DEEPVISUALMATH: COMPREHENSIVE EVALUATION OF VISUAL MATHEMATICAL REASONING IN LARGE MULTIMODAL MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Visual mathematical reasoning requires integrating visual evidence with mathematical concepts, yet its evaluation in multimodal large language models (MLLMs) remains fragmented across benchmarks targeting limited domains and skills. We introduce DEEPVISUALMATH, a unified benchmark and evaluation framework for holistic assessment of visual mathematical reasoning. The benchmark comprises **10,298 questions**, organized into **7 categories, 18 subcategories, and 80 tasks**, spanning applied sciences, continuous mathematics, relational geometry across dimensions, discrete mathematics and formal logic, plane geometry, contextual mathematics, and solid geometry. Beyond the benchmark, we provide an automated curation pipeline, a reproducible evaluation harness, and an automated error analysis framework for detailed diagnosis across the task hierarchy. Evaluation of 10 MLLMs reveals substantial and uneven performance gaps: overall accuracy ranges from 15.50% to 66.67%, with even the strongest model leaving approximately one third of questions unresolved. Performance also varies markedly across mathematical domains; the leading model achieves 77.36% in continuous mathematics but only 51.85% in plane geometry, highlighting limitations obscured by aggregate scores. These findings demonstrate that strong performance in one mathematical domain does not imply broad visual mathematical competence. DEEPVISUALMATH provides a systematic foundation for evaluating these capabilities and identifying priorities for improving mathematical reasoning grounded in visual evidence.

1 INTRODUCTION

Mathematical reasoning is fundamental to scientific discovery, engineering, education, data analysis, and decision making (Hendrycks et al., 2021; Cobbe et al., 2021). Yet in practice, mathematical concepts rarely appear as plain text alone; they are intrinsically tied to rich visual representations, including geometric constructions, function plots, structured charts, and physical illustrations (Agrawal et al., 2015; Johnson et al., 2016; Masry et al., 2022). Solving such problems requires models to execute a tightly coupled pipeline: isolating visual primitives, inferring spatial and relational structures, grounding visual evidence into formal mathematical representations, and conducting multi-step symbolic or quantitative reasoning (Lu et al., 2023b; Zhang et al., 2024). This interplay between visual perception and formal reasoning demands capabilities that far transcend isolated image understanding or text-based problem solving.

Recent large multimodal models have advanced substantially in visual understanding and instruction following (Alayrac et al., 2022; Li et al., 2023; Liu et al., 2023; Gemini Team, 2023; Wang et al., 2024). However, relying solely on aggregate accuracy masks critical failure modes: a model may recognize a diagram yet misread its scale, extract a plotted point but fail to infer the underlying trend, select the correct formula but plug in incorrect visual quantities, or output a correct answer without relying on the image. Aggregate metrics alone obscure whether errors stem from visual perception, mathematical grounding, symbolic reasoning, or numerical computation—conflating distinct competencies into a single score.

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



Figure 1: **Overview of DEEPVISUALMATH.** Existing benchmarks suffer from fragmented evaluation, incomplete coverage, hidden capability gaps, and limited diagnosis. DEEPVISUALMATH provides a unified taxonomy of 7 categories, 18 subcategories, and 80 tasks, supported by automated curation, evaluation, and error analysis pipelines for fine grained model assessment.

Existing benchmarks offer valuable but fragmented perspectives on visual mathematical reasoning. Standard text-based datasets such as GSM8K and MATH assess symbolic manipulation and competition-level mathematics without visual grounding (Cobbe et al., 2021; Hendrycks et al., 2021). Geometry-focused benchmarks (e.g., Geometry3K, GeoQA, UniGeo, TheoremQA) incorporate diagrams but remain restricted to narrow geometric skills (Lu et al., 2021a; Chen et al., 2021; 2022; 2023). Similarly, chart- and plot-reading benchmarks (DVQA, PlotQA, ChartQA) emphasize visual data extraction rather than mathematical derivation, while broader multimodal benchmarks like ScienceQA and MMMU cover general scientific knowledge (Kafle et al., 2018; Methani et al., 2019; Masry et al., 2022; Lu et al., 2022; Yue et al., 2023b). Although recent efforts such as MathVista, MathVerse, MATH Vision, and WE MATH combine mathematical problems with visual contexts, they rely on disparate taxonomies, task definitions, and evaluation protocols (Lu et al., 2023b; Zhang et al., 2024; Cotilla et al., 2024; Qiao et al., 2025a), making standardized cross-domain comparison and fine-grained diagnosis difficult.

A unified account of visual mathematical competence is therefore still missing. Current evaluations fragment visual mathematics across isolated benchmarks rather than assessing it holistically, leaving major domains – including continuous analysis, relational geometry across dimensions, discrete logic, applied science, 3D spatial reasoning, socio-economic contexts, and knowledge-grounded reasoning – evaluated in isolation or omitted altogether. Furthermore, relying on a single aggregate score conflates distinct reasoning stages: visual perception, domain knowledge, symbolic manipulation, and numerical execution. Evaluating these dimensions separately masks critical interdependencies, such as a model succeeding via textual shortcutting rather than visual grounding, or excelling at symbolic computation while failing on spatial relations. Examining these complementary domains jointly across a consistent evaluation harness is therefore essential to reveal systematic capability gaps and track genuine visual mathematical intelligence.

To address this gap, we introduce DEEPVISUALMATH, a unified benchmark and diagnostic framework for holistic assessment of visual mathematical reasoning in MLLMs (Figure 1). Rather than treating evaluation as a collection of isolated tasks, DEEPVISUALMATH provides an integrated framework that standardizes assessment across perceptual, structural, symbolic, quantitative, and knowledge-grounded reasoning. We make the following contributions:

1. **Comprehensive Taxonomy of Visual Mathematical Reasoning.** We introduce a unified taxonomy with 7 categories, 18 subcategories, and 80 fine-grained tasks spanning continuous, discrete, geometric, applied, and contextual reasoning, enabling analysis beyond a single score.
2. **Automated Curation, Evaluation, and Error Analysis.** We introduce a reproducible pipeline for curating, standardizing, verifying, de-duplicating, and balancing heterogeneous visual mathematics resources, producing a benchmark of 10,298 questions with hierarchical error diagnostics spanning perception, mathematical interpretation, reasoning, computation, and answering.
3. **Large Scale Empirical Study.** We evaluate 10 contemporary MLLMs at category, subcategory, and task levels, revealing systematic capability gaps that remain hidden in narrow benchmarks.
4. **Agent Based Hard Case Evaluation.** We study difficult instances missed by all evaluated VLMs using three agentic setups: *MathSolver*, a multi agent system, a *Claude Code* agent, and

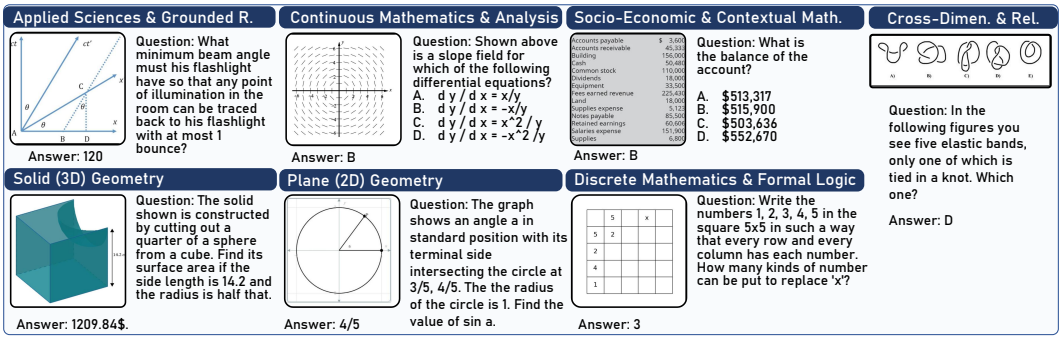


Figure 2: Some representative examples from DEEPVISUALMATH.

a *Codex* agent. Their ability to solve some of these cases suggests that decomposition and tool assisted reasoning can improve performance on challenging visual mathematics problems.

Current MLLMs are not yet uniformly reliable visual mathematical reasoners. Overall accuracy ranges from **15.50%** to **66.67%**, with large differences across domains. Even the strongest model scores **77.36%** on Continuous Mathematics and Analysis but only **51.85%** on Plane Geometry, a gap of over 25 points. Similar variation appears in geometry, discrete mathematics, applied reasoning, and socio economic tasks. Thus, high overall accuracy does not guarantee broad competence. DEEPVISUALMATH offers a common framework for measuring these differences, identifying weaknesses, and guiding more reliable multimodal mathematical reasoning.

2 PRELIMINARIES

Scope and Components. Visual mathematical reasoning is the ability to extract mathematical structure from images and use it to derive a valid answer. It differs from text based reasoning because it requires interpreting diagrams, plots, tables, charts, geometric constructions, and symbolic expressions (Lu et al., 2023b; Zhang et al., 2024; Qiao et al., 2025a). It also goes beyond general visual question answering by requiring mathematical abstraction, symbolic manipulation, quantitative comparison, and valid derivation (Agrawal et al., 2015; Johnson et al., 2016). We represent the underlying visual content as $\mathcal{Z}_i = (\mathcal{E}_i, \mathcal{R}_i, \mathcal{M}_i, \mathcal{C}_i)$, where the four components denote visual entities, their relations, numerical measurements, and mathematical constraints.

Formal Task Model. A visual mathematical instance is $x_i = (I_i, Q_i, A_i^*, m_i)$, where I_i is the visual input, Q_i is the question, A_i^* is the reference answer, and m_i stores task metadata. A multimodal model produces $\widehat{A}_i = f_\theta(I_i, Q_i)$ through the reasoning chain where $\widehat{\mathcal{Z}}_i$ is the grounded mathematical representation and $\widehat{D}_i$ is the derivation or computation. Reliable performance therefore requires visual grounding, mathematical abstraction, compositional derivation, and answer realization, so an error at any stage can invalidate the final response.

Why Partial Evaluation is Insufficient. Existing mathematical benchmarks typically isolate one source of difficulty: MATH and GSM8K focus on symbolic and numerical reasoning (Hendrycks et al., 2021; Cobbe et al., 2021), geometry benchmarks emphasize diagrammatic and theorem based problems (Lu et al., 2021a; Chen et al., 2021; 2022), and chart benchmarks target visual data extraction and comparison (Kafle et al., 2018; Methani et al., 2019; Masry et al., 2022). Although recent visual mathematics benchmarks provide broader coverage, they still differ in taxonomy, source distribution, visual complexity, answer format, and evaluation protocol (Lu et al., 2023b; Zhang et al., 2024; Cotilla et al., 2024; Qiao et al., 2025a). Realistic visual mathematics often requires several capabilities simultaneously: a question about a transformed figure may involve object identification, geometric interpretation, numerical recovery, comparison, and final calculation, while a three dimensional problem may combine perception, dimensional reasoning, algebra, and measurement. Let $\mathcal{C} = \{c_1, \ldots, c_7\}$ denote the top level categories and $\mathcal{S}_i \subseteq \mathcal{C}$ the capabilities required by instance x_i, which is compositional when $|\mathcal{S}_i| \geq 2$. Since aggregate accuracy can hide failures on rare or compositional tasks, we report results at the instance, task, subcategory, category, and overall levels. This hierarchy distinguishes broad mathematical competence from narrow success on particular visual problem types and motivates the design of DEEPVISUALMATH.

A Taxonomy of Visual Mathematical Reasoning. DEEPVISUALMATH organizes visual mathematical reasoning into a four level hierarchy: Category $\longrightarrow$ Subcategory $\longrightarrow$ Task $\longrightarrow$ Instance.

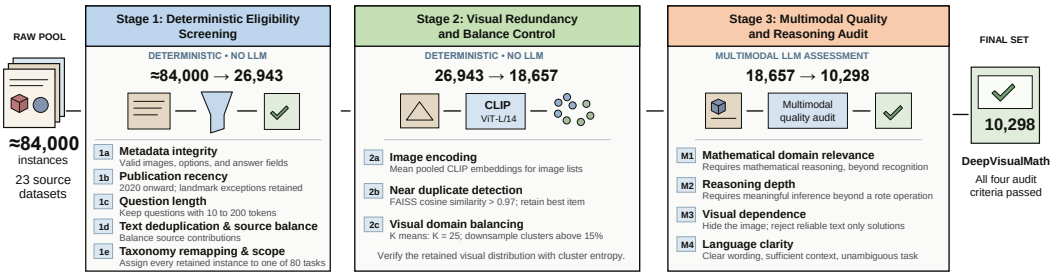


Figure 3: **Automated curation pipeline for DEEPVISUALMATH.** Starting from approximately 84,000 instances across 23 datasets, deterministic eligibility screening, visual de-duplication and balancing, and multi-modal quality auditing retain 10,298 instances, with explicit criteria and logged rejection reasons throughout.

The seven top level categories represent complementary mathematical faculties and visual problem settings. Their distributions are summarized in Table 1 and some examples are provided in Figure 5.

1. **Applied Sciences and Grounded Reasoning** (*Applied*; $n = 324$, 1 subcategory, 5 tasks). This category evaluates mathematical reasoning grounded in scientific, engineering, and real world contexts. Problems may require interpreting physical quantities, scientific measurements, experimental setups, or applied formulas from visual evidence.
2. **Continuous Mathematics and Analysis** (*Continuous*; $n = 1{,}910$, 4 subcategories, 20 tasks). This category covers functions, limits, derivatives, integrals, rates of change, continuity, optimization, and the interpretation of graphs or continuous quantities. It tests whether models can connect visual trends and curves with the underlying analytical structure.
3. **Cross Dimensional and Relational Geometry** (*Cross Dim.*; $n = 3{,}322$, 4 subcategories, 12 tasks). This category focuses on relations that span coordinate systems, dimensions, or geometric representations, including transformations, projections, distances, angles, intersections, correspondences, and relationships between visual objects. It emphasizes structural and relational reasoning beyond a single geometric theorem.
4. **Discrete Mathematics and Formal Logic** (*Discrete*; $n = 1{,}892$, 3 subcategories, 18 tasks). This category includes counting, sequences, sets, combinatorial structures, graphs, patterns, Boolean relations, and proof like deductions. The relevant visual evidence may encode symbolic configurations, finite structures, or logical constraints.
5. **Plane Geometry** (*Plane*; $n = 812$, 2 subcategories, 8 tasks). This category evaluates reasoning over two dimensional Euclidean constructions, including triangles, polygons, circles, angles, lengths, areas, congruence, similarity, and geometric relations represented in plane diagrams.
6. **Socio Economic and Contextual Mathematics** (*Socio Econ.*; $n = 473$, 2 subcategories, 7 tasks). This category covers percentages, ratios, rates, financial quantities, statistics, population information, and other mathematical problems grounded in everyday or socio economic contexts. Solving these tasks requires combining visual information with contextual interpretation.
7. **Solid Three Dimensional Geometry** (*Solid*; $n = 1{,}565$, 2 subcategories, 10 tasks). This category evaluates reasoning about three dimensional objects, including volume, surface area, spatial constructions, nets, cross sections, and relationships among faces, edges, and vertices. It requires models to infer three dimensional structure from two dimensional visual representations.

The distinction between Cross Dimensional and Relational Geometry, Plane Geometry, and Solid Geometry is deliberate. Plane and Solid Geometry focus on domain specific properties and measurements within two and three dimensional objects, respectively. Cross Dimensional and Relational Geometry instead emphasizes transformations, correspondences, projections, and relations that may connect multiple representations or dimensions. Together, these categories separate the geometric object being reasoned about from the relational operations required to interpret it.

3 DEEPVISUALMATH

DEEPVISUALMATH is a large scale, holistic benchmark for evaluating mathematical reasoning in vision language models (VLMs). We construct it by consolidating and curating instances from 20 existing datasets through a three stage automated pipeline (§C). The resulting benchmark contains **10,298 image question answer pairs**, organized in a four level hierarchy comprising **7 categories**, **18 subcategories**, and **80 fine grained tasks**. Its taxonomy spans major dimensions of mathemati-

cal reasoning, including arithmetic and algebraic foundations, continuous analysis, plane and solid geometry, discrete structures, applied sciences, and socio economic quantitative reasoning.

Unlike prior benchmarks that focus on narrower portions of visual mathematical reasoning (Lu et al., 2023a; Zhang et al., 2024; Yue et al., 2023a; Chernyshev et al., 2024), DEEPVISUALMATH provides simultaneous coverage across these complementary capabilities. It supports both multiple choice and open ended formats, enabling evaluation of discriminative accuracy as well as generative mathematical reasoning. As shown in Table 4, DEEPVISUALMATH is the only benchmark in the comparison with coverage across all 18 subcategories in our taxonomy. Representative examples are shown in Figure 5, while additional examples are provided in §L.

3.1 DATA CURATION PIPELINE

We construct DEEPVISUALMATH through a three stage automated curation pipeline (Figure 3, Table 8). The pipeline is designed to be reproducible, scalable, and transferable to other reasoning domains. Starting from 84,000 instances collected from 20 source datasets, we apply increasingly strict quality checks and retain 10,298 high quality instances. Further details are provided in §C.

Stage 1: Deterministic Eligibility Screening ($\approx 84{,}000 \rightarrow 26{,}943$). We first apply five deterministic filters without using an LLM. We validate metadata integrity by checking that image paths resolve and answer fields are non empty; remove instances associated with pre 2020 annotation artifacts; retain questions containing between 10 and 200 tokens; remove duplicates and rebalance categories across source datasets; and remap each instance to our taxonomy. We also remove any instance that cannot be assigned to one of the 80 tasks in the DEEPVISUALMATH hierarchy. This final taxonomy check acts as a quality gate and removes mathematically irrelevant or otherwise incompatible examples. Together, these filters reduce the raw pool by ≈68% at zero LLM cost.

Stage 2: Visual Redundancy and Distribution Control ($26{,}943 \rightarrow 18{,}657$). We next remove visual redundancy and control the distribution of image types. We encode every image using a CLIP ViT-L/14 encoder and index the embeddings with FAISS. We cluster near duplicates using a cosine similarity threshold of 0.97 and retain the instance with the strongest text quality signal from each cluster, reducing the pool from 26,943 to 23,468 instances. We then apply K means clustering to the image embeddings, identify over represented visual domains, and stratified downsample those domains to obtain a more balanced collection of 18,657 instances. This step shows that approximately 31% of the Stage 1 survivors contain near duplicate visual content across source datasets, redundancy that text only filtering would miss.

