix for CATE task

Figures 23 and 24 present the confusion matrices for InternVL2-76B and GPT-4o, two MLLMs evaluated on five error categories. The matrices show the count of predictions for each category, with diagonal entries representing correct predictions and off-diagonal entries indicating misclassifications. These visualizations provide insights into each model's strengths and weaknesses.

InternVL2-76B shows strong performance in detecting CAL, with 843 correct predictions, indicating its robust numerical reasoning capability. However, the model struggles to distinguish between REAS and CAL, misclassifying 626 REAS instances as CAL. This confusion suggests an over-reliance on numerical features and an inability to separate logical reasoning tasks from computational ones. Additionally, there is significant misclassification of VIS into CAL, with 244 cases, highlighting a potential weakness in integrating visual and textual modalities. These trends may stem from InternVL2-76B's limited domain-specific reasoning ability.

GPT-4o, on the other hand, demonstrates relatively good performance in VIS, with 183 correct predictions, significantly outperforming InternVL2-76B. Its capability in REAS is also notable, with 617 correct predictions, suggesting a more balanced reasoning ability. However, GPT-4o struggles more with CAL, achieving only 460 correct predictions, and shows significant confusion between CAL and REAS, with 299 CAL instances misclassified as REAS. Furthermore, the model has difficulty with MIS, misclassifying 45 MIS cases as REAS, pointing to challenges in identifying nuanced interpretational issues. These trends suggest that GPT-4o's emphasis on multimodal alignment and contextual understanding contributes to its strengths in VIS and REAS but comes at the expense of CAL performance.

Comparing the two models reveals distinct strengths and weaknesses. GPT-4o significantly outperforms InternVL2-76B in VIS, likely due to superior multimodal visual-text alignment capabilities. Both models exhibit confusion between

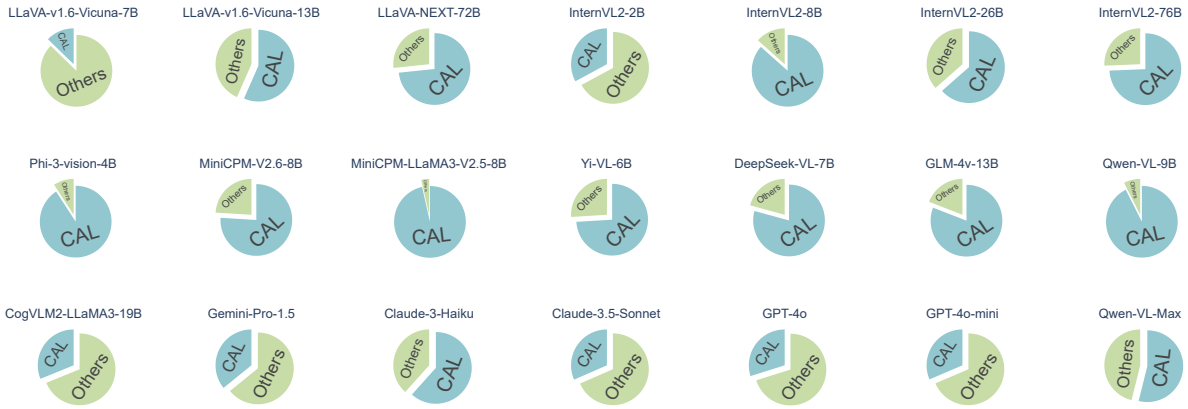


Figure 21: Distribution of CAL and non-CAL category predictions of all MLLMs we evaluate.

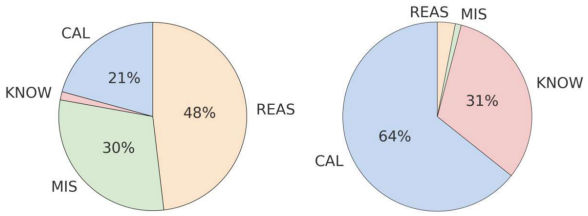


Figure 22: The error category distribution of misjudged VIS cases of GPT-4o (left) and CogVLM2-LLaMA3 (right).