Stage 3: Multimodal Quality and Reasoning Audit ($18{,}657 \rightarrow 10{,}298$). Finally, we use a structured multimodal LLM evaluation to score each remaining instance on four criteria. **M1** measures mathematical domain relevance, **M2** measures the depth of reasoning required, **M3** measures visual dependence, namely whether the image contributes essential information beyond the question text, and **M4** measures language clarity. We flag 2,795 instances for M1, 1,536 for M2, 6,018 for M3, and 1,685 for M4. We retain an instance only when it satisfies all four criteria. Because a single instance may fail multiple criteria, the counts overlap: the combined reduction is 8,359 instances, approximately 45% of the Stage 2 pool, rather than the sum of the individual flagged counts, 12,034.

Benchmark Statistics and Coverage. Table 7 reports the final category distribution. This distribution is determined by source availability, filtering decisions, and the per category and per task coverage thresholds; it should therefore be interpreted as benchmark composition rather than as the prevalence of mathematical skills in a natural population. Because the overall score is instance weighted, we also report the unweighted macro average across the categories included in Table 1 (Macro). The resulting rankings are highly consistent with the weighted scores, with Kendall's $\tau = 0.933$, at most a two position shift, and the top four and bottom two models unchanged. Further details on task construction and coverage are provided in §B.

3.2 AUTOMATED EVALUATION HARNESS

To support reproducible, large scale evaluation across **10 models** and **10,298 mathematical reasoning instances**, we developed DEEPMATHEVALUATOR, a two stage automated pipeline. First, a structured LLM judge evaluates each response and produces machine readable signals for *accuracy*, *off topic content*, and *refusal*. This separates answer correctness from irrelevant or non responsive outputs and supports both multiple choice and open ended responses. Second, the harness summarizes and compares model performance using **500 resample** bootstrap confidence intervals, paired McNemar tests with Bonferroni correction, and Cohen's d. The harness saves intermediate outputs and skips rows that have already been processed, allowing evaluations to resume after interruptions and extend to new models or benchmark items without repeating completed work. Further

Table 1: **Accuracy (%)(↑) across mathematical reasoning categories, with macro aggregates.** Macro is the unweighted mean across the seven categories.

Model	Cross-Dim. (#3322)	Discrete (#1842)	Solid-3D (#1565)	Continuous (#1398)	Applied (#836)	Plane-2D (#812)	Socio-Econ. (#523)	Overall (#10298)	Macro Avg.
Proprietary Models									
Claude Sonnet 4.6	45.45	34.04	32.42	60.52	50.36	34.98	45.32	43.04	43.30
Gemini 2.5 Flash	48.42	34.75	34.79	55.99	49.82	39.98	37.86	43.84	43.09
Gemini 3.1 Flash Lite	55.52	**40.07**	**39.41**	63.99	51.50	46.16	46.73	49.95	49.05
Gemini 3.5 Flash	**60.54**	34.64	36.36	**66.60**	**56.10**	**51.23**	**48.76**	**51.36**	**50.60**
GPT-5.4 mini	57.10	37.35	39.11	65.52	55.86	44.33	45.32	50.27	49.23
Open-Source Models									
Gemma 4 31B it	38.29	23.13	25.26	48.21	48.09	29.19	27.92	34.49	34.30
GLM-4.5V	34.01	20.47	21.31	44.70	45.04	27.25	27.99	31.17	31.54
Kimi K2.5	**69.47**	**63.77**	**66.33**	**77.36**	**60.58**	**51.85**	**64.29**	**66.67**	**64.81**
Qwen3.6 Plus	20.63	12.92	15.28	26.70	31.94	18.23	16.44	19.78	20.31
Qwen3.7 Flash	14.81	11.40	13.10	22.25	25.00	13.30	11.66	15.50	15.93
Human Baselines									
High School	86.91	82.14	82.15	91.08	88.84	83.70	85.66	85.74	85.78
Undergraduate	92.87	92.35	91.37	91.42	92.22	93.97	89.29	92.20	91.93
Engineering Grad	99.39	98.62	98.49	99.33	98.76	98.48	98.09	98.92	98.74
Mathematics Grad	**99.77**	**99.33**	**99.47**	**99.71**	**99.72**	**99.51**	**99.55**	**99.60**	**99.58**

implementation details are provided in §D, and the code and reproduction instructions are available through the Code URL in the submission form.

3.3 Error Analysis Pipeline

Aggregate accuracy shows whether a model succeeds, but not why it fails. We therefore add an automated error-analysis pipeline that converts incorrect responses into fine-grained, reproducible diagnostics. The pipeline has three components: **(1) Failure Signal Preparation**, **(2) Inductive Error Pattern Discovery**, and **(3) Cross-Model Diagnostic Reporting**.

Failure Signal Preparation. We normalize the scored responses and retain correctness, off-topic, refusal, category, subcategory, and task metadata. This separates answer failures from invalid or non-responsive outputs and creates a consistent failure record for downstream analysis.

Inductive Error Pattern Discovery. We embed each incorrect instance using its question, reference answer, and model prediction. HDBSCAN then groups recurring failure patterns, while an LLM labels representative examples. The resulting clusters are mapped to a reference taxonomy covering perceptual, grounding, conceptual, computational, multi-step, language–mathematical, and format/refusal errors.

Cross-Model Diagnostic Reporting. We aggregate the error labels into per-model error profiles and task-level skill vectors. Bootstrap confidence intervals, paired McNemar tests with multiple-comparison correction, and Cohen's d quantify uncertainty and differences between models. These analyses reveal failure mechanisms that aggregate accuracy can hide, including distinct error profiles among similarly accurate models and persistent bottlenecks in visual grounding and multi-step mathematical reasoning. Further details are provided in §E, with code and reproduction instructions available through the Code URL in the submission form.

4 Experiments and Evaluation

Models. We evaluate ten contemporary multimodal language models, divided into five proprietary and five open source systems. The proprietary group includes Claude Sonnet 4.6 Anthropic (2026b), Gemini 2.5 Flash, Gemini 3.1 Flash Lite, Gemini 3.5 Flash Google Gemini Team (2025), and GPT 5.4 mini OpenAI (2026). The open source group includes Gemma 4 31B it Gemma Team (2026), GLM 4.5V Team et al. (2025), Kimi K2.5 Kimi Team (2026), Qwen3.6 Plus, and Qwen3.7 Flash Yang et al. (2025a). These models span different training sources, model families, and capability levels, allowing us to compare both overall performance and category specific strengths under a common evaluation protocol. All models receive the same images and questions, and their responses are scored using the evaluation harness described in §3.2; no task specific fine tuning is applied. We use Gemma 4 31B it as the evaluation judge, achieving Cohen's $\kappa = 0.930$ against the human majority compared with $\kappa = 0.983$ for human to human agreement (Table 14; Appendix F.2).

Human Baselines. We also report human baselines across four levels of mathematical training (§F.4): high school students in Grades 11 and 12, undergraduates in Years 2 and 3, recent engineering graduates, and mathematics graduates. Three volunteers from each level independently answer the complete benchmark, and we aggregate their responses by majority agreement to reduce the effect of

Table 2: **Distribution of the number of models that answered each question.** Each cell reports the raw number of questions followed by the percentage within the row; ranging from 0 models answered correctly to all 10 models answered correctly.

Category	#N	0	1	2	3	4	5	6	7	8	9	10
Cross Dimensional and Relational Geometry	3322	322 9.69%	485 14.60%	493 14.84%	434 13.06%	288 8.67%	255 7.68%	250 7.53%	290 8.73%	258 7.77%	207 6.23%	40 1.20%
Discrete Mathematics and Formal Logic	1842	499 27.09%	412 22.37%	261 14.17%	134 7.27%	138 7.49%	88 4.78%	72 3.91%	84 4.56%	78 4.23%	67 3.64%	9 0.49%
Solid and Three Dimensional Geometry	1565	431 27.54%	343 21.92%	208 13.29%	132 8.43%	107 6.84%	72 4.60%	59 3.77%	35 2.24%	31 1.98%	45 2.88%	102 6.52%
Continuous Mathematics and Analysis	1398	184 13.16%	131 9.37%	108 7.73%	97 6.94%	124 8.87%	135 9.66%	164 11.73%	172 12.30%	161 11.52%	118 8.44%	4 0.29%
Applied Sciences and Grounded Reasoning	836	242 28.95%	63 7.54%	44 5.26%	34 4.07%	45 5.38%	47 5.62%	37 4.43%	50 5.98%	91 10.89%	136 16.27%	47 5.62%
Plane Geometry	812	209 25.74%	150 18.47%	92 11.33%	59 7.27%	60 7.39%	41 5.05%	49 6.03%	42 5.17%	54 6.65%	55 6.77%	1 0.12%
Socio-Economic and Contextual Mathematics.	523	151 28.87%	73 13.96%	46 8.80%	42 8.03%	34 6.50%	43 8.22%	36 6.88%	40 7.65%	26 4.97%	31 5.93%	1 0.19%
Overall	10298	2038 19.79%	1657 16.09%	1252 12.16%	932 9.05%	796 7.73%	681 6.61%	667 6.48%	713 6.92%	699 6.79%	659 6.40%	204 1.98%

individual annotator bias. These baselines provide reference points for interpreting both the overall model gap and differences across mathematical categories.

Benchmark Representation and Evaluation. Let $\mathcal{B} = \{(I_i, Q_i, A_i^*, c_i, s_i, t_i)\}_{i=1}^N$ denote the complete DEEPVISUALMATH benchmark, where c_i, s_i, and t_i identify the category, subcategory, and task associated with instance i. For a model f_θ, the evaluation harness generates $\widehat{A}_i = f_\theta(I_i, Q_i)$ and applies the task specific evaluator $g_{t_i}(\widehat{A}_i, A_i^*) \in \{0, 1\}$. The evaluator supports numeric, symbolic, categorical, structured, and free form answers. Scores are computed at the task, subcategory, category, and benchmark levels.

We report two aggregate measures. Macro accuracy gives equal weight to the seven categories, while overall accuracy is the instance weighted micro accuracy:

$$\mathrm{Acc}_{\mathrm{macro}} = \frac{1}{7}\sum_{c\in\mathcal{C}} \mathrm{Acc}(c), \qquad \mathrm{Acc}_{\mathrm{overall}} \equiv \mathrm{Acc}_{\mathrm{micro}} = \frac{1}{N}\sum_{i=1}^{N} g_{t_i}(\widehat{A}_i, A_i^*).$$

Reporting both measures shows whether performance is broad or dominated by larger categories. Our error analysis further links incorrect responses to visual, abstraction, relation extraction, derivation, arithmetic, symbolic, and formatting failures, revealing which reasoning component breaks. Together, the taxonomy, evaluation protocol, and fine grained diagnostics support the empirical analysis that follows.

Agent Based Evaluation Setup. We create a balanced hard case set from instances missed by all evaluated VLMs, sampling 50 questions from each of the seven categories for a total of 350 cases. This adversarial subset measures progress on problems that current models cannot solve reliably. We evaluate three agentic systems under identical inputs and constraints: **MathSolver** (Appendix I.1), a multi agent framework with specialized reasoning roles; a **Claude Code agent** (Anthropic, 2026a); and a **Codex agent** (OpenAI, 2026), which performs sequential, program assisted reasoning. MathSolver emphasizes parallel decomposition, whereas the Claude Code and Codex agents rely on iterative tool assisted execution.

5 EXPERIMENTAL FINDINGS

5.1 OVERALL FINDINGS

Across the evaluated MLLMs in Table 1, overall accuracy ranges from **15.50%** to **66.67%**, revealing a wide capability gap. **Kimi K2.5** achieves the strongest result among both open and proprietary models, outperforming the best proprietary model, Gemini 3.5 Flash, by **15.31** points and leading in all seven categories. Its performance is strongest on Continuous Mathematics and Analysis (**77.36%**), Cross Dimensional and Relational Geometry (**69.47%**), and Solid Geometry (**66.33%**), but drops to **51.85%** on Plane Geometry. The remaining open models perform much worse, with accuracy ranging from 15.50% to 34.49%, showing that model availability alone does not predict mathematical reasoning ability.

The human baselines expose a substantial remaining gap. High school students reach **85.74%**, undergraduates reach **92.20%**, engineering graduates reach **98.92%**, and mathematics graduates reach **99.60%**; even Kimi K2.5 remains 19.07 points below the high school baseline and 32.93

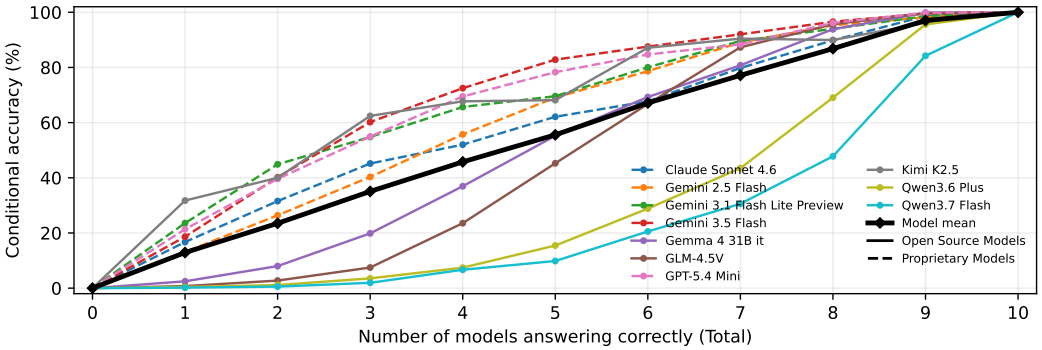


Figure 4: **Conditional accuracy by model agreement.** Open source models trail proprietary models at low and moderate agreement, while Kimi K2.5 largely closes the gap and all models converge at high agreement.

points below the mathematics graduate baseline. The difference between overall and macro accuracy also confirms that category imbalance affects aggregate results: Kimi scores 66.67% overall versus 64.81% macro, while Gemini 3.5 Flash scores 51.36% versus 50.60%. Reporting both measures therefore distinguishes broad competence from gains driven by larger categories, supporting the need for fine grained evaluation.

5.2 Difficulty and Model Capacity

✓ **Question-Level Coverage Reveals Uneven Difficulty.** Distribution in Table 2 shows that **19.79%** of all questions were answered correctly by no model, while only **1.98%** were solved by all ten, confirming that many benchmark items remain difficult for current MLLMs. The largest concentration occurs at the lower end, with a further **16.09%** of questions solved by only one model, indicating limited shared competence. Difficulty varies strongly by category: Discrete Mathematics, Applied Sciences, Solid Geometry, Plane Geometry, and Socio Economic Mathematics each have roughly one quarter of their questions unanswered by every model. In contrast, Continuous Mathematics has the smallest zero answer rate (**13.16%**) and the largest concentration of questions solved by six to nine models, suggesting stronger but still incomplete model competence. Applied Sciences and Solid Geometry show a polarized pattern, combining high zero answer rates with relatively large fractions solved by all ten models, which points to a mixture of easy and highly resistant tasks. Overall, the broad spread across coverage levels supports reporting question level difficulty and agreement in addition to aggregate accuracy.

✓ **Agreement Analysis and the Open Model Gap.** Figure 4 shows each model's accuracy conditioned on the number of models that answer a question correctly, with higher agreement generally corresponding to higher accuracy. Proprietary models consistently outperform the open source average across the low and moderate agreement range, with the largest gap when only two to six models are correct. The gap is especially clear for Qwen3.7 Flash and Qwen3.6 Plus, which remain below 50% even when eight models answer correctly, whereas most proprietary models exceed 80% in this region. Kimi K2.5 is a notable exception, closely matching or exceeding proprietary models across much of the curve and approaching near perfect accuracy at high agreement. The convergence of all models near nine or ten correct answers suggests that open source weaknesses are concentrated in ambiguous and moderately difficult questions rather than universally difficult items.

5.3 Agent-Based Evaluation

Across the 350 questions missed by all VLMs, the Multi Agent System leads overall (31.71%), outperforming Codex (17.71%) and Claude Code (15.43%); it performs best on Socio Economic Mathematics (38.00%), Cross Dimensional Geometry, and Solid Geometry (34.00% each). Claude Code leads on Continuous Mathematics (32.00%), while Codex reaches 28.00%; however, both show substantial weaknesses on specific geometry and contextual categories. These results indicate that agentic reasoning can recover VLM failures, but its effectiveness depends on task structure, supporting hybrid systems that combine decomposition, visual verification, and deep reasoning.

5.4 Error Analysis

To move beyond aggregate accuracy, we conduct an item level analysis of model outputs and judge rationales. We assign each incorrect response to the earliest point at which its solution loses ground-

Table 3: Agent-based performance on hard-question evaluation (50 hard tasks per category).

Category	# Questions	Multi-Agent System	Claude Code Agent	Codex Agent
Applied Sciences & Grounded Reasoning	50	32.00%	12.00%	16.00%
Continuous Mathematics & Analysis	50	30.00%	32.00%	28.00%
Cross-Dimensional & Relational Geometry	50	34.00%	14.00%	24.00%
Discrete Mathematics & Formal Logic	50	26.00%	24.00%	24.00%
Plane (2D) Geometry	50	28.00%	8.00%	12.00%
Socio-Economic & Contextual Mathematics	50	38.00%	4.00%	14.00%
Solid (3D) Geometry	50	34.00%	14.00%	6.00%
Overall	**350**	**31.71%**	**15.43%**	**17.71%**

ing in the image or question, following a first break principle. The resulting taxonomy contains seven categories: perceptual failure (E1), representation grounding failure (E2), conceptual or procedural failure (E3), quantitative computation failure (E4), multi step reasoning failure (E5), language to mathematics mapping failure (E6), and format or refusal failure (E7). This organization separates visual mistakes from mathematical selection, execution, reasoning chain, language interpretation, and answer delivery. It also connects failures to targeted interventions, such as visual verification for E1 and E2, constraint checking for E3 and E5, computational verification for E4, and constrained decoding for E7. Because the stored error tags are incomplete and occasionally noisy, we use correctness scores for quantitative comparisons and judge rationales for qualitative diagnosis; full audit details are provided in Appendix J.

The dominant failure is loss of visual or task grounding before the final calculation: many incorrect responses contain fluent algebra but operate on an incorrect chart value, geometric relation, answer option, or question polarity. Stronger models reduce gross perceptual and output failures, yet remain vulnerable to ambiguous visual configurations and long chains of dependent operations. Weaker models additionally show truncation, refusal, malformed responses, and failure to follow the requested answer format. Across models, discrete and solid geometry remain persistent stress points, whereas continuous and cross dimensional tasks are relatively stronger for the best systems. The response matrix also reveals complementary capabilities, so models with similar aggregate accuracy can fail on different questions and for different reasons. These findings motivate capability specific evaluation and targeted improvements, while future releases should assign complete first break labels to every incorrect item and validate them against human annotations.

6 Discussion

The evaluation shows that visual mathematical reasoning is not a single capability, but an interaction among visual grounding, mathematical interpretation, symbolic manipulation, and computation. The variation across categories indicates that models can possess strong mathematical competence while remaining unreliable when success depends on extracting and relating information from visual representations. This suggests that aggregate performance can obscure qualitatively different reasoning profiles, where similar overall scores may arise from very different combinations of strengths and failure modes. The fine-grained evaluation therefore provides a more informative view of model behavior by locating failures at the level of specific reasoning components rather than treating every incorrect answer as the same type of error. The hard-case evaluation further suggests that some failures are not necessarily due to missing mathematical knowledge, but may arise from how models decompose a problem, maintain intermediate structure, or connect visual evidence to executable reasoning. The gains from tool-assisted and multi-step agents consequently point to a distinction between the capabilities available to a model and the capabilities it can reliably realize through a single-pass response. More broadly, these findings motivate evaluation settings that test not only whether a model reaches the correct answer, but whether it can consistently transform visual evidence into the structured reasoning required to obtain that answer.

7 Conclusion

We introduced DeepVisualMath, a unified taxonomy and evaluation suite for holistic visual mathematical reasoning in multimodal language models. Its hierarchical taxonomy, automated curation pipeline, reproducible evaluation harness, and structured error analysis expose capability differences that narrow benchmarks and aggregate scores can conceal. Our experiments show that current MLLMs remain substantially below human performance and are highly uneven across mathematical domains, while agentic decomposition can recover some otherwise unsolved cases. We hope DeepVisualMath supports more precise measurement, diagnosis, and development of multimodal systems that reason reliably across diverse mathematical representations.

AI USE STATEMENT

The authors used AI tools for writing assistance, grammar checking, sentence editing, and polishing. LLMs were also used in the data curation pipeline to assist with filtering and quality assessment of existing instances; no synthetic benchmark instances were generated. An LLM judge was used to evaluate model responses under the protocol described in §3.2. AI tools also supported code debugging, finding relevant literature, and figure ideation. The authors reviewed all AI outputs, verified the resulting analyses, and take full responsibility for the content and claims of this paper.

REPRODUCIBILITY STATEMENT

We provide the taxonomy, benchmark construction procedure, evaluation protocol, and implementation details needed to reproduce our results. The complete data curation process, including source aggregation, filtering, deduplication, balancing, and quality checks, is described in Appendix C, while the task hierarchy and category definitions are given in §B. The evaluation harness, answer judging procedure, aggregation rules, statistical comparisons, and error analysis protocol are documented in §3.2 and Appendix J. We also describe the human evaluation protocol in §F.4 and the agent configurations in §I, including the hard case construction and model settings.

REFERENCES

Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Dhruv Batra, and Devi Parikh. Vqa: Visual question answering. arXiv, 2015. doi: 10.48550/ARXIV.1505.00468. URL https://arxiv.org/abs/1505.00468.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning. arXiv, 2022. doi: 10.48550/ARXIV.2204.14198. URL https://arxiv.org/abs/2204.14198.

Shaden Alshammari, Kevin Wen, Abrar Zainal, Mark Hamilton, Navid Safaei, Sultan Albarakati, William T. Freeman, and Antonio Torralba. Mathnet: a global multimodal benchmark for mathematical reasoning and retrieval, 2026. URL https://arxiv.org/abs/2604.18584.

Anthropic. Claude code, 2026a. URL https://www.anthropic.com/product/claude-code. Accessed: 2026-05-05.

Anthropic. Introducing claude sonnet 4.6, February 2026b. URL https://www.anthropic.com/news/claude-sonnet-4-6. Accessed: 2026-09-19.

Huanqia Cai, Yijun Yang, and Winston Hu. Mm-iq: Benchmarking human-like abstraction and reasoning in multimodal models, 2025. URL https://arxiv.org/abs/2502.00698.

Jie Cai. Comparebench: A benchmark for visual comparison reasoning in vision-language models, 2025. URL https://arxiv.org/abs/2509.22737.

Jiacheng Chen, Tianhao Liang, Sherman Siu, Zhengqing Wang, Kai Wang, Yubo Wang, Yuansheng Ni, Wang Zhu, Ziyan Jiang, Bohan Lyu, Dongfu Jiang, Xuan He, Yuan Liu, Hexiang Hu, Xiang Yue, and Wenhu Chen. Mega-bench: Scaling multimodal evaluation to over 500 real-world tasks, 2024. URL https://arxiv.org/abs/2410.10563.

Jiaqi Chen, Jianheng Tang, Jinghui Qin, Xiaodan Liang, Lingbo Liu, Eric P. Xing, and Liang Lin. Geoqa: A geometric question answering benchmark towards multimodal numerical reasoning. arXiv, 2021. doi: 10.48550/ARXIV.2105.14517. URL https://arxiv.org/abs/2105.14517.

Jiaqi Chen, Tong Li, Jinghui Qin, Pan Lu, Liang Lin, Chongyu Chen, and Xiaodan Liang. Unigeo: Unifying geometry logical reasoning via reformulating mathematical expression. arXiv, 2022. doi: 10.48550/ARXIV.2212.02746. URL https://arxiv.org/abs/2212.02746.

Wenhu Chen, Ming Yin, Max Ku, Pan Lu, Yixin Wan, Xueguang Ma, Jianyu Xu, Xinyi Wang, and Tony Xia. Theoremqa: A theorem-driven question answering dataset. arXiv, 2023. doi: 10.48550/ARXIV.2305.12524. URL https://arxiv.org/abs/2305.12524.

Konstantin Chernyshev, Vitaliy Polshkov, Ekaterina Artemova, Alex Myasnikov, Vlad Stepanov, Alexei Miasnikov, and Sergei Tilga. U-math: A university-level benchmark for evaluating mathematical skills in llms, 2024. URL https://arxiv.org/abs/2412.03205.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. arXiv, 2021. doi: 10.48550/ARXIV.2110.14168. URL https://arxiv.org/abs/2110.14168.

Mario Cotilla, Diego Cordoba, and Diana Nunez. Morphotectonic study of the greater antilles. 2024. doi: 10.48550/ARXIV.2402.14826. URL https://arxiv.org/abs/2402.14826.

Rocktim Jyoti Das, Simeon Emilov Hristov, Haonan Li, Dimitar Iliyanov Dimitrov, Ivan Koychev, and Preslav Nakov. Exams-v: A multi-discipline multilingual multimodal exam benchmark for evaluating vision language models, 2024. URL https://arxiv.org/abs/2403.10378.

Jun Feng, Zixin Wang, Zhentao Zhang, Yue Guo, Zhihan Zhou, Xiuyi Chen, Zhenyang Li, and Dawei Yin. Mathreal: We keep it real! a real scene benchmark for evaluating math reasoning in multimodal large language models, 2025. URL https://arxiv.org/abs/2508.06009.

Deqing Fu, Ruohao Guo, Ghazal Khalighinejad, Ollie Liu, Bhuwan Dhingra, Dani Yogatama, Robin Jia, and Willie Neiswanger. Isobench: Benchmarking multimodal foundation models on isomorphic representations, 2024. URL https://arxiv.org/abs/2404.01266.

Junyuan Gao, Jiahe Song, Jiang Wu, Runchuan Zhu, Guanlin Shen, Shasha Wang, Xingjian Wei, Haote Yang, Weijia Li, Bin Wang, Lijun Wu, and Conghui He. Benchmarking and boosting multilingual capabilities of lvlms via ocr-centric reinforcement learning, 2025. URL https://arxiv.org/abs/2503.18484.

Gemini Team. Gemini: A family of highly capable multimodal models. 2023. doi: 10.48550/ARXIV.2312.11805. URL https://arxiv.org/abs/2312.11805.

Gemma Team. Gemma 4 technical report, 2026. URL https://arxiv.org/abs/2607.02770.

Google Gemini Team. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. July 2025.

Meng-Hao Guo, Xuanyu Chu, Qianrui Yang, Zhe-Han Mo, Yiqing Shen, Pei-lin Li, Xinjie Lin, Jinnian Zhang, Xin-Sheng Chen, Yi Zhang, Kiyohiro Nakayama, Zhengyang Geng, Houwen Peng, Han Hu, and Shi-Min Hu. Rbench-v: A primary assessment for visual reasoning models with multi-modal outputs, 2025. URL https://arxiv.org/abs/2505.16770.

Patrick Haller, Fabio Barth, Jonas Golde, Georg Rehm, and Alan Akbik. Pisa-bench: The pisa index as a multilingual and multimodal metric for the evaluation of vision-language models, 2025. URL https://arxiv.org/abs/2510.24792.

Yunzhuo Hao, Jiawei Gu, Huichen Will Wang, Linjie Li, Zhengyuan Yang, Lijuan Wang, and Yu Cheng. Can mllms reason in multimodality? emma: An enhanced multimodal reasoning benchmark, 2025. URL https://arxiv.org/abs/2501.05444.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems, 2024. URL https://arxiv.org/abs/2402.14008.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. arXiv, 2021. doi: 10.48550/ARXIV.2103.03874. URL https://arxiv.org/abs/2103.03874.

Md Tanzib Hosain, Salman Rahman, Md Kishor Morol, and Md Rizwan Parvez. Xolver: Multi-agent reasoning with holistic experience learning just like an olympiad team, 2025. URL https://arxiv.org/abs/2506.14234.

Kung-Hsiang Huang, Can Qin, Haoyi Qiu, Philippe Laban, Shafiq Joty, Caiming Xiong, and Chien-Sheng Wu. Why vision language models struggle with visual arithmetic? towards enhanced chart and geometry understanding, 2025. URL https://arxiv.org/abs/2502.11492.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. arXiv, 2016. doi: 10.48550/ARXIV.1612.06890. URL https://arxiv.org/abs/1612.06890.

Kushal Kafle, Brian Price, Scott Cohen, and Christopher Kanan. Dvqa: Understanding data visualizations via question answering. arXiv, 2018. doi: 10.48550/ARXIV.1801.08163. URL https://arxiv.org/abs/1801.08163.

Mehran Kazemi, Hamidreza Alvari, Ankit Anand, Jialin Wu, Xi Chen, and Radu Soricut. Geomverse: A systematic evaluation of large models for geometric reasoning, 2023. URL https://arxiv.org/abs/2312.12241.

Kimi Team. Kimi k2.5: Visual agentic intelligence, 2026. URL https://arxiv.org/abs/2602.02276.

Chen Li, Han Zhang, Zhantao Yang, Fangyi Chen, Zihan Wang, Anudeepsekhar Bolimera, and Marios Savvides. Stelar-vision: Self-topology-aware efficient learning for aligned reasoning in vision, 2025a. URL https://arxiv.org/abs/2508.08688.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. arXiv, 2023. doi: 10.48550/ARXIV.2301.12597. URL https://arxiv.org/abs/2301.12597.

Yifan Li, Zhenghao Chen, Ziheng Wu, Kun Zhou, Ruipu Luo, Can Zhang, Zhentao He, Yufei Zhan, Wayne Xin Zhao, and Minghui Qiu. Unleashing perception-time scaling to multimodal reasoning models, 2025b. URL https://arxiv.org/abs/2510.08964.

Hao Liang, Linzhuang Sun, Minxuan Zhou, Zirong Chen, Meiyi Qiang, Mingan Lin, Tianpeng Li, Fan Yang, Zenan Zhou, and Wentao Zhang. Mathscape: Benchmarking multimodal large language models in real-world mathematical contexts, 2024. URL https://arxiv.org/abs/2408.07543.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. arXiv, 2023. doi: 10.48550/ARXIV.2304.08485. URL https://arxiv.org/abs/2304.08485.

Pan Lu, Ran Gong, Shibiao Jiang, Liang Qiu, Siyuan Huang, Xiaodan Liang, and Song-Chun Zhu. Inter-gps: Interpretable geometry problem solving with formal language and symbolic reasoning. arXiv, 2021a. doi: 10.48550/ARXIV.2105.04165. URL https://arxiv.org/abs/2105.04165.

Pan Lu, Ran Gong, Shibiao Jiang, Liang Qiu, Siyuan Huang, Xiaodan Liang, and Song-Chun Zhu. Inter-gps: Interpretable geometry problem solving with formal language and symbolic reasoning, 2021b. URL https://arxiv.org/abs/2105.04165.

Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. arXiv, 2022. doi: 10.48550/ARXIV.2209.09513. URL https://arxiv.org/abs/2209.09513.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts, 2023a. URL https://arxiv.org/abs/2310.02255.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. 2023b. doi: 10.48550/ARXIV.2310.02255. URL https://arxiv.org/abs/2310.02255.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. arXiv, 2022. doi: 10.48550/ARXIV.2203.10244. URL https://arxiv.org/abs/2203.10244.

Nitesh Methani, Pritha Ganguly, Mitesh M. Khapra, and Pratyush Kumar. Plotqa: Reasoning over scientific plots. arXiv, 2019. doi: 10.48550/ARXIV.1909.00997. URL https://arxiv.org/abs/1909.00997.

OpenAI. Openai gpt-5 system card, 2026. URL https://arxiv.org/abs/2601.03267.

OpenAI. Codex: The same powerful coding agent—now in chatgpt, 2026. URL https://openai.com/codex/. Accessed: 2026-09-19.

Nilay Pande, Sahiti Yerramilli, Jayant Sravan Tamarapalli, and Rynaa Grover. Marvl-qa: A benchmark for mathematical reasoning over visual landscapes, 2025. URL https://arxiv.org/abs/2508.17180.

Runqi Qiao, Qiuna Tan, Peiqing Yang, Yanzi Wang, Xiaowan Wang, Enhui Wan, Sitong Zhou, Guanting Dong, Yuchen Zeng, Yida Xu, Jie Wang, Chong Sun, Chen Li, and Honggang Zhang. We-math 2.0: A versatile mathbook system for incentivizing visual mathematical reasoning, 2025a. URL https://arxiv.org/abs/2508.10433.

Runqi Qiao, Qiuna Tan, Peiqing Yang, Yanzi Wang, Xiaowan Wang, Enhui Wan, Sitong Zhou, Guanting Dong, Yuchen Zeng, Yida Xu, Jie Wang, Chong Sun, Chen Li, and Honggang Zhang. We-math 2.0: A versatile mathbook system for incentivizing visual mathematical reasoning, 2025b. URL https://arxiv.org/abs/2508.10433.

Jonathan Roberts, Kai Han, Neil Houlsby, and Samuel Albanie. Scifibench: Benchmarking large multimodal models for scientific figure interpretation, 2024. URL https://arxiv.org/abs/2405.08807.

Jonathan Roberts, Mohammad Reza Taesiri, Ansh Sharma, Akash Gupta, Samuel Roberts, Ioana Croitoru, Simion-Vlad Bogolin, Jialu Tang, Florian Langer, Vyas Raina, Vatsal Raina, Hanyi Xiong, Vishaal Udandarao, Jingyi Lu, Shiyang Chen, Sam Purkis, Tianshuo Yan, Wenye Lin, Gyungin Shin, Qiaochu Yang, Anh Totti Nguyen, David I. Atkinson, Aaditya Baranwal, Alexandru Coca, Mikah Dang, Sebastian Dziadzio, Jakob D. Kunz, Kaiqu Liang, Alexander Lo, Brian Pulfer, Steven Walton, Charig Yang, Kai Han, and Samuel Albanie. Zerobench: An impossible visual benchmark for contemporary large multimodal models, 2025. URL https://arxiv.org/abs/2502.09696.

5 Team, Aohan Zeng, Xin Lv, Qinkai Zheng, Zhenyu Hou, Bin Chen, Chengxing Xie, Cunxiang Wang, Da Yin, Hao Zeng, Jiajie Zhang, Kedong Wang, Lucen Zhong, Mingdao Liu, Rui Lu, Shulin Cao, Xiaohan Zhang, Xuancheng Huang, Yao Wei, Yean Cheng, Yifan An, Yilin Niu, Yuanhao Wen, Yushi Bai, Zhengxiao Du, Zihan Wang, Zilin Zhu, Bohan Zhang, Bosi Wen, Bowen Wu, Bowen Xu, Can Huang, Casey Zhao, Changpeng Cai, Chao Yu, Chen Li, Chendi Ge, Chenghua Huang, Chenhui Zhang, Chenxi Xu, Chenzheng Zhu, Chuang Li, Congfeng Yin, Daoyan Lin, Dayong Yang, Dazhi Jiang, Ding Ai, Erle Zhu, Fei Wang, Gengzheng Pan, Guo Wang, Hailong Sun, Haitao Li, Haiyang Li, Haiyi Hu, Hanyu Zhang, Hao Peng, Hao Tai, Haoke Zhang, Haoran Wang, Haoyu Yang, He Liu, He Zhao, Hongwei Liu, Hongxi Yan, Huan Liu, Huilong Chen, Ji Li, Jiajing Zhao, Jiamin Ren, Jian Jiao, Jiani Zhao, Jianyang Yan, Jiaqi Wang, Jiayi Gui, Jiayue Zhao, Jie Liu, Jijie Li, Jing Li, Jing Lu, Jingsen Wang, Jingwei Yuan, Jingxuan Li, Jingzhao Du, Jinhua Du, Jinxin Liu, Junkai Zhi, Junli Gao, Ke Wang, Lekang Yang, Liang Xu, Lin Fan, Lindong Wu, Lintao Ding, Lu Wang, Man Zhang, Minghao Li, Minghuan Xu, Mingming Zhao, Mingshu Zhai, Pengfan Du, Qian Dong, Shangde Lei, Shangqing Tu, Shangtong Yang, Shaoyou Lu, Shijie Li, Shuang Li, Shuang-Li, Shuxun Yang, Sibo Yi, Tianshu Yu, Wei Tian, Weihan Wang, Wenbo Yu, Weng Lam Tam, Wenjie Liang, Wentao Liu, Xiao Wang, Xiaohan Jia, Xiaotao Gu, Xiaoying Ling, Xin Wang, Xing Fan, Xingru Pan, Xinyuan Zhang, Xinze Zhang,

Xiuqing Fu, Xunkai Zhang, Yabo Xu, Yandong Wu, Yida Lu, Yidong Wang, Yilin Zhou, Yiming Pan, Ying Zhang, Yingli Wang, Yingru Li, Yinpei Su, Yipeng Geng, Yitong Zhu, Yongkun Yang, Yuhang Li, Yuhao Wu, Yujiang Li, Yunan Liu, Yunqing Wang, Yuntao Li, Yuxuan Zhang, Zezhen Liu, Zhen Yang, Zhengda Zhou, Zhongpei Qiao, Zhuoer Feng, Zhuorui Liu, Zichen Zhang, Zihan Wang, Zijun Yao, Zikang Wang, Ziqiang Liu, Ziwei Chai, Zixuan Li, Zuodong Zhao, Wenguang Chen, Jidong Zhai, Bin Xu, Minlie Huang, Hongning Wang, Juanzi Li, Yuxiao Dong, and Jie Tang. Glm-4.5: Agentic, reasoning, and coding (arc) foundation models, 2025. URL https://arxiv.org/abs/2508.06471.