REAS and CAL, but GPT-4o shows a more balanced classification ability in REAS. MIS remains a challenging category for both models, though GPT-4o struggles slightly more in distinguishing it from REAS. These differences may arise from variations in model architecture and training objectives. This analysis underscores the complementary strengths of these models: InternVL2-76B excels in numerical reasoning, while GPT-4o performs better in visual perception and logical reasoning. Future research could explore ways to integrate their strengths for a more robust multimodal error detection system.

G.9 Visual Bad Cases Predicted by GPT-4o

Visual perception errors are critical in multimodal error detection tasks, as they impact the accurate comprehension of mathematical problems presented with both text and diagrams. As illustrated in Figures 25, 26, 27, 28 and 29, the five primary categories of visual errors observed in GPT-4o (the MLLM with best overall and VIS performance) include **distance perception**, **diagram perception**, **spatial perception**, **flip/fold perception**, and **shape perception**. These categories differ in their cognitive demands: distance perception focuses on point identification; diagram perception on quantitative estimation; spatial perception on geometric visualization; flip/fold perception on mental rotation; and shape perception on object classification (Lu et al., 2024b; Zhang et al., 2024b). Detecting such errors is challenging because they often require both intricate visual processing and precise interpretation of mathematical relations, which can be difficult to encode in current MLLMs. To overcome these challenges, future MLLMs should incorporate more advanced visual reasoning capabilities, possibly through enhanced alignment between vision and language modalities, enabling better detection and correction of complex perception errors (Song et al., 2023). This could significantly improve the robustness of MLLMs in mathematical and other perception-heavy tasks.

G.10 Relation between Error Category and Error Step

Finding #1: There is a close relationship between different error category and their distribution in the reasoning steps. As shown in Figure 30, VIS tends to occur in the earlier to mid-stages, accounting for a median proportion of 0.5 of total steps. In contrast, MIS, REAS, CAL, and KNOW are more likely to arise in the later stages, with their median proportions ranging from 0.7 to 0.9. More analysis of this relationship across MLLMs in terms of cognitive load analysis can be seen in Appendix G.11.

Finding #2: VIS occurs in the earlier stages of problem-solving reasoning. This finding could be closely linked to the sequence in which students approach the task (Binz and Schulz, 2023; Kennedy and Romig, 2024). Since image content often serves as a key reference early on, any misinterpretation of this visual information directly impacts the subsequent problem-solving steps. Stu-

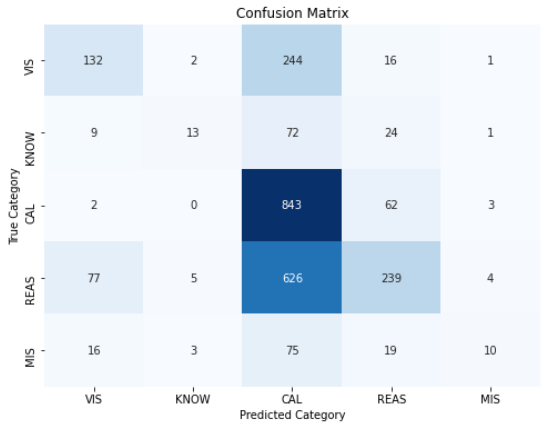


Figure 23: The confusion matrix of five error categories predicted by InternVL2-76B, the open-source MLLM with the best overall performance on error detection.

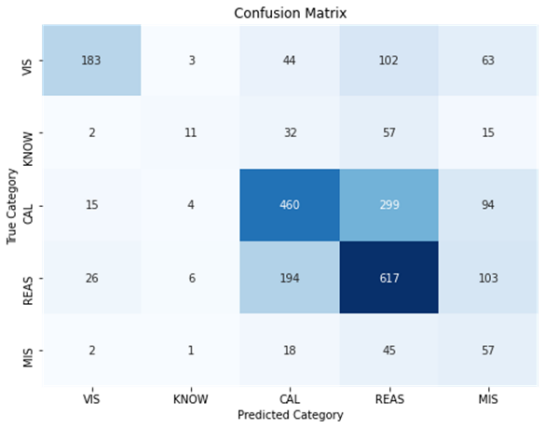
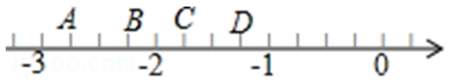


Figure 24: The confusion matrix of five error categories predicted by GPT-4o, the closed-source MLLM with the best overall performance on error detection.