Aleix Torres-Camps, Nathaniel Mitrani Hadida, Víctor Conchello Vendrell, Àlex Batlle Casellas, Arnau Padrés Masdemont, and Jordi Ros-Giralt. M3kang: Evaluating multilingual multimodal mathematical reasoning in vision-language models, 2026. URL https://arxiv.org/abs/2601.16218.

Junling Wang, Anna Rutkiewicz, April Yi Wang, and Mrinmaya Sachan. Generating pedagogically meaningful visuals for math word problems: A new benchmark and analysis of text-to-image models, 2025a. URL https://arxiv.org/abs/2506.03735.

Peijie Wang, Zhong-Zhi Li, Fei Yin, Xin Yang, Dekang Ran, and Cheng-Lin Liu. Mv-math: Evaluating multimodal math reasoning in multi-visual contexts, 2025b. URL https://arxiv.org/abs/2502.20808.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. 2024. doi: 10.48550/ARXIV.2409.12191. URL https://arxiv.org/abs/2409.12191.

Siting Wang, Minnan Pei, Luoyang Sun, Cheng Deng, Yuchen Li, Kun Shao, Zheng Tian, Haifeng Zhang, and Jun Wang. Spatialviz-bench: A cognitively-grounded benchmark for diagnosing spatial visualization in mllms, 2025c. URL https://arxiv.org/abs/2507.07610.

Changti Wu, Shijie Lian, Zihao Liu, Lei Zhang, Laurence Tianruo Yang, and Kai Chen. Dynasolidgeo: A dynamic benchmark for genuine spatial mathematical reasoning of vlms in solid geometry, 2025. URL https://arxiv.org/abs/2510.22340.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025a. URL https://arxiv.org/abs/2505.09388.

Tengchao Yang, Sichen Guo, Mengzhao Jia, Jiaming Su, Yuanyang Liu, Zhihan Zhang, and Meng Jiang. Mmtutorbench: The first multimodal benchmark for ai math tutoring, 2025b. URL https://arxiv.org/abs/2510.23477.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi, 2023a. URL https://arxiv.org/abs/2311.16502.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. arXiv, 2023b. doi: 10.48550/ARXIV.2311.16502. URL https://arxiv.org/abs/2311.16502.

Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Peng Gao, and Hongsheng Li. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems?, 2024. URL https://arxiv.org/abs/2403.14624.

Xintong Zhang, Xiaowen Zhang, Jingrong Wu, Zhi Gao, Shilin Yan, Zhenxin Diao, Kunpeng Gao, Xuanyan Chen, Yuwei Wu, Yunde Jia, and Qing Li. Adaptmmbench: Benchmarking adaptive multimodal reasoning for mode selection and reasoning process, 2026. URL https://arxiv.org/abs/2602.02676.

Pengfei Zhou, Fanrui Zhang, Xiaopeng Peng, Zhaopan Xu, Jiaxin Ai, Yansheng Qiu, Chuanhao Li, Zhen Li, Ming Li, Yukang Feng, Jianwen Sun, Haoquan Zhang, Zizhen Li, Xiaofeng Mao, Wangbo Zhao, Kai Wang, Xiaojun Chang, Wenqi Shao, Yang You, and Kaipeng Zhang. Mdk12-bench: A multi-discipline benchmark for evaluating reasoning in multimodal large language models, 2025. URL https://arxiv.org/abs/2504.05782.

Chengke Zou, Xingang Guo, Rui Yang, Junyu Zhang, Bin Hu, and Huan Zhang. Dynamath: A dynamic visual benchmark for evaluating mathematical reasoning robustness of vision language models, 2024. URL https://arxiv.org/abs/2411.00836.

Appendix

Table of Content

A Related Work **19**

A.1 Text Based and Formal Mathematical Reasoning 19

A.2 Geometry and Diagrammatic Mathematical Reasoning 19

A.3 Charts, Tables, Scientific Figures, and Grounded Quantitative Reasoning 19

A.4 General Multimodal Mathematical Benchmarks 20

A.5 Robustness, Compositionality, and Genuine Visual Use 20

A.6 Applied, Educational, Real World, and Multilingual Mathematical Reasoning 21

A.7 Positioning DEEPVISUALMATH 21

B Task Taxonomy and Descriptions **24**

B.1 Discrete Mathematics and Formal Logic 24

B.1.1 Abstract Logic and Procedural Puzzles 24

B.1.2 Arithmetic and Concrete Number Theory 25

B.1.3 Combinatorics and Graph Structures 25

B.2 Solid Three Dimensional Geometry 26

B.2.1 Solid Measurement 26

B.2.2 Solid Visualization and Projection 26

B.3 Continuous Mathematics and Analysis 27

B.3.1 Calculus and Advanced Analysis 27

B.3.2 Foundations of Algebra 27

B.3.3 Functional Analysis and Precalculus 28

B.3.4 Probability, Statistics and Analytics 28

B.4 Socio Economic and Contextual Mathematics 28

B.4.1 Operational and Financial Analytics 29

B.4.2 Semantic and Grounded Representation 29

B.5 Cross Dimensional and Relational Geometry 29

B.5.1 Analytic and Transformation Geometry 29

B.5.2 Comparative Relations 30

B.5.3 Foundational and Topological Geometry 30

B.5.4 Metric Measurement 30

B.6 Applied Sciences and Grounded Reasoning 30

B.6.1 Physical Systems Reasoning 30

B.7 Plane Two Dimensional Geometry 31

B.7.1 Plane Coordinate and Transformation 31

B.7.2 Plane Measurement 31

B.8 Taxonomy Summary 32

C Automated Dataset Curation Pipeline **32**

C.1 Stage 1: Deterministic Eligibility Screening (84K $\rightarrow$ 26,943) 33

C.2 Stage 2: Visual Redundancy and Balance Control (26,943 $\rightarrow$ 18,657) 33

C.3 Stage 3: Multimodal Quality and Reasoning Audit (18,657 $\rightarrow$ 10,298) 34

C.4 Generalizability of the Pipeline 34

D Automated Evaluation Pipeline **34**

D.1 Pipeline Overview 35

D.2 Component 1: Response Scoring and Validity Checks 35

D.3 Component 2: Statistical Comparison and Reporting 36

D.4 Output Schema 36

D.5 Limitations and Design Trade-offs 37

E Automated Error Analysis Pipeline **37**

E.1 Stage 1: Response-Level Failure Characterization 37

E.2 Stage 2: Inductive Error Pattern Discovery 38

E.2.1 Step 2a: Failure Representation 38

E.2.2 Step 2b: Density Based Clustering 38

E.2.3 Step 2c: Cluster Interpretation 38

E.2.4 Step 2d: Reference Error Taxonomy 38

E.3 Stage 3: Cross-Model Diagnostic Reporting 39

F More Details on Methodology **40**

F.1 Evaluation Stability: Statistical Comparison and Reporting 40

F.2 Establishing AI Judge Performance 41

F.3 Reproducibility and Infrastructure Considerations 42

F.4 Human Evaluation 43

G Model Performance Breakdown on DEEPVISUALMATH **43**

G.1 Model Complementarity 43

G.2 Model Performance On Off topic Rate 43

G.3 Model Performance On Refusal Rate 45

H Category-wise Model Performance of DEEPVISUALMATH **46**

H.1 Closed-source Models 46

H.2 Open-source Models 46

I Agent-Based Evaluation **50**

I.1 MATHSOLVER: Multi-Agent system for Visual Mathematical Reasoning 50

I.1.1 Agents 50
I.2 Claude Code Setup 52
I.3 Codex Setup 52

J Error Analysis **52**
J.1 Taxonomy 53
J.2 E1: Perceptual failure 53
J.3 E2: Representation grounding failure 53
J.4 E3: Conceptual or procedural failure 54
J.5 E4: Quantitative computation failure 54
J.6 E5: Multi-step reasoning failure 55
J.7 E6: Language-to-mathematics mapping failure 55
J.8 E7: Format or refusal failure 55

K Prompts used in DEEPVISUALMATH **56**
K.1 Prompts for MATHSOLVER 58

L Representative Benchmark Samples Across All Tasks **69**

A Related Work

A.1 Text Based and Formal Mathematical Reasoning

Research on mathematical reasoning initially focused on text only problems that isolate arithmetic, algebraic manipulation, symbolic derivation, and multi step deduction. The MATH benchmark evaluates competition level problems across several mathematical subjects and remains an important reference for measuring formal problem solving in language models (Hendrycks et al., 2021). GSM8K instead emphasizes grade school word problems, requiring models to translate natural language descriptions into arithmetic operations and intermediate calculations (Cobbe et al., 2021). U-MATH extends this direction to university level mathematics and includes problems drawn from real academic curricula (Chernyshev et al., 2024). OlympiadBench targets substantially more difficult bilingual scientific and mathematical problems, including multimodal questions and olympiad level reasoning (He et al., 2024). These datasets provide useful measures of mathematical knowledge and symbolic reasoning, but their inputs generally do not require a model to recover mathematical structure from diagrams, plots, tables, or physical scenes. Their evaluation also tends to emphasize final answer correctness, which makes it difficult to distinguish failures in mathematical knowledge from failures in visual grounding or representation. Multimodal tutoring benchmarks such as MMTutorBench broaden the problem toward explanation, feedback, and educational interaction, but remain centered on tutoring scenarios rather than a comprehensive taxonomy of mathematical capabilities (Yang et al., 2025b). DeepVisualMath builds on this foundation by retaining the need for formal mathematical reasoning while introducing broad visual coverage across continuous, discrete, geometric, applied, contextual, and quantitative problem settings.

A.2 Geometry and Diagrammatic Mathematical Reasoning

Geometry has been one of the most extensively studied areas of visual mathematical reasoning. Geometry3K introduced a large collection of diagram based geometry problems and established a standard setting for jointly interpreting natural language, geometric figures, and mathematical relations (Lu et al., 2021a). InterGPS further connected geometric perception with formal language generation and symbolic reasoning, demonstrating the value of structured intermediate representations for solving diagrammatic problems (Lu et al., 2021b). GeoQA and UniGeo expanded the problem by linking diagram understanding to numerical calculation, geometric recognition, and unified solution generation (Chen et al., 2021; 2022). GeomVerse provided a systematic evaluation of geometric reasoning in large models, with emphasis on the diversity of geometric configurations and reasoning patterns (Kazemi et al., 2023). MathVerse examined whether multimodal models genuinely use diagrams by constructing multiple versions of the same problem with different proportions of textual and visual information (Zhang et al., 2024). More recent work studies specialized settings such as solid geometry, spatial visualization, and dynamic geometric reasoning (Wu et al., 2025; Wang et al., 2025c; Qiao et al., 2025b). These benchmarks are valuable because they expose failures in diagram parsing, coordinate reasoning, projection, and theorem application, but they remain concentrated on geometric subsets of mathematical intelligence. As shown in Table 4, most geometry benchmarks cover only a small number of the foundational, metric, plane, or solid geometry dimensions and provide limited coverage of calculus, probability, applied science, or socio economic mathematics. DeepVisualMath incorporates geometric reasoning as a central component while explicitly separating plane measurement, solid measurement, solid visualization and projection, and cross dimensional relational reasoning within a broader mathematical hierarchy.

A.3 Charts, Tables, Scientific Figures, and Grounded Quantitative Reasoning

A separate line of work evaluates mathematical reasoning over charts, tables, plots, and scientific figures. DVQA introduced visual question answering over synthetic data visualizations, focusing on retrieving values and answering questions about chart structure (Kafle et al., 2018). PlotQA extended this setting to scientific plots and required models to perform numerical and relational reasoning over plotted data (Methani et al., 2019). ChartQA added natural and synthetic charts with questions requiring visual and logical reasoning, including comparisons, trends, and numerical operations (Masry et al., 2022). SciFIBench moved toward scientific figure interpretation and evaluated whether multimodal models can recover information from figures used in scientific communication (Roberts et al., 2024). ScienceQA and MMMU further incorporate scientific diagrams, tables,

and discipline specific visual materials into broader multimodal question answering (Lu et al., 2022; Yue et al., 2023a). Recent benchmarks such as CompareBench and CogAlign examine visual comparison, chart interpretation, and the reasoning failures that arise when models must align visual relations with numerical operations (Cai, 2025; Huang et al., 2025). These datasets make visual quantitative reasoning measurable, but many of them are organized around a particular representation type rather than around the mathematical capability required to interpret it. A single chart may test arithmetic, probability, statistical comparison, semantic grounding, or operational reasoning, yet representation based taxonomies often place all such problems under one broad chart category. DEEPVISUALMATH instead categorizes instances according to their underlying mathematical demands and includes dedicated coverage for metric measurement, probability and statistics, operational and financial analytics, semantic and grounded representation, and physical systems reasoning.

A.4 GENERAL MULTIMODAL MATHEMATICAL BENCHMARKS

Recent multimodal mathematics benchmarks attempt to combine multiple visual formats and mathematical subjects in a single evaluation. MathVista aggregates diverse visual mathematics problems and evaluates arithmetic, algebraic, geometric, logical, scientific, and statistical reasoning in visual contexts (Lu et al., 2023a). MathVerse emphasizes diagram dependence and constructs multiple information conditions to test whether models actually use the image rather than relying on textual shortcuts (Zhang et al., 2024). MATH-Vision collects competition problems with visual contexts across a broad range of mathematical disciplines and introduces difficulty based analysis of multimodal mathematical reasoning (Cotilla et al., 2024). WE-MATH focuses on hierarchical mathematical concepts and distinguishes insufficient knowledge, inadequate generalization, complete mastery, and rote memorization (Qiao et al., 2025a). WE-MATH 2.0 further develops a large math book style evaluation setting intended to support knowledge organized visual reasoning (Qiao et al., 2025b). Broad multimodal evaluations such as MMMU, MEGA-Bench, EMMA, MDK12-Bench, and MathNet place mathematical questions alongside problems from science, education, and other disciplines (Yue et al., 2023a; Chen et al., 2024; Hao et al., 2025; Zhou et al., 2025; Alshammari et al., 2026). These works substantially improve on isolated geometry or chart benchmarks, but their taxonomies remain heterogeneous and their coverage is uneven across mathematical subfields. In particular, calculus and advanced analysis, combinatorics and graph structures, operational mathematics, physical systems, socio economic contexts, and solid visualization are rarely evaluated together under a shared framework. Their aggregate scores therefore support broad model comparison but provide limited evidence about which mathematical capabilities are responsible for success or failure. DEEPVISUALMATH complements these efforts with a fixed four level hierarchy of 7 categories, 18 subcategories, and 80 tasks, allowing results from heterogeneous visual problems to be compared on a common mathematical axis.

A.5 ROBUSTNESS, COMPOSITIONALITY, AND GENUINE VISUAL USE

Several recent studies move beyond static accuracy and investigate whether multimodal models are robust, compositional, and genuinely dependent on visual evidence. DynaMath introduces dynamic visual variants to test the stability of mathematical reasoning under controlled changes to visual inputs (Zou et al., 2024). IsoBench studies isomorphic representations and asks whether models preserve their answers when the same underlying structure is expressed through different visual forms (Fu et al., 2024). Math Blind shows that failures in diagram understanding can undermine apparently successful mathematical reasoning, highlighting the importance of separating perception from downstream derivation (Gao et al., 2025). ZeroBench and RBench deliberately construct difficult visual reasoning settings to expose weaknesses that are hidden by conventional benchmark distributions (Roberts et al., 2025; Guo et al., 2025). MM-IQ studies abstraction and reasoning beyond direct visual recognition, while MV-MATH evaluates problems requiring multiple visual inputs rather than a single image (Cai et al., 2025; Wang et al., 2025b). Other work investigates perception time scaling, topology aware visual reasoning, and adaptive selection of reasoning modes (Li et al., 2025b;a; Zhang et al., 2026). These benchmarks provide important stress tests, but they are usually designed around one failure mechanism, such as visual dependence, invariance, multi image integration, or adaptive inference. Consequently, they do not by themselves provide broad estimates of mathematical capability across all major domains. DEEPVISUALMATH is complementary to these robustness studies: it supplies the large capability taxonomy needed to localize failures, while

the robustness protocols developed by prior work motivate future perturbation, counterfactual, and representation invariance extensions of our benchmark.

A.6 Applied, Educational, Real World, and Multilingual Mathematical Reasoning

Another group of benchmarks evaluates mathematical reasoning in settings that are closer to education, scientific practice, or everyday visual experience. ScienceQA combines questions, explanatory text, diagrams, and scientific knowledge, making it a useful reference for grounded scientific reasoning (Lu et al., 2022). OlympiadBench evaluates difficult bilingual scientific problems, while EXAMS-V studies multilingual multimodal examination questions across academic disciplines (He et al., 2024; Das et al., 2024). MathScape and MathReal move toward real world mathematical contexts, including visual scenes and practical quantitative reasoning (Liang et al., 2024; Feng et al., 2025). MaRVL-QA investigates mathematical reasoning over visual landscapes, where spatial and contextual interpretation is intertwined with quantitative questions (Pande et al., 2025). MMTutorBench frames mathematical reasoning as an educational interaction involving explanation and feedback, and PISA-Bench uses multilingual assessment material to evaluate vision language models in an education oriented setting (Yang et al., 2025b; Haller et al., 2025). M3Kang extends the evaluation of multimodal mathematics to multilingual settings, while Math2Visual studies the generation of meaningful visual representations for mathematical word problems (Torres-Camps et al., 2026; Wang et al., 2025a). These works broaden the ecological validity of evaluation, but each is tied to a particular application, language setting, educational level, or visual scenario. Such specialization is valuable for deployment specific analysis, yet it makes it difficult to compare capabilities across unrelated mathematical domains. DeepVisualMath retains applied, contextual, and grounded reasoning as explicit categories while placing them alongside formal analysis, geometry, logic, and quantitative mathematics. This design supports both broad capability measurement and future domain specific subsets for educational, scientific, and practical applications.

A.7 Positioning DeepVisualMath

The comparison in Table 4 makes the central distinction between DeepVisualMath and prior benchmarks explicit. Existing resources often provide deep coverage of one representation or reasoning family, but their coverage is concentrated in a small subset of the 18 subcategories used in our taxonomy. Geometry3K, for example, is highly valuable for foundational geometric reasoning but does not evaluate calculus, probability, financial analytics, or physical systems. MathVerse and MathVista provide broader visual mathematics coverage, yet their category distributions remain different from a unified taxonomy of mathematical capabilities (Lu et al., 2023a; Zhang et al., 2024). MathVision, WE-MATH, and STELAR-VISION increase subject or representation diversity, but still leave substantial portions of the subcategory space untested (Cotilla et al., 2024; Qiao et al., 2025a; Li et al., 2025a). Specialized robustness benchmarks such as DynaMath, Math Blind, ZeroBench, and MV-MATH diagnose important failure modes, but they are not intended to serve as a complete map of mathematical competence (Zou et al., 2024; Gao et al., 2025; Roberts et al., 2025; Wang et al., 2025b). DeepVisualMath addresses this fragmentation by organizing 10,298 questions into 7 categories, 18 subcategories, and 80 fine grained tasks, with coverage across every subcategory shown in the comparison table. Its contribution is therefore not simply a larger collection of visual questions, but a common evaluation coordinate system that connects heterogeneous mathematical tasks. The automated curation pipeline preserves source diversity while normalizing task definitions, answer formats, and category assignments. The evaluation harness supports reproducible comparison across models, and the error analysis framework enables failures to be traced from aggregate categories to individual task types. This structure allows researchers to use DeepVisualMath as a broad diagnostic benchmark while continuing to use specialized datasets for controlled studies of geometry, visual invariance, tutoring, multilingual reasoning, or robustness. In this sense, DeepVisualMath complements rather than replaces prior benchmarks by providing the missing holistic layer required to understand how multimodal models distribute their mathematical abilities across visual reasoning domains.

Table 4: Comparison of Benchmarks Across Mathematical Reasoning Subcategories

Benchmark	Abstract Logic & Procedural Puzzles	Analytic & Transformation	Arithmetic & Concrete Number Theory	Calculus & Advanced Analysis	Combinatorics & Graph Structures	Comparative Relations	Foundational & Topological Geometry	Foundations of Algebra	Functional Analysis & Precalculus	Metric Measurement	Operational & Financial Analytics	Physical Systems Reasoning	Plane Coordinate & Transformation	Plane Measurement	Probability, Statistics & Analytics	Semantic & Grounded Representation	Solid Measurement	Solid Visualization & Projection
AdaptMMBench Zhang et al. (2026)							✓	✓							✓			
CogAlign Huang et al. (2025)						✓												✓
DynaMath_Sample Zou et al. (2024)	✓	✓	✓		✓			✓						✓	✓		✓	
MATHEMETRIC_Benchmark Gao et al. (2025)					✓									✓			✓	
MM-IQ Cai et al. (2025)													✓			✓		✓
MMMU Yue et al. (2023a)				✓	✓		✓				✓				✓			
MathNet Alshammari et al. (2026)			✓	✓	✓			✓						✓			✓	
MathReal Feng et al. (2025)	✓													✓			✓	
MathVerse Zhang et al. (2024)		✓							✓				✓	✓			✓	
MathVision Cotilla et al. (2024)	✓	✓	✓		✓		✓	✓		✓					✓		✓	✓
MathVista Lu et al. (2023a)	✓		✓					✓							✓			
OlympiadBench He et al. (2024)			✓		✓		✓	✓				✓						
R-Bench Guo et al. (2025)	✓				✓							✓				✓		
STELAR-topo_vision_reasoning_preference_50l et al. (2025a)		✓	✓		✓		✓	✓		✓					✓		✓	✓
ScienceQA Lu et al. (2022)											✓	✓						
Spatial-Visualization-Benchmark Wang et al. (2025c)	✓												✓					✓
We-Math2.0-Pro Qiao et al. (2025b)																	✓	
geometry3k Lu et al. (2021a)							✓											
mmtutorbench Yang et al. (2025b)				✓			✓	✓							✓			
u-math Chernyshev et al. (2024)				✓				✓										
zerobench Roberts et al. (2025)	✓																	
DEEPVISUALMATH(Ours)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 5: Hierarchical category breakdown (Part 1). Tasks are organised into three levels: (L1) domain categories, (L2) sub-categories, and (L3) fine-grained task types. Counts indicate the number of samples per task.

Category	Sub-Category	Task	Count
Discrete Mathematics & Formal Logic	Abstract Logic & Procedural Puzzles	Arrow Moving	50
		Block Moving	50
		Game	77
		Logic	78
		Logical Reasoning	50
		Puzzle	56
		Puzzle Test	50
	Arithmetic & Concrete Number Theory	Arithmetic	240
		Arithmetic Reasoning	50
		Divisibility / Factorization	47
		Math - Number Theory	50
		Numeric Commonsense and Arithmetic Reasoning	50
	Combinatorics & Graph Structures	Combinatorial Geometry	386
		Combinatorics	250
		Counting	120
		Graph Theory	188
		Graphs	50
		Math - Combinatorics	50
Solid (3D) Geometry	Solid Measurement	Solid Geometry	760
		Solid Geometry - Area	50
		Solid Geometry - Length	50
		Solid Geometry - Volume	50
	Solid Visualization & Projection	3D Rotation	48
		3D-Geometry	186
		Chart Projection	166
		CrossSection	73
		CubeCounting	50
		Descriptive Geometry	132
Continuous Mathematics & Analysis	Calculus & Advanced Analysis	Analysis and Calculus	50
		Calculus	87
		Differential Calculus	49
		Integral Calculus	47
	Foundations of Algebra	Algebra	671
		Algebraic Expressions	46
		Algebraic Reasoning	50
		Equations and Inequalities	45
		Geometry and Algebraic Reasoning	105
		Math - Algebra	50
	Functional Analysis & Precalculus	Functions - Applied	50
		Functions - Expression	50
		Functions - Property	50
	Probability, Statistics & Analytics	Arithmetic and Statistical Reasoning	50
		Probability and Statistics	50
		Scientific Figure	78
		Scientific Reasoning	50
		Statistical Reasoning	65
		Statistics	219
		statistical	48
Socio-Economic & Contextual Mathematics	Operational & Financial Analytics	Cost Management	50
		Economics	49
		Financial Accounting	61
		Investment	62
		Managerial Accounting	53
		Operations Research	49
	Semantic & Grounded Representation	Mathematical	149

Table 6: Hierarchical category breakdown (Part 2). Continuation of the category breakdown, showing the remaining domain categories along with the overall total.

Category	Sub-Category	Task	Count
Cross-Dimensional & Relational Geometry	Analytic & Transformation	Analytic Geometry	192
		Functions - Coordinate	50
		Transformation Geometry	199
	Comparative Relations	Angle Comparison	211
		Parallel Comparison	227
		Perpendicular Comparison	266
	Foundational & Topological Geometry	Geometry	572
		Math - Geometry	70
		Topology	50
	Metric Measurement	Metric Geometry - Angle	228
		Metric Geometry - Area	688
		Metric Geometry - Length	569
Applied Sciences & Grounded Reasoning	Physical Systems Reasoning	Physics	129
		Physics - Electromagnetism	50
		Physics - Mechanics	50
		Physics - Optics	46
		Physics - Thermodynamics	49
Plane (2D) Geometry	Plane Coordinate & Transformation	2D Rotation	50
		2D-Geometry	113
		Plane Geometry - Analytic	50
	Plane Measurement	Plane Geometry	277
		Plane Geometry - Angle	122
		Plane Geometry - Applied	55
		Plane Geometry - Area	50
		Plane Geometry - Length	95
Total			10298

B Task Taxonomy and Descriptions

DeepVisualMath evaluates visual mathematical reasoning through a three level hierarchy. At the first level, **L1**, we define seven domain categories that represent distinct mathematical abilities. At the second level, **L2**, each category is divided into subcategories according to the underlying reasoning operation. At the third level, **L3**, each subcategory is divided into fine grained task types with a clear problem structure and answer criterion. The taxonomy contains seven categories, eighteen subcategories, and eighty tasks. The descriptions below define the scope of every category, subcategory, and task in the benchmark.

Table 7: Distribution of instances across the seven top-level categories of DeepVisualMath.

Category	# Questions	# Sub-cats	# Tasks
Applied Sciences & Grounded Reasoning (Applied)	324	1	5
Continuous Mathematics & Analysis (Continuous)	1,910	4	20
Cross-Dimensional & Relational Geometry (Cross-Dim.)	3,322	4	12
Discrete Mathematics & Formal Logic (Discrete)	1,892	3	18
Plane (2D) Geometry (Plane)	812	2	8
Socio-Economic & Contextual Mathematics (Socio-Econ.)	473	2	7
Solid (3D) Geometry (Solid)	1,565	2	10
Total	**10,298**	**18**	**80**

B.1 Discrete Mathematics and Formal Logic

Discrete Mathematics and Formal Logic contains 1,892 instances that require reasoning over finite objects, symbolic rules, logical constraints, counting structures, and graph relations. These problems usually have a finite solution space, but solving them may require several dependent deductions rather than direct calculation.

B.1.1 Abstract Logic and Procedural Puzzles

This subcategory contains 411 instances involving rule based puzzles, procedural manipulation, and symbolic deduction. The visual input may show objects, states, moves, or constraints, while the question asks for a valid action, final configuration, or logical conclusion.

✓ **Arrow Moving.** Arrow Moving tasks require the model to follow legal arrow movements or direction changes and determine the resulting configuration. They test ordered state updates, direction tracking, and compliance with movement constraints.

✓ **Block Moving.** Block Moving tasks present blocks or tiles that must be moved under specified rules. The model must identify valid moves, track intermediate states, and determine whether a target arrangement can be reached.

✓ **Game.** Game tasks describe a finite game with players, actions, and outcomes. They require reasoning about legal strategies, winning conditions, turn order, or the consequences of a particular move.

✓ **Logic.** Logic tasks ask the model to evaluate statements, implications, conditions, or small formal systems. Correct solutions require identifying which conclusions follow from the given premises.

✓ **Logical Reasoning.** Logical Reasoning tasks involve several natural language or visual premises that must be combined to select the only consistent conclusion. They emphasize multi premise deduction rather than the interpretation of a single statement.

✓ **Puzzle.** Puzzle tasks cover general constraint problems in which the model must combine several clues to identify a valid object, ordering, assignment, or outcome. They test systematic elimination and consistency checking.

✓ **Puzzle Test.** Puzzle Test tasks use a standardized puzzle format with multiple conditions and candidate answers. The model must determine which candidate satisfies all constraints simultaneously.

B.1.2 Arithmetic and Concrete Number Theory

This subcategory contains 437 instances focused on numerical operations and elementary properties of integers. The visual material may provide numbers, symbols, quantities, or contextual information that must be converted into a valid arithmetic procedure.

✓ **Arithmetic.** Arithmetic tasks require addition, subtraction, multiplication, division, fractions, percentages, or other elementary numerical operations. They measure whether the model can extract the relevant quantities and compute the result accurately.

✓ **Arithmetic Reasoning.** Arithmetic Reasoning tasks embed numerical operations in a short problem description or visual context. The model must determine which quantities matter, select the correct operations, and combine them in the required order.

✓ **Divisibility and Factorization.** These tasks ask about factors, multiples, divisibility rules, prime decomposition, or common divisors. The model must apply integer properties rather than rely only on surface numerical comparison.

✓ **Mathematics and Number Theory.** This task type covers elementary number theoretic relations such as primes, remainders, congruences, parity, and integer sequences. Questions may require symbolic reasoning over several numerical conditions.

✓ **Numeric Commonsense and Arithmetic Reasoning.** These tasks require numerical judgment grounded in ordinary knowledge of scales, units, magnitudes, or plausible quantities. The model must combine basic arithmetic with a reasonable interpretation of the visual or contextual setting.

B.1.3 Combinatorics and Graph Structures

This subcategory contains 1,044 instances involving counting, arrangements, networks, and finite structural objects. The central challenge is to reason about relationships among many discrete elements without overlooking cases or counting the same configuration more than once.

✓ **Combinatorial Geometry.** Combinatorial Geometry tasks ask the model to count or characterize geometric objects formed by a finite collection of points, lines, regions, or shapes. They combine geometric interpretation with discrete enumeration.

✓ **Combinatorics.** Combinatorics tasks involve permutations, combinations, arrangements, cases, or selections subject to explicit constraints. The model must construct a complete counting argument and avoid double counting.

✓ **Counting.** Counting tasks ask for the number of objects, configurations, paths, or outcomes shown or described in the input. They test systematic enumeration over a finite visual or symbolic structure.

✓ **Graph Theory.** Graph Theory tasks require reasoning about vertices, edges, paths, cycles, connectivity, degrees, or graph properties. The model must translate the displayed graph into a formal relational structure before answering.

✓ **Graphs.** Graphs tasks focus on reading and interpreting graph structures, including adjacency, reachability, ordering, and local relations. They are usually grounded directly in a visual graph rather than in an extended theorem.

✓ **Mathematics and Combinatorics.** This task type contains textbook style combinatorial problems involving several formal constraints. It evaluates whether the model can combine symbolic notation with a complete finite case analysis.

B.2 Solid Three Dimensional Geometry

Solid Three Dimensional Geometry contains 1,565 instances involving volumes, areas, lengths, projections, rotations, cross sections, and spatial visualization. These tasks require the model to reconstruct a three dimensional object from a diagram or projection and then apply suitable geometric relations.

B.2.1 Solid Measurement

This subcategory contains 910 instances that ask for quantitative properties of three dimensional objects. The model must identify the relevant dimensions, select the correct formula, and account for hidden or composite structure.

✓ **Solid Geometry.** Solid Geometry tasks require general reasoning about polyhedra, cylinders, cones, spheres, prisms, and composite solids. Questions may involve several properties at once, such as determining one dimension from a known volume and surface area.

✓ **Solid Geometry and Area.** These tasks ask for the surface area of a three dimensional object, including the areas of visible, hidden, curved, or composite surfaces. The model must identify which faces or surfaces contribute to the requested quantity.

✓ **Solid Geometry and Length.** These tasks require finding a length, diagonal, height, radius, or distance in a solid figure. They often require a combination of spatial interpretation, similarity, the Pythagorean theorem, or coordinate reasoning.

✓ **Solid Geometry and Volume.** These tasks ask for the volume of a solid or a combination of solids. The model must decompose the figure when needed and combine the component volumes using the correct dimensions.

B.2.2 Solid Visualization and Projection

This subcategory contains 655 instances that test whether a model can mentally transform, project, or reconstruct three dimensional structures from visual representations.

✓ **Three Dimensional Rotation.** These tasks require predicting the appearance or orientation of an object after a specified rotation. The model must preserve the identity of faces, edges, labels, and relative positions during the transformation.

✓ **Three Dimensional Geometry.** Three Dimensional Geometry tasks ask the model to interpret spatial relations within a solid figure, such as incidence, parallelism, intersection, or relative position of components.

✓ **Chart Projection.** Chart Projection tasks require interpreting a three dimensional object or quantity through a projected chart, diagram, or visual representation. The model must connect the displayed projection to the underlying spatial or quantitative structure.

✓ **Cross Section.** Cross Section tasks ask which shape results when a solid is cut by a plane. They test whether the model can infer the intersection between the cutting plane and the three dimensional object.

✓ **Cube Counting.** Cube Counting tasks present stacks or arrangements of unit cubes and ask for the number of cubes, visible faces, layers, or missing elements. The model must reason about both visible and occluded parts of the structure.

✓ **Descriptive Geometry.** Descriptive Geometry tasks use multiple views, projections, or construction lines to represent a three dimensional object. The model must recover spatial properties from the available views and relate them consistently.

B.3 Continuous Mathematics and Analysis

Continuous Mathematics and Analysis contains 1,910 instances involving calculus, algebra, functions, probability, statistics, and scientific quantities. These problems require reasoning over continuously varying quantities, symbolic expressions, equations, curves, and measured data.

B.3.1 Calculus and Advanced Analysis

This subcategory contains 233 instances that involve rates of change, accumulation, limits, derivatives, integrals, and related analytical concepts.

✓ **Analysis and Calculus.** These tasks combine several ideas from calculus or real analysis, such as continuity, monotonicity, limits, derivatives, and integrals. The model must connect the visual or symbolic representation to the relevant analytical property.

✓ **Calculus.** Calculus tasks require applying standard differential or integral methods to a function, graph, or quantitative problem. They test both formula selection and interpretation of the result.

✓ **Differential Calculus.** These tasks focus on derivatives, slopes, local change, extrema, tangent lines, and related rates. The model must identify the relevant variable and interpret the derivative in the given context.