Distance Perception

Question: As shown in the figure, the point on the number line representing -2.75 is point ___.



True Answer: A

Student's Answer: B

Reasoning Steps:

◆ Step 1: Read the question and the text on the number line, which states "The point representing -2.75 on the number line is _____ point."
◆ Step 2: Determine the position of the point representing -2.75 on the number line.
◆ Step 3: Identify from the diagram the values represented by points A, B, C, and D.
◆ Step 4: Establish that point B is near the position representing -3, point A represents -2.25, point C represents -1.75, and point D represents -1.25.
◆ Step 5: Based on the number line position, conclude that the point representing -2.75 is point B.
◆ Step 6: Output the result as B.

Err. Cate. by GPT-4o: Reasoning Err.

Figure 25: Distance bad case where GPT-4o predicts visual perception errors incorrectly.

Diagram Perception

Question: To understand the growth situation of a certain type of flower in a botanical garden, a sample of 200 flowers was randomly selected from a plot with approximately 3000 such flowers, and their heights were measured. After organizing the statistical results, the data is listed as follows: (each group of data may include the minimum value but does not include the maximum value) Estimate the number of such flowers in the plot with a height less than 55 cm and not less than 45 cm to be about _____ flowers.

True Answer: 960

Student's Answer: 1440

Height(cm)	40~45	45~50	50~55	55~60	60~65	65~70
Frequency	33	42	22	24	43	36

Reasoning Steps:

◆ Step 1: Read the problem statement and chart text "Estimate the number of such flowers in the garden with a height less than 55 cm and not less than 45 cm is about _____ plants."
◆ Step 2: Determine the total sample size as 200 plants.
◆ Step 3: Determine the total number of flowers as 3000 plants.
◆ Step 4: Read from the chart that the number of flowers with a height of 45-50 cm is 42 plants.
◆ Step 5: Read from the chart that the number of flowers with a height of 50-55 cm is 32 plants.
◆ Step 6: Calculate the number of flowers with a height between 45-55 cm as 42 + 32 = 72 plants.
◆ Step 7: Calculate the proportion of flowers with a height between 45-55 cm in the sample as 72 / 200 = 0.36.
◆ Step 8: Multiply the total number of flowers, 3000 plants, by the proportion in the sample, 0.36, to calculate approximately 3000 × 0.36 = 1440 plants.
◆ Step 9: Output the result as 1440.

Err. Cate. by GPT-4o: Calculation Err.

Figure 26: Diagram bad case where GPT-4o predicts visual perception errors incorrectly.

dents typically first examine the image, and then integrate the information before proceeding to reasoning or calculation. As a result, visual perception errors arise earlier compared to other types of errors.

Finding #3: Other error categories are primarily in later stages of problem-solving reasoning. This may be linked to the increasing cognitive load students encounter during problem-solving. Cognitive Load Theory posits that infor-

Spatial Perception

Question: Using small cubes with an edge length of 1 cm to form the geometric shape shown in the figure, place this shape on a table and paint the exposed surfaces with color (excluding the bottom surface), then the total area of the painted surfaces is _____ cm².
True Answer: 27
Student's Answer: 41
Reasoning Steps:
◆ Step 1: Read the problem statement and the text in the diagram, "The geometric shape shown is made up of small cubes with an edge length of 1 cm. Place this geometric shape on the table and paint the exposed surfaces (excluding the bottom surface), then the total area of the painted surfaces is ___1___."
◆ Step 2: Observe the structure of the geometric shape and confirm it is composed of small cubes.
◆ Step 3: From the front view, count the number of small squares, totaling 6.
◆ Step 4: From the left view, count the number of small squares, totaling 6.
◆ Step 5: From the top view, count the number of small squares, totaling 8.
◆ Step 6: From the right view, count the number of small squares, totaling 6.
◆ Step 7: From the back view, count the number of small squares, totaling 6.
◆ Step 8: Add the number of small squares from each face, totaling 6 + 6 + 8 + 6 + 6 = 32.
◆ Step 9: Count the number of small squares on the bottom face, totaling 9.
◆ Step 10: Add the number of small squares on the bottom face to the previous total, 32 + 9 = 41.
◆ Step 11: Calculate the total area of the painted surfaces, which is the area of 41 small squares.
◆ Step 12: Output the result as 41.
Err. Cate. by GPT-4o: Misinterpretation of the Question

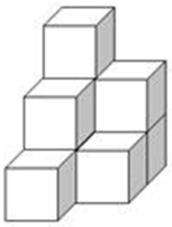


Figure 27: Spatial bad case where GPT-4o predicts visual perception errors incorrectly.

Flip/Fold Perception

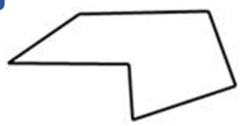
Question: A number is shown in the mirror as illustrated, then the actual number is _____.
True Answer: 2502
Student's Answer: 2505
Reasoning Steps:

◆ Step 1: Read the problem statement and the text in the image: "The number in the mirror is shown in the figure."
◆ Step 2: Observe that the number in the mirror is "5025."
◆ Step 3: Based on the properties of mirror symmetry, the number seen in the mirror is actually flipped horizontally and vertically.
◆ **Step 4: Flip the number "2502" horizontally to get the actual number "2505."**
◆ Step 5: Output the result as 2505.
Err. Cate. by GPT-4o: Reasoning Err.

Figure 28: Flip & fold bad case where GPT-4o predicts visual perception errors incorrectly.

Shape Perception

Question: True or False: The following figure is a convex polygon___. (Fill in "True" or "False")
True Answer: True
Student's Answer: False



Reasoning Steps:
◆ Step 1: Read the problem statement and the text on the image: "True or False: The following shape is a convex polygon _____. (Fill in 'True' or 'False')".
◆ Step 2: Observe the accompanying image.
◆ Step 3: Determine if the shape meets the characteristics of a convex polygon based on its definition.
◆ Step 4: Through observation, confirm that the shape is a convex polygon.
◆ Step 5: Conclude that the shape is a convex polygon.
◆ Step 6: Output the result as "True".
Err. Cate. by GPT-4o: Misinterpretation of the Question

Figure 29: Shape bad case where GPT-4o predicts visual perception errors incorrectly.

mation complexity ranges from low to high interactivity (Paas et al., 2010; Binz and Schulz, 2023). While low-interactivity information can be understood independently, high-interactivity information requires simultaneous processing of related elements, thus increasing cognitive load (Kennedy and Romig, 2024; Abbad-Andaloussi et al., 2023). In later stages, students must integrate complex information from multiple sources. For instance, calculating the distance between two points needs increasing interactivity heightens cognitive load, leading to errors like forgetting to take the square root or miscalculating differences. Consequently, as cognitive load rises, the frequency of errors in later steps also increases.

G.11 Cognitive Load Analysis Across MLLMs

In analyzing the error step distribution for the multimodal error detection task using InternVL2-76B (see Figure 31) and GPT-4o (see Figure 32), we observe a consistency in the pattern of error category distribution across both MLLM's predictions and those in ErrorRadar (see Figure 30). In