✓ **Integral Calculus.** Integral Calculus tasks involve antiderivatives, definite integrals, area, accumulation, or average value. The model must determine the correct bounds and interpret the geometric or physical meaning of the integral.

B.3.2 Foundations of Algebra

This subcategory contains 967 instances covering symbolic manipulation, equations, inequalities, algebraic structure, and algebraic interpretation of geometric information.

✓ **Algebra.** Algebra tasks require manipulating expressions, solving symbolic problems, or deriving an unknown quantity from several relations. They assess general symbolic fluency and consistency.

✓ **Algebraic Expressions.** These tasks ask the model to simplify, factor, expand, evaluate, or compare algebraic expressions. The visual input may provide a formula, diagram, or structured expression that must be transcribed correctly.

✓ **Algebraic Reasoning.** Algebraic Reasoning tasks require constructing an equation or relation from the information in the problem. The main challenge is translating the given conditions into a valid symbolic model.

✓ **Equations and Inequalities.** These tasks involve solving equalities or inequalities, often with restrictions on the allowed values. The model must preserve the solution set while applying algebraic transformations.

✓ **Geometry and Algebraic Reasoning.** These tasks connect a geometric configuration to algebraic variables and equations. The model must extract lengths, angles, coordinates, or areas from the figure and use them in a symbolic derivation.

✓ **Mathematics and Algebra.** This task type contains general textbook algebra problems with multiple symbolic steps. It evaluates whether the model can combine manipulation, substitution, and verification.

B.3.3 Functional Analysis and Precalculus

This subcategory contains 150 instances involving functions and their representations. The questions may use equations, tables, plots, mappings, or contextual descriptions.

✓ **Functions and Applications.** These tasks ask the model to use a function to describe or predict a quantity in an applied setting. The model must identify inputs, outputs, and the meaning of the functional relationship.

✓ **Functions and Expressions.** These tasks require evaluating, composing, inverting, or transforming functions expressed symbolically. Correct answers depend on preserving the domain and the order of operations.

✓ **Functions and Properties.** These tasks ask about properties such as injectivity, periodicity, monotonicity, symmetry, continuity, or growth. The model must infer the property from a formula, graph, table, or verbal description.

B.3.4 Probability, Statistics and Analytics

This subcategory contains 560 instances involving uncertainty, distributions, summary statistics, statistical plots, and scientific data interpretation.

✓ **Arithmetic and Statistical Reasoning.** These tasks combine numerical computation with statistical concepts such as means, rates, proportions, or variation. The model must select the correct summary or comparison and calculate it accurately.

✓ **Probability and Statistics.** These tasks ask about events, conditional probability, distributions, expectation, variance, or statistical conclusions. They require translating the visual or textual setup into a valid probabilistic model.

✓ **Scientific Figure.** Scientific Figure tasks require extracting values, trends, comparisons, or relationships from plots and scientific diagrams. The model must distinguish axes, legends, scales, and relevant data series.

✓ **Scientific Reasoning.** These tasks ask the model to infer a scientific conclusion from measured quantities, experimental evidence, or a visual scientific representation. They evaluate whether the model can connect observation to quantitative reasoning.

✓ **Statistical Reasoning.** Statistical Reasoning tasks require interpreting uncertainty, sampling, correlation, comparison, or evidence. The answer depends on understanding what a statistic means rather than merely computing it.

✓ **Statistics.** Statistics tasks focus on direct computation or interpretation of descriptive statistics, distributions, tables, and plots. They may involve averages, percentiles, variance, regression, or hypothesis related quantities.

✓ **`statistical`.** This source labeled task contains visually grounded statistical questions with varied formats. We retain the label to preserve the provenance of the source instances while evaluating the same core abilities of data interpretation and statistical calculation.

B.4 Socio Economic and Contextual Mathematics

Socio Economic and Contextual Mathematics contains 473 instances that place mathematical reasoning in financial, economic, managerial, operational, or social contexts. These tasks test whether a model can connect formal quantities to decisions and interpretations in realistic settings.

B.4.1 Operational and Financial Analytics

This subcategory contains 324 instances involving costs, investments, accounting records, economic quantities, and operational decisions. The model must identify the relevant quantities, apply the appropriate formula, and interpret the result in context.

✓ **Cost Management.** Cost Management tasks ask about fixed and variable costs, budgeting, margins, break even points, or cost comparisons. The model must combine numerical calculation with an interpretation of the management decision.

✓ **Economics.** Economics tasks involve supply, demand, prices, growth, elasticity, inflation, or other economic relationships. The model must read the visual or tabular evidence and determine the implied quantitative conclusion.

✓ **Financial Accounting.** These tasks require interpreting financial statements, transactions, balances, revenues, expenses, or accounting ratios. The model must preserve the accounting relationship while computing or comparing values.

✓ **Investment.** Investment tasks involve returns, interest, risk, cash flows, valuation, or portfolio comparisons. The model must apply the relevant financial formula and interpret the result over time.

✓ **Managerial Accounting.** Managerial Accounting tasks ask about internal planning, performance measurement, allocation, or decision analysis. They combine accounting quantities with a practical operational objective.

✓ **Operations Research.** Operations Research tasks involve optimization, scheduling, allocation, queues, transportation, or resource planning. The model must identify constraints and determine the best or feasible solution.

B.4.2 Semantic and Grounded Representation

This subcategory contains 149 instances in which the model must connect a mathematical concept to a visual, semantic, or contextual representation.

✓ **Mathematical.** Mathematical tasks cover general contextual mathematics that does not belong to a narrower specialized category. They may require interpreting notation, connecting a diagram to a mathematical idea, or selecting the concept that best explains the presented representation.

B.5 Cross Dimensional and Relational Geometry

Cross Dimensional and Relational Geometry contains 3,322 instances involving coordinates, transformations, angle and line relations, topology, and metric comparison. These tasks emphasize relationships between geometric objects rather than measurement of a single isolated figure.

B.5.1 Analytic and Transformation Geometry

This subcategory contains 441 instances that connect geometric objects with coordinates, equations, and transformations.

✓ **Analytic Geometry.** Analytic Geometry tasks require using coordinates, equations, slopes, or distance formulas to reason about points, lines, curves, and regions.

✓ **Functions and Coordinates.** These tasks interpret a function through a coordinate plane, table, or graph. The model must relate symbolic values to positions, intersections, or changes in the plotted representation.

✓ **Transformation Geometry.** Transformation Geometry tasks ask about translations, reflections, rotations, scaling, or compositions of transformations. The model must track how points and shapes change while preserving the appropriate invariants.

B.5.2 COMPARATIVE RELATIONS

This subcategory contains 704 instances that compare geometric relations among lines, rays, and angles. The focus is on identifying relative properties from a diagram rather than computing a single metric.

✓ **Angle Comparison.** Angle Comparison tasks ask which angle is larger, smaller, equal, or related to another angle. The model must compare visual or symbolic angle information without being misled by scale or orientation.

✓ **Parallel Comparison.** These tasks require identifying or comparing parallel lines and the angles or regions they create. The model must use corresponding, alternate, or related geometric relations.

✓ **Perpendicular Comparison.** These tasks ask whether lines or segments are perpendicular or compare relations induced by right angles. The model must recognize orthogonality from the diagram, coordinates, or stated constraints.

B.5.3 FOUNDATIONAL AND TOPOLOGICAL GEOMETRY

This subcategory contains 692 instances involving basic geometric structure, incidence, connectivity, continuity, and qualitative shape relations.

✓ **Geometry.** Geometry tasks ask about general relations among points, lines, shapes, and regions. They may combine definitions, diagram interpretation, and elementary theorems.

✓ **Mathematics and Geometry.** This task type contains broader textbook geometry problems that combine several geometric facts or require a multi step proof like derivation.

✓ **Topology.** Topology tasks focus on qualitative properties such as connectedness, inside and outside relations, continuity, holes, or deformation. Exact metric lengths are secondary to the underlying structural relation.

B.5.4 METRIC MEASUREMENT

This subcategory contains 1,485 instances that require measuring or comparing geometric quantities in a plane or coordinate representation.

✓ **Metric Geometry and Angle.** These tasks require calculating or comparing angles using geometric, trigonometric, or coordinate information. The model must identify the relevant rays and apply the correct relation.

✓ **Metric Geometry and Area.** These tasks ask for the area of a region, polygon, composite figure, or coordinate defined shape. The model must decompose the region or use an appropriate metric formula.

✓ **Metric Geometry and Length.** These tasks require finding distances, segment lengths, perimeters, or coordinate differences. They test scale interpretation and the correct use of distance or similarity relations.

B.6 APPLIED SCIENCES AND GROUNDED REASONING

Applied Sciences and Grounded Reasoning contains 324 instances that connect visual mathematical reasoning with physical laws and scientific quantities. The category tests whether a model can read a scientific diagram and apply the appropriate domain relation.

B.6.1 PHYSICAL SYSTEMS REASONING

This subcategory contains five task types covering common areas of physics. Questions may require extracting measurements from a figure before applying a physical law.

✓ **Physics.** General Physics tasks involve quantities such as force, energy, motion, pressure, temperature, or fields. The model must identify the governing relation and use the visual evidence to solve the problem.

✓ **Physics and Electromagnetism.** These tasks involve electric charge, current, voltage, resistance, magnetic fields, or electromagnetic forces. The model must interpret circuit diagrams, field directions, or quantitative relationships.

✓ **Physics and Mechanics.** Mechanics tasks focus on motion, force, momentum, work, energy, equilibrium, or collisions. The model must combine a visual configuration with the appropriate mechanical principle.

✓ **Physics and Optics.** Optics tasks involve reflection, refraction, lenses, mirrors, rays, or image formation. The model must trace visual paths and apply geometric or physical relations.

✓ **Physics and Thermodynamics.** These tasks involve heat, temperature, pressure, volume, work, or energy transfer. The model must interpret diagrams or state changes and apply the relevant thermodynamic relation.

B.7 Plane Two Dimensional Geometry

Plane Two Dimensional Geometry contains 812 instances involving planar shapes, coordinate figures, transformations, and measurements. These tasks focus on geometry that can be represented entirely in a two dimensional plane.

B.7.1 Plane Coordinate and Transformation

This subcategory contains 213 instances involving planar coordinates and transformations of two dimensional objects.

✓ **Two Dimensional Rotation.** These tasks ask the model to determine the position or orientation of a point or shape after a planar rotation. The model must track the center, direction, and angle of rotation.

✓ **Two Dimensional Geometry.** These tasks require reasoning about planar figures, lines, polygons, and their relations. They may involve incidence, symmetry, similarity, or elementary geometric properties.

✓ **Plane Geometry and Analytic Methods.** These tasks combine planar geometry with coordinates, equations, slopes, or distance formulas. The model must translate between the diagram and its analytic representation.

B.7.2 Plane Measurement

This subcategory contains 599 instances that ask for quantitative properties of planar figures.

✓ **Plane Geometry.** General Plane Geometry tasks require solving a problem about planar shapes using one or more geometric relations. They may combine angles, lengths, areas, and auxiliary constructions.

✓ **Plane Geometry and Angle.** These tasks ask for unknown or comparative angles in a planar diagram. The model must use angle sums, parallel line relations, cyclic properties, or other geometric constraints.

✓ **Plane Geometry and Applications.** These tasks place planar measurement in an applied setting such as design, construction, navigation, or layout. The model must interpret the context before applying a geometric formula.

✓ **Plane Geometry and Area.** These tasks require finding the area of a triangle, polygon, circle, sector, or composite planar region. The model must identify the correct boundaries and combine component areas when necessary.

✓ **Plane Geometry and Length.** These tasks ask for a side, diagonal, radius, perimeter, or distance in a planar figure. The solution may require similarity, trigonometry, coordinates, or the Pythagorean theorem.

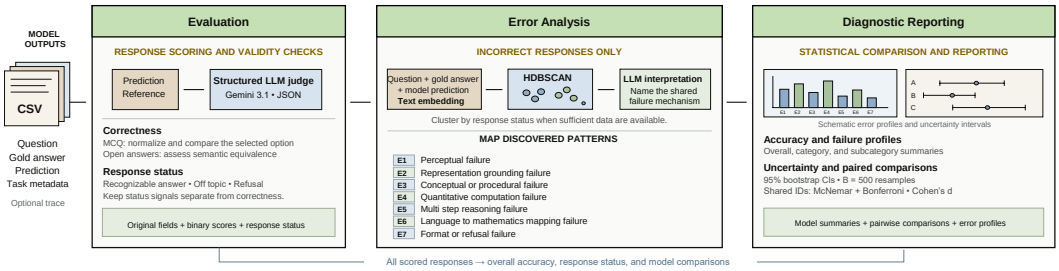


Figure 5: **Automated evaluation and error analysis pipeline of DEEPVISUALMATH.** DEEPMATHEVALUATOR scores response correctness and status, clusters incorrect responses into patterns mapped to seven error types, and produces diagnostic profiles with uncertainty estimates and paired model comparisons.

B.8 TAXONOMY SUMMARY

Together, these categories cover foundational computation, symbolic manipulation, discrete structures, planar and solid geometry, scientific reasoning, and contextual quantitative analysis. The hierarchy allows DEEPVISUALMATH to report performance at the instance, task, subcategory, and category levels. It also supports targeted diagnosis: a model may perform well on arithmetic while failing on visual geometry, or solve direct measurements while struggling with transformations, topology, or multi step combinatorial reasoning. The complete task counts and category distribution are reported in Tables 5 and 6.

C AUTOMATED DATASET CURATION PIPELINE

In this appendix, we describe the complete three stage automated curation pipeline used to construct DEEPVISUALMATH from approximately 84,000 candidate instances collected from 21 source datasets. We design the pipeline as a reproducible and transferable methodology for building multimodal reasoning benchmarks, rather than as a collection of ad hoc dataset cleaning steps. Each stage applies explicit and auditable criteria, records the reason for every rejection, and operates at a different cost and quality level. After all three stages, 10,298 instances remain. The final benchmark therefore contains items that satisfy structural validity, visual uniqueness, taxonomy coverage, mathematical relevance, reasoning depth, visual dependence, and language clarity.

Table 8 summarizes the number of instances retained and removed at each stage.

Table 8: Instance counts and reductions at each pipeline stage.

Stage	Step	Remaining	Reduced By
Raw pool	—	84,000	—
Stage 1 Structural Heuristics	1a. Metadata integrity	76,987	7,013
	1b. Publication recency	54,714	22,273
	1c. Question length filter	45,174	9,540
	1d. Deduplicated & balanced	37,548	7,626
	1e. Taxonomy remapping & pruning	26,943	10,605
Stage 2 Vision-Space Deduplication	2a+2b. Near-duplicate detection	23,468	3,475
	2c. Visual domain clustering	18,657	4,811
Stage 3 Multi-Criteria Quality Filtering	M1. Mathematical domain	—	2,795
	M2. Reasoning depth	—	1,536
	M3. Visual dependency	—	6,018
	M4. Language clarity	—	1,685
Final benchmark	—	**10,298**	**73,702**

We deliberately increase the computational cost across stages while reducing the aggressiveness of the filters. Stages 1 and 2 are deterministic and do not use an LLM, whereas Stage 3 applies

multimodal LLM based quality assessment. This ordering allows us to reserve expensive evaluation for candidates that have already passed inexpensive structural and visual redundancy checks.

C.1 Stage 1: Deterministic Eligibility Screening (84K $\rightarrow$ 26,943)

In Stage 1, we apply five deterministic filters in sequence. These checks are fast, reproducible, and free of LLM inference. They remove invalid, underspecified, duplicated, or out of scope instances before any expensive processing takes place. We also record the rejection reason for every removed item, which makes the stage fully auditable.

1a. Metadata integrity ($-7{,}013$). We discard an instance when its image path is missing or cannot be resolved to a valid file, when a multiple choice answer is not included among the provided options, or when an open ended answer field is empty or null. This check removes broken records inherited from heterogeneous source datasets and prevents malformed examples from reaching later stages.

1b. Publication recency ($-22{,}273$). We flag instances originating from datasets published before 2020 for review. We retain such items only when they belong to landmark collections that remain relevant and have no clear modern replacement. All other instances are removed. This filter produces the largest reduction because the raw pool contains many older collections with outdated annotation conventions and inconsistent problem formats.

1c. Question length ($-9{,}540$). We retain questions whose length falls between 10 and 200 tokens. Very short questions, such as "*What is the answer?*," do not provide enough information for reliable reasoning evaluation. Very long questions often contain extended textual explanations or answer revealing context, which can make the item solvable without using the visual input.

1d. Deduplication and source balancing ($-7{,}626$). We lowercase and normalize whitespace in every question, then compute a SHA-256 hash. When multiple source datasets contain the same normalized question, we retain a single copy, preferring the instance with richer metadata or higher resolution imagery. In the same step, we balance the contribution of individual source datasets so that no single source dominates the retained pool or determines the benchmark's question style and difficulty distribution.

1e. Taxonomy remapping and scope filtering ($-10{,}605$). We map each source label to the hierarchical taxonomy of DeepVisualMath (Section B). We remove every instance that cannot be assigned to one of the 80 tasks after remapping. This step turns our taxonomy into an explicit quality gate: every retained question must correspond to a defined mathematical reasoning capability. When adapting the pipeline to another reasoning domain, the domain specific remapping rules can be changed without modifying the other Stage 1 filters.

C.2 Stage 2: Visual Redundancy and Balance Control (26,943 $\rightarrow$ 18,657)

In Stage 2, we remove redundancy that cannot be detected from question text alone and control the distribution of visual content. Multi source benchmarks often contain the same diagram, chart, or figure under different questions or minor rendering variations. Such duplication can inflate apparent performance and reduce the effective diversity of the evaluation set. We therefore perform all operations in visual embedding space.

2a–2b. Image encoding and near duplicate detection ($-3{,}475$). We encode every unique image with a CLIP ViT-L/14 encoder. For instances containing multiple images, we compute the mean pooled embedding across the image list. We store the embeddings in a FAISS flat inner product index and compare image pairs using cosine similarity. We treat pairs with similarity greater than 0.97 as near duplicates, including the same figure rendered, cropped, or formatted in slightly different ways. Within each duplicate cluster, we retain the instance with the strongest composite text quality signal, defined by question length and answer specificity, and remove the remaining instances.