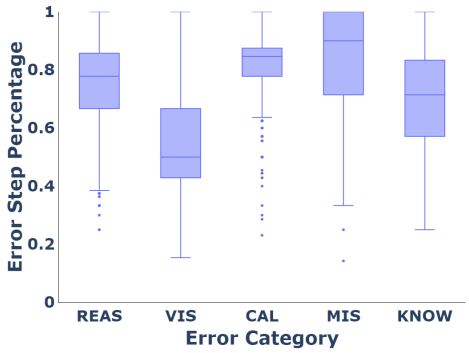


Figure 30: The error step distribution (in percentage) of error categories in ErrorRadar dataset.

particular, VIS tends to occur in the earlier stages of problem-solving for both MLLMs, which aligns with the sequence in which students typically approach tasks. Since visual content often serves as a key reference at the outset, any misinterpretation of this information can significantly impact subsequent steps. Students generally examine the image first and then integrate the information before proceeding to reasoning or calculation, leading to visual perception errors arising earlier compared to other types of errors.

Other error categories, such as REAS, CAL, MIS, and KNOW, are more likely to emerge in the later stages of problem-solving. This pattern is linked to the increasing cognitive load students encounter as they progress. According to Cognitive Load Theory, information complexity ranges from low to high interactivity. Low-interactivity information can be understood independently, whereas high-interactivity information requires the simultaneous processing of related elements, thereby increasing cognitive load. In the later stages, students must integrate complex information from multiple sources, which can lead to errors like forgetting to take the square root or miscalculating differences when calculating distances, for example. Consequently, the frequency of errors in later steps increases with the rising cognitive load.

Despite the overall pattern being consistent, there may be subtle differences between InternVL2-76B and GPT-4o in terms of error step distribution, especially for MIS category. These differences could be attributed to the models' distinct architectures and training data, which might influence their approaches to error detection. As an open-source MLLM, InternVL2-76B might not have been optimized for specific types of questions or educational contexts, which could lead to a higher variability in MIS.

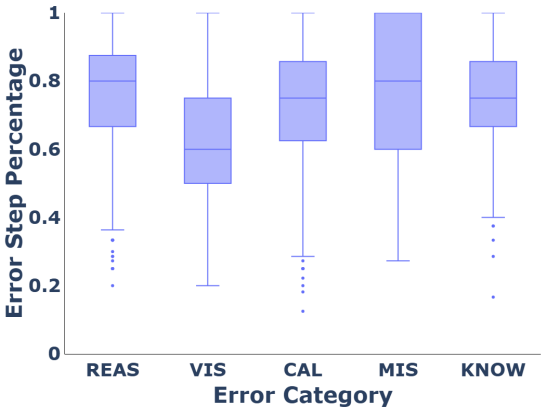


Figure 31: The error step distribution (in percentage) of error categories predicted by InternVL2-76B, the open-source MLLM with the best overall performance on error detection.

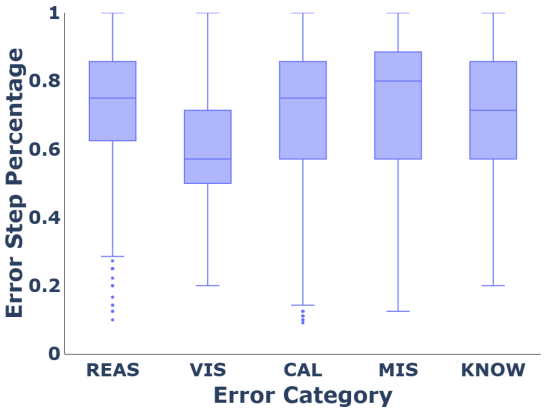


Figure 32: The error step distribution (in percentage) of error categories predicted by GPT-4o, the closed-source MLLM with the best overall performance on error detection.

H Clarification of LLM Usage

In the spirit of transparency, we clarify that Gemini-Pro-2.5 was utilized in the preparation of this manuscript. Its use was strictly limited to language polishing, including grammar correction, syntax refinement, and improving the overall fluency of the text. The LLM did not contribute to any of the core scientific aspects of this work. The conceptualization of the ErrorRadar benchmark, the experimental design, the data analysis, and the interpretation of the results are entirely the original work of the human authors, who retain full responsibility for the intellectual content and integrity of this paper.