2c. Visual domain balancing ($-4{,}811$). We apply K means clustering with $K = 25$ to the image embeddings and use the resulting clusters to identify broad visual domains, such as plane geometry diagrams, solid geometry renderings, statistical charts, algebraic plots, and applied science figures. When a cluster contains more than 15% of the post deduplication pool, we apply stratified downsampling to reduce its dominance. This step prevents the benchmark from being driven by one recurring figure type, a failure mode that is easy to miss when balancing only question text. The resulting pool

of 18,657 instances has a more balanced visual distribution, which we verify using cluster entropy before moving to Stage 3.

C.3 Stage 3: Multimodal Quality and Reasoning Audit ($18{,}657 \rightarrow 10{,}298$)

In Stage 3, we apply four quality criteria through structured multimodal LLM scoring. Each criterion targets a failure mode that cannot be reliably detected by the deterministic filters. We retain an instance only when it passes all four criteria.

M1. Mathematical domain relevance ($-2{,}795$). We assess whether solving the instance genuinely requires mathematical reasoning, rather than surface pattern matching, visual recognition, or general commonsense. We remove items that carry a mathematical source label but can be solved through non mathematical shortcuts.

M2. Reasoning depth ($-1{,}536$). We remove items that require only a single rote operation and provide little meaningful inference. This criterion preserves questions with a non trivial reasoning chain and supports more informative profiling of model capabilities.

M3. Visual dependence ($-6{,}018$). We hide the associated image or figure and assess whether the question can still be answered reliably from its text alone. If the answer remains available through a language shortcut, we discard the instance. This filter ensures that the visual input contributes essential evidence to the solution and produces the largest reduction in Stage 3, showing that many aggregated source questions can be answered without actually interpreting their images.

M4. Language clarity ($-1{,}685$). We remove instances with ambiguous wording, translation artifacts, incomplete context, or underspecified problem statements. This final gate ensures that each retained question is stated clearly enough for differences in model performance to reflect reasoning ability rather than avoidable linguistic noise.

C.4 Generalizability of the Pipeline

We design the pipeline to transfer across multimodal reasoning domains. The eligibility checks in Stage 1 apply to any benchmark pool containing images, questions, answers, and metadata. The visual deduplication and clustering operations in Stage 2 also remain unchanged apart from the choice of image encoder and balancing thresholds. In Stage 3, the quality criteria retain their general meaning, while the domain relevance prompt can be adapted to the target capability. To construct a benchmark for another domain, such as mathematical or causal reasoning, we therefore need to redefine only the taxonomy mapping rules in Step 1e and the domain relevance criterion M1. The remaining filters can be transferred without modification.

We evaluate the contribution of the stages through a checkpoint ablation at approximately 84K, 27K, 19K, and 10.3K instances. At each checkpoint, we measure benchmark statistics and model performance. These experiments show that the stages contribute independently: structural screening removes invalid records, visual processing reduces cross source redundancy, and multimodal quality auditing improves the specificity of task level diagnostics.

D Automated Evaluation Pipeline

This appendix specifies DeepMathEvaluator, the automated evaluation pipeline used to score, diagnose, and compare model outputs across the mathematical reasoning tasks in DeepVisual-Math. We design the pipeline around three requirements. First, it must scale to all evaluated models and tasks without requiring repeated human annotation. Second, it must support statistically principled comparisons, including uncertainty estimates, multiple testing correction, and effect sizes. Third, it must be resumable: partial runs caused by API failures or hardware interruptions should not require already processed responses to be scored again.

The pipeline consists of three distinct components: *Response Scoring and Validity Checks*, *Failure Pattern Discovery*, and *Statistical Comparison and Reporting*. The first component converts free form model responses into structured correctness signals. The second organizes failed responses into interpretable error patterns. The third summarizes model performance and performs paired statistical

comparisons. Section D.1 gives an overview, Sections D.2–D.3 describe the three components, Section D.4 defines the output files, and Section D.5 discusses design limitations.

D.1 PIPELINE OVERVIEW

DEEPMATHEVALUATOR takes as input a root directory containing one subfolder per evaluated model. Each folder contains an `agent_response.csv` file with the columns `id`, `task_description`, `answer`, and `response`. We then process each model folder through the following three components:

Component 1. Response Scoring and Validity Checks. The configured LLM judge evaluates each response using structured binary criteria and writes the results to `agent_scores.csv`.

Component 2. Failure Pattern Discovery. Incorrect responses are embedded, reduced, and clustered. Each discovered cluster is assigned an interpretable label from the E1–E7 mathematical error taxonomy.

Component 3. Statistical Comparison and Reporting. We compute bootstrap confidence intervals, paired McNemar tests with Bonferroni correction, and Cohen's d effect sizes. The results are exported to `model_evaluations.csv` and `model_evaluations.json`.

The entry point is the `DeepMathEvaluator` constructor, which accepts the parameters summarized in Table 9.

Table 9: DEEPMATHEVALUATOR API parameters.

Parameter	Default	Description
`results_path`	*required*	Root directory containing one subfolder per model, each with an `agent_response.csv` file.
`max_concurrent`	`10`	Number of workers used for parallel response scoring.
`model_name`	`as needed`	Identifier of the LLM used for response scoring and error cluster labeling.

D.2 COMPONENT 1: RESPONSE SCORING AND VALIDITY CHECKS

Scoring dimensions. For each row in `agent_response.csv`, the LLM judge receives the task description, ground truth answer, and model response. It returns a strict JSON object containing the binary scores defined in Table 10. For the results reported in this paper, we use Gemini 3.1 as the judge.

Table 10: Binary response level signals produced by Component 1.

Dimension	Range	Definition
`accuracy_score`	$\{0, 1\}$	The final answer is correct, including semantic equivalence for open ended responses.
`off_topic_score`	$\{0, 1\}$	The response fails to address the question and instead gives unrelated or tangential content.
`refusal_score`	$\{0, 1\}$	The model declines to provide a final answer, for example by citing insufficient information or inability to solve the problem.

Parallel scoring and resumability. We parallelize scoring with a `ThreadPoolExecutor` using `max_concurrent` workers. Each worker submits one scoring request and retries up to five times after transient failures. If a row remains unscored, we record its identifier and preserve the original response rather than silently deleting it. We save the partially completed result after each finished request. On a subsequent run, DEEPMATHEVALUATOR reads the existing `agent_scores.csv`, identifies rows with non null `accuracy_score`, and sends only the remaining rows for scoring. This makes the evaluation resumable and avoids unnecessary API calls.

Judge prompt and answer normalization. The scoring prompt requires the judge to return only a valid JSON object, with no markdown fences or trailing commas. Missing values are represented by `null`. We use a temperature of 0.1 to reduce scoring variability and allow up to 1,000 output tokens for the optional reasoning field. For multiple choice questions, we normalize the response and compare the selected option with the reference answer. For open ended questions, the judge assesses semantic equivalence so that correct paraphrases are not marked incorrect solely because their wording differs

D.3 Component 2: Statistical Comparison and Reporting

Bootstrap confidence intervals. For each model, we compute a 95% percentile bootstrap confidence interval over the `accuracy_score` values using $B = 500$ resamples:

$$\widehat{\mu}_{\text{boot}} = \frac{1}{B}\sum_{b=1}^{B} \bar{X}_b, \qquad \text{CI}_{95} = [\hat{q}_{2.5}, \hat{q}_{97.5}]\,, \tag{1}$$

where $\bar{X}_b$ is the mean accuracy of bootstrap resample b and $\hat{q}_p$ is the empirical pth percentile of the bootstrap means. Rows with missing accuracy scores are excluded before resampling. If a model has no valid scored rows, its summary statistics are recorded as `NaN`.

Paired McNemar comparison. We compare every pair of models using a continuity corrected McNemar test on the shared task IDs. Let b denote the number of instances on which model 1 succeeds and model 2 fails, and let c denote the number on which model 1 fails and model 2 succeeds. We compute

$$\chi^2_{\text{Mc}} = \frac{(|b-c|-1)^2}{b+c}. \tag{2}$$

The corresponding p value is obtained from a $\chi^2(1)$ distribution. If $b + c = 0$, the two models agree on every shared instance, so we set the statistic to zero and the p value to one.

Multiple testing correction. If M models are evaluated, we apply the Bonferroni correction across the $\binom{M}{2}$ pairwise comparisons:

$$\alpha_{\text{Bonf}} = \frac{0.05}{\binom{M}{2}}.$$

A pair is marked statistically significant when its corrected comparison satisfies $p < \alpha_{\text{Bonf}}$.

Effect size. For each ordered model pair (m_1, m_2), we compute Cohen's d:

$$d = \frac{\bar{X}_{m_1} - \bar{X}_{m_2}}{s_p}, \qquad s_p = \sqrt{\frac{(n_1-1)s_1^2 + (n_2-1)s_2^2}{n_1+n_2-2}}, \tag{3}$$

where s_j^2 is the sample variance of model j's score vector and n_j is its number of valid instances. The reverse comparison receives $-d$, making the resulting effect size matrix antisymmetric.

Error type comparisons. We repeat the same pairwise analyses for each error type $e \in \{\text{E1}, \ldots, \text{E7}\}$. For a given error type, the binary score is one when an instance is assigned to that error cluster and zero otherwise. This produces model comparisons at the level of specific failure patterns rather than only at the level of overall accuracy.

D.4 Output Schema

`agent_scores.csv`. This file is written separately for each model. It contains all columns from the original `agent_response.csv`, the three binary response scores from Component 1, the HDBSCAN `cluster` identifier, and the `error_type` assigned by Component 2. A cluster value of -1 denotes noise.

`model_evaluations.csv`. This file contains one row for each `(model, metric, error_type)` combination. The `to` field is populated for pairwise comparisons and is `None` for model level summaries. The supported metrics are `Score`, `Refusal Score`, `Off Topic Score`, `bootstrap_mean`, `cohens_d`, `bonferroni_alpha`, and `mcnemar_stat`.

model_evaluations.json. This file stores a nested dictionary keyed by model name. Each model entry contains `bootstrap_ci`, `average_score`, `overall_score`, `refusal_score`, `off_topic_score`, `Cohen_d`, `Bonferroni`, and `Mcnemar`. The last three fields contain dictionaries of pairwise comparisons indexed by error type.

D.5 Limitations and Design Trade-offs

Dependence on the automated judge. The quality of Component 1 depends on the calibration of the judge model for the mathematical reasoning tasks being evaluated. We reduce variability by using a low temperature, a fixed structured prompt, and strict JSON parsing. The separate judge validation study in Section F.2 shows high agreement between Gemini 3.1 and human annotations, but the judge remains a high agreement automated evaluator rather than a perfect substitute for human assessment. We randomly took 200 samples to check if the curation judge is working at human capacity and we find cohen's kappa of 0.88; showing the judge is very consistent with humans.

Small failure sets. HDBSCAN requires a sufficiently large set of failed responses to identify stable clusters. For models with very high accuracy, there may be too few failures to form two valid clusters. In such cases, the pipeline records no error clusters and retains `error_type = None`. This outcome is logged and is treated as an absence of sufficient evidence for clustering rather than as a pipeline failure.

Task alignment for pairwise tests. McNemar tests and effect sizes require instance level alignment by `id`. When two models are evaluated on different task subsets, the statistics are computed only on their shared instances. The intersection size should therefore be checked before interpreting comparisons based on a small number of shared items.

Failure logging and recovery. All transient scoring failures and unassigned clustering outcomes are recorded explicitly. Partial output files are saved during execution, allowing the pipeline to resume from the last completed row. This design improves reproducibility while preserving the original response whenever automated scoring cannot be completed.

E Automated Error Analysis Pipeline

Aggregate accuracy indicates whether a model succeeds, but it does not explain why a response fails. To obtain more informative diagnostics, we use a three component error analysis pipeline that converts raw model outputs into structured failure profiles. The pipeline consists of Response-Level Failure Characterization, Inductive Error Pattern Discovery, and Cross-Model Diagnostic Reporting. These components separate response validity, error pattern discovery, and statistical comparison, allowing us to analyze failures across models, categories, and mathematical tasks.

E.1 Stage 1: Response-Level Failure Characterization

Input format. Stage 1 accepts a JSON or CSV file containing, at minimum, the fields `id`, `question`, `choices` for multiple choice items, `answer`, `prediction`, `category`, and `sub_category`. An optional `reasoning_trace` field may also be stored when a model produces an explicit reasoning trace.

Correctness signal. For multiple choice items, we normalize the predicted answer and compare it with the gold option. For open ended items, we use the structured LLM judge to assess semantic equivalence between the prediction and the reference answer. This allows the evaluator to distinguish genuinely incorrect mathematical answers from correct answers expressed using different wording or notation.

Response status. We also record whether the response is valid, off topic, or a refusal. A response is marked as `valid` when it contains a recognizable final answer, `refused` when the model declines to answer or cites inability to solve the problem, and `off_topic` when it does not address the question. These signals are stored separately from correctness because a wrong mathematical answer, a refusal, and an irrelevant response represent different failure behaviors.

Failure routing. The output of Stage 1 is an enriched prediction file containing the original instance fields, the correctness score, response status, and task metadata. Incorrect responses are then routed to Stage 2 for error analysis. The response level signals remain available as auxiliary variables and are not collapsed into a single error label before clustering.

E.2 Stage 2: Inductive Error Pattern Discovery

Stage 2 identifies recurring failure patterns from the incorrect responses. Rather than grouping failures only by benchmark category, we analyze the relationship between the question, the reference answer, and the model prediction. This allows similar mistakes from different mathematical domains to be grouped together while separating different failure mechanisms within the same category.

E.2.1 Step 2a: Failure Representation

For each incorrect instance, we construct a composite text representation by concatenating the question, the gold answer, and the model prediction. We encode this representation using the configured text embedding model, such as `e5-large-v2` or `text-embedding-3-large`. Embedding the complete failure record rather than the question alone emphasizes the difference between the requested mathematical operation, the correct solution, and the model's response.

We perform this operation separately for each response status when sufficient data are available, so that refusals, off topic outputs, and substantive mathematical errors do not dominate one another during clustering.

E.2.2 Step 2b: Density Based Clustering

We apply HDBSCAN to the failure embeddings. We use density based clustering because failure patterns may have different sizes and densities, and because some responses may represent isolated mistakes rather than a systematic failure mode. HDBSCAN can identify dense groups without requiring a fixed number of clusters and can assign atypical responses to a noise class. A cluster identifier of -1 denotes a noise instance that does not belong to a stable failure group.

The minimum cluster size is selected from a predefined range using cluster quality and interpretability checks. For a point x, the silhouette coefficient is

$$s(x) = \frac{b(x) - a(x)}{\max\{a(x), b(x)\}}, \tag{4}$$

where $a(x)$ is the mean distance to points in the same cluster and $b(x)$ is the smallest mean distance to points in another cluster. We use the resulting clusters only when they contain a sufficiently coherent set of failures.

E.2.3 Step 2c: Cluster Interpretation

For each non noise cluster, we select representative instances near the cluster centre and provide them to an LLM with a structured labeling prompt. The prompt asks the model to identify the common failure mechanism, provide a short descriptive name, and define the error in one sentence. The prompt explicitly instructs the judge to describe what the model did incorrectly rather than simply naming the topic of the question. This prevents the analysis from reproducing the benchmark taxonomy instead of identifying cross cutting failure modes.

E.2.4 Step 2d: Reference Error Taxonomy

The discovered cluster labels are mapped to the following fixed seven type reference taxonomy. The inductive clusters preserve model specific patterns, while the fixed labels support comparison across models.

E1: Perceptual failure.
The model fails to detect or correctly read a visual element, such as a symbol, digit, operator, curve, or geometric feature.

E2: Representation grounding failure.
The model identifies the relevant visual elements but assigns them the wrong mathematical meaning or notation.

E3: Conceptual or procedural failure.
The model understands the setup but applies an incorrect theorem, formula, rule, or mathematical operation.

E4: Quantitative computation failure.
The model selects the correct procedure but makes an arithmetic, algebraic, or numerical mistake during execution.

E5: Multi-step reasoning failure.
The model solves one or more intermediate steps but fails when combining those results into the final conclusion.

E6: Language to mathematics mapping failure.
The model misinterprets the mathematical relation expressed in the question, such as confusing "at least" with "at most" or "difference" with "sum."

E7: Format or refusal failure.
The model does not provide a usable final answer because it refuses, produces an incomplete response, or violates the expected answer format.

This representation allows us to distinguish failures caused by perception, mathematical interpretation, computation, reasoning depth, language understanding, and answer production. It also makes it possible to compare the error profiles of models that have similar aggregate accuracy but fail on different types of instances.

E.3 Stage 3: Cross-Model Diagnostic Reporting

Stage 3 converts the error labels into model level profiles and statistically supported comparisons. For each model, we compute the proportion of failures assigned to each error type:

$$\mathbf{p} = (p_1, \ldots, p_7), \qquad p_k = \frac{\#\{\text{failures assigned to } \mathbf{Ek}\}}{\#\{\text{all analyzed failures}\}}.$$

We report these profiles overall and, where sample sizes permit, within each benchmark category and subcategory. This distinguishes models that make many perceptual errors from models that identify the visual structure correctly but fail during mathematical derivation.

Bootstrap confidence intervals. We compute 95% percentile bootstrap confidence intervals using $B = 500$ resamples. For each resample b, we calculate the relevant accuracy or error rate $\bar{X}_b$ and report

$$\mathrm{CI}_{95\%} = [\hat{q}_{2.5}, \hat{q}_{97.5}], \tag{5}$$

where $\hat{q}_p$ is the empirical pth percentile of the bootstrap distribution.

Paired model comparisons. For each pair of models, we apply McNemar's test to their binary correctness vectors on the shared instances. Let b and c denote the two discordant counts. We compute

$$\chi^2_{\mathrm{Mc}} = \frac{(|b - c| - 1)^2}{b + c}, \tag{6}$$

and obtain the p value from a $\chi^2(1)$ distribution. We apply the Bonferroni correction across all pairwise model comparisons and report Cohen's d alongside statistical significance. This ensures that diagnostic claims are based on both inferential evidence and practical effect size.

Diagnostic summaries. The final output contains per model accuracy, error type frequencies, category level summaries, bootstrap intervals, pairwise test results, and effect sizes. These structured results support reproducible tables and plots showing which failure modes are common across models and which are model-specific. All intermediate files, cluster assignments, and labels are retained so that each diagnostic result can be traced back to the underlying prediction records.

Limitations. The quality of the analysis depends on the judge's assessment of open ended answers and on the stability of the discovered clusters. Models with very few incorrect responses may not

Table 11: Overall model scores and 95% percentile bootstrap confidence intervals ($B = 500$). Values are percentages.

Model	Score	Bootstrap mean	95% CI	Bootstrap SD
GPT-5.4 mini	91.45	91.41	[90.22, 92.60]	0.65
Claude Sonnet 4.6	80.91	80.86	[79.04, 82.54]	0.92
Kimi K2.5	68.15	68.10	[65.67, 71.00]	1.36
Gemma 4 31B it	65.53	65.51	[63.06, 68.01]	1.29
Gemini 2.5 Flash	62.58	62.52	[59.76, 65.03]	1.37
Gemini 3.1 Flash Lite Preview	61.90	62.04	[58.85, 64.78]	1.48
Gemini 3.5 Flash	48.71	48.73	[47.26, 50.18]	0.78
Qwen3.6 Plus	31.05	31.03	[29.44, 32.69]	0.90
Qwen3.7 Flash	30.29	30.38	[28.15, 32.84]	1.23
GLM-4.5V	27.99	27.99	[26.60, 29.54]	0.78

Table 12: Summary of paired McNemar comparisons after Bonferroni correction.

Error	Entries	Threshold	Significant	Fraction
E1	20	0.001111	12	60.0%
E7	90	0.001111	52	57.8%

produce enough examples for reliable clustering. Pairwise tests also require instance level alignment; when two models are evaluated on different subsets, we restrict comparisons to their shared instances. These limitations are recorded in the output and should be considered when interpreting small error groups.

F More Details on Methodology

F.1 Evaluation Stability: Statistical Comparison and Reporting

As detailed in Appendix D.3, we complement the point estimates with uncertainty intervals, paired comparisons, multiplicity control, and effect sizes. The analysis covers ten models evaluated on the same task instances. Scores are reported as proportions in the statistical analysis and are converted to percentages in the descriptive table below.

Bootstrap confidence intervals. Bootstrap confidence interval estimates are shown in Table 11. The ranking is headed by GPT-5.4 mini (91.45%), followed by Claude Sonnet 4.6 (80.91%). The intervals for the two leading models do not overlap with each other or with the next model, indicating a substantial separation in the observed scores. The intervals for Gemini 2.5 Flash and Gemini 3.1 Flash Lite Preview overlap, so their ordering should not be interpreted as a definitive difference without the paired test.

Paired McNemar comparison. We compared each model pair on shared task IDs using the continuity-corrected McNemar statisticas described before. With $M = 10$ models, there are $\binom{10}{2} = 45$ comparisons and the Bonferroni threshold is 0.001111. The JSON report stores paired McNemar results for the error-specific binary outcomes E1 and E7. Table 12 summarizes the complete set of stored comparisons.

The pairwise results therefore indicate broad separation among models on the error outcomes, particularly for comparisons involving the strongest and weakest systems. These tests are paired and consequently account for the fact that the same task instances are evaluated by both models.

Effect sizes. For each ordered pair, the report computes Cohen's d using the pooled standard deviation, as described before. The reverse ordered comparison has the opposite sign. Table 13 reports the observed ranges and the largest absolute effect for each error type.

Thus, the error-level comparisons combine statistical significance with practical magnitude: E1 reaches a large effect at its most separated pair, whereas E7 reaches a very large effect. Because the effect sizes are based on binary error indicators, they should be interpreted as standardized separation in error incidence rather than as a continuous quality scale.

Table 13: Cohen's d summary over ordered model pairs.

Error	Minimum d	Maximum d	Maximum $\lvert d \rvert$
E1	-1.648	1.648	1.648
E7	-3.208	3.208	3.208

Key Evaluation Stability Findings. The evaluation is stable, with 500-resample bootstrap confidence intervals remaining narrow for all models, ranging from approximately ±0.8 to ±2.5 percentage points. GPT-5.4 mini achieved the highest score at 91.45% (95% CI: 90.22–92.60%), followed by Claude Sonnet 4.6 at 80.91% (79.04–82.54%). The ranking is consistent across the evaluation, with the strongest and weakest models clearly separated and only closely ranked models showing overlapping intervals. Paired McNemar tests identified significant differences in 60.0% of E1 comparisons and 57.8% of E7 comparisons after the strict Bonferroni correction ($\alpha = 0.00111$). The corresponding Cohen's d values reached 1.65 for E1 and 3.21 for E7, indicating that the evaluation captures substantial and practically meaningful differences between models.

F.2 Establishing AI Judge Performance

We use Gemma-4-31B-it as the automated judge for the binary evaluation task. To assess whether its decisions are sufficiently reliable for large scale evaluation, we construct an independently annotated reference set of $N = 320$ instances. Each instance receives binary judgments from multiple human annotators and one Gemma-4-31B-it judgment under the same evaluation rubric. We use the majority human label as the reference target for measuring AI judge performance. We also report human human agreement to characterize the practical agreement ceiling imposed by the task and the annotation protocol.

For an instance i, let $y_i^* \in \{0, 1\}$ denote the majority human label and $\widehat{y}_i^{\mathrm{AI}}$ denote the Gemma-4-31B-it prediction. We compute accuracy, balanced accuracy, Cohen's κ, Matthews correlation coefficient, F1, precision, recall, mean absolute error, and root mean squared error. Because the labels are binary, MAE is the fraction of incorrect predictions and RMSE is the square root of this quantity. Table 14 compares Gemma-4-31B-it with the human majority and the corresponding human human agreement.

Table 14: Agreement and classification performance for Gemma-4-31B-it and human annotations on the 320 item judge validation set.

Metric	Gemma-4-31B-it vs. human majority	Human human agreement
Accuracy	96.56%	99.17%
Balanced accuracy	96.72%	99.21%
Cohen's κ	0.930	0.983
Matthews correlation	0.931	0.983
F1	0.961	0.990
Precision	0.944	0.985
Recall	0.978	0.995
MAE	0.034	0.008
RMSE	0.185	0.090

Gemma-4-31B-it agrees with the majority human label on 309 of the 320 instances, corresponding to an accuracy of 96.56%. Its balanced accuracy is 96.72%, which shows that the high agreement is not explained by a strong preference for the more frequent class. The agreement coefficient is also high: Cohen's $\kappa = 0.930$ and the Matthews correlation coefficient is 0.931. The F1 score of 0.961 combines high precision and recall, with precision 0.944 and recall 0.978 for the positive class. The small MAE and RMSE values are consistent with the low number of disagreements. Nevertheless, every metric is weaker than the corresponding human human value, indicating that the judge is highly aligned with human decisions but does not reproduce them perfectly.

The error pattern is shown in Table 15. Gemma-4-31B-it correctly predicts 175 negative and 134 positive instances. It produces three false negatives and eight false positives. The larger number of false positives explains why recall is higher than precision: the judge identifies nearly all human positive cases but occasionally assigns the positive label to human negative cases.

Table 15: Confusion matrix for Gemma-4-31B-it against the majority human label.

	Human 0	**Human 1**
Gemma-4-31B-it: 0	175	3
Gemma-4-31B-it: 1	8	134

Human annotation reliability. The human labels themselves are highly consistent. Human annotators are unanimous on 316 of the 320 instances, corresponding to 98.75% unanimous agreement. Fleiss' κ among the human annotators is 0.983, indicating near complete agreement under a multi annotator reliability measure. When the Gemma-4-31B-it judgment is included as an additional annotator, Fleiss' κ decreases to 0.952. This remains a high level of agreement, but the decrease quantifies the residual disagreement introduced by the automated judge. The results therefore show that the automated judge operates close to the human agreement ceiling while remaining measurably less consistent than the human panel.

Table 16: Multi annotator reliability on the judge validation set.

Measure	**Value**
Human unanimous agreement	$316/320$ (98.75%)
Fleiss' κ among humans	0.983
Fleiss' κ including Gemma-4-31B-it	0.952

Agreement with individual annotators. Agreement with the human majority can conceal variation across individual annotators, so we also compare Gemma-4-31B-it separately with each human annotator. Individual AI human accuracy ranges from 95.94% to 96.56%, and individual Cohen's κ ranges from 0.917 to 0.930. These values are consistently high and remain close to the majority based estimates, suggesting that the result is not caused by agreement with only one particular annotator.

Table 17: Gemma-4-31B-it agreement with individual human annotators.

Metric	**Minimum**	**Maximum**
Accuracy	95.94%	96.56%
Cohen's κ	0.917	0.930

Uncertainty in the agreement estimate. To quantify sampling uncertainty, we bootstrap the 320 validation instances with replacement and recompute AI versus majority human accuracy for each bootstrap sample. The resulting approximate 95% confidence interval is 94.38%–98.44% (Table 18). The interval remains well above chance and supports the conclusion that Gemma-4-31B-it is a reliable automated judge for this binary task. However, the interval alone does not establish equivalence with human performance, because equivalence requires a predefined tolerance margin and a corresponding equivalence test.

Overall, Gemma-4-31B-it achieves 96.56% agreement with the majority human label, with Cohen's $\kappa = 0.930$ and Matthews correlation of 0.931. Its accuracy is approximately 2.61 percentage points below the human human agreement value of 99.17%, while its Cohen's κ is 0.053 lower than the human human value of 0.983. We therefore describe the judge as showing *near human performance* and *high agreement with human judgments*. We do not claim strict equivalence to human annotation, since no equivalence margin was specified before evaluation. These results support using Gemma-4-31B-it for scalable evaluation, provided that its judgments are interpreted as highly reliable automated annotations rather than as a perfect substitute for human assessment.

F.3 Reproducibility and Infrastructure Considerations

The pipeline is built to make its results easy to reproduce and inspect. We release the intermediate outputs from 1–3, including enriched signal files, cluster assignments and centroids, LLM judge prompts and responses, statistical results, and inputs used for narrative synthesis. These files allow each reported numerical result to be traced back to the corresponding analysis step and reproduced without repeating the full pipeline.

Table 18: Bootstrap uncertainty for Gemma-4-31B-it agreement with the majority human label.

Statistic	Value
Validation set size	320
Observed accuracy	96.56%
Approximate 95% bootstrap CI	[94.38%, 98.44%]

Incremental model evaluation. The taxonomy is built once for an initial set of models. Adding a new model does not require rebuilding it. Its failure items are embedded and mapped to the existing clusters using nearest-centroid assignment, after which 3 incorporates the new model into the statistical analysis. This design supports continued evaluation as new models become available while keeping the evaluation procedure fixed.

Cost. LLM calls in 2 and 3 operate on cluster representatives rather than individual benchmark items. Their cost therefore grows more slowly than the benchmark size. For $N = 10{,}000$ items evaluated on 10 models, we estimate a total LLM cost of approximately \$500–600 at current API rates, making the pipeline practical for research groups with limited compute budgets.

F.4 Human Evaluation

We collected human baselines from four levels of mathematical training: **High School** (Grades 11–12), **Undergraduate** (Years 2–3), **Engineering Graduate**, and **Mathematics Graduate**. Three volunteers from each level independently answered every benchmark question using the same general instructions provided to the models. We aggregated the three responses for each question using majority agreement, reducing the influence of individual annotator bias and producing level specific human baselines.

G Model Performance Breakdown on DeepVisualMath

G.1 Model Complementarity

Figure 6 reports the percentage of questions solved by at least one model in each pair, thereby measuring complementary coverage rather than agreement. The strongest pair is Gemini 3.5 Flash with Gemini 3.1 Flash Lite, which jointly solves **66.1%** of the questions, followed by Gemini 3.5 Flash with GPT-5.4 Mini (**64.3%**) and Claude Sonnet 4.6 with Gemini 3.5 Flash (**63.4%**). These values show that combining independently strong models can substantially expand coverage beyond the accuracy of either model alone. The consistently high coverage among Claude, Gemini, and GPT models suggests that their remaining errors are only partially overlapping, indicating useful heterogeneity in their reasoning behaviour.

Complementarity decreases sharply among the weaker models. The pairwise coverage of Kimi K2.5, Qwen3.6 Plus, and Qwen3.7 Flash ranges from only **15.5%** to **26.7%**, suggesting that these models often fail on the same questions rather than providing distinct solutions. Strong models paired with weaker models still achieve moderate coverage, typically around **46%–57%**, but the improvement is smaller than that obtained by pairing strong models with one another. Overall, the heatmap indicates that model ensembles should be selected according to error diversity, not model strength alone, and that combining complementary frontier models may provide more reliable coverage of difficult visual mathematical problems.

G.2 Model Performance On Off topic Rate

As per Table 19, off topic responses are rare across the benchmark, with overall rates below **0.25%** for every model except Gemini 3.5 Flash. Claude Sonnet 4.6, GLM-4.5V, and Kimi K2.5 achieve near perfect topical relevance, with GLM-4.5V and Kimi recording **0.000%** across all categories. Most other models also remain highly focused, with overall rates of **0.031%** for Gemini 2.5 Flash, **0.039%** for Gemini 3.1 Flash Lite, and **0.058%** for GPT-5.4 Mini. Gemini 3.5 Flash is a clear outlier, reaching **0.729%** overall and exceeding **0.7%** in Cross Dimensional, Discrete, Solid, Applied, and Socio Economic Mathematics. Among the open source models, Qwen3.6 Plus and Qwen3.7 Flash show slightly higher off topic rates in Applied Sciences, at **0.957%** and **1.435%**,

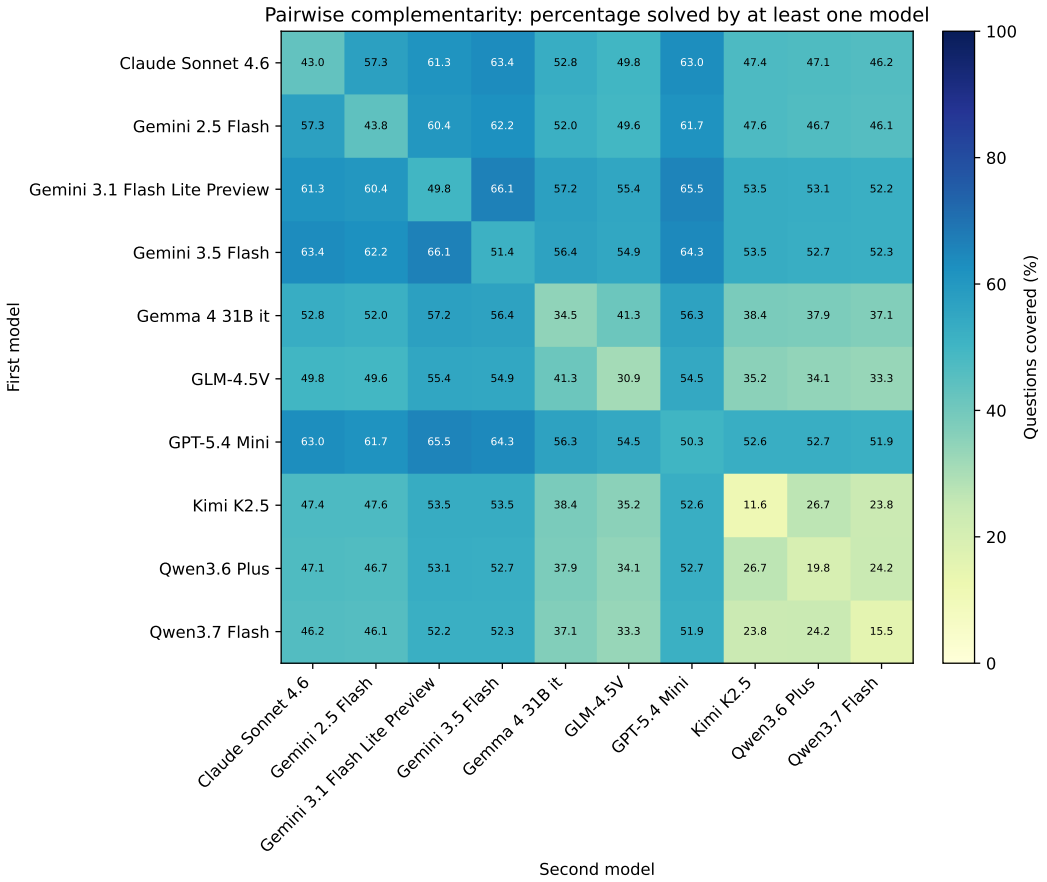


Figure 6: **Pairwise model complementarity.** Each cell reports the percentage of questions solved by at least one of the two models in the corresponding row and column. Higher values indicate greater joint coverage and therefore more complementary error patterns. Strong frontier models, particularly Gemini 3.1 Flash Lite, Gemini 3.5 Flash, and GPT-5.4 Mini, provide the highest pairwise coverage, whereas pairs of weaker open models exhibit substantially lower joint coverage and more overlapping failures.

respectively. Overall, topical irrelevance is not a dominant failure mode, although the category specific spike for Gemini 3.5 Flash suggests occasional instability in response targeting.

Table 19: Off-topic rate (%)(↓) across mathematical reasoning categories.

Model	Cross-Dim. (#3322)	Discrete (#1842)	Solid-3D (#1565)	Continuous (#1398)	Applied (#836)	Plane-2D (#812)	Socio-Econ. (#523)	Overall (#10298)
Proprietary Models								
Claude Sonnet 4.6	0.000	0.000	0.000	0.000	0.120	0.000	0.000	0.020
Gemini 2.5 Flash	0.000	0.054	0.000	0.143	0.000	0.000	0.000	0.031
Gemini 3.1 Flash Lite Preview	0.000	0.054	0.000	0.000	0.239	0.000	0.000	0.039
Gemini 3.5 Flash	0.783	0.814	0.895	0.286	0.718	0.493	0.956	0.729
GPT-5.4 mini	0.090	0.000	0.064	0.072	0.000	0.000	0.000	0.058
Open-Source Models								
Gemma 4 31B it	0.271	0.163	0.320	0.143	0.359	0.123	0.382	0.253
GLM-4.5V	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Kimi K2.5	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
Qwen3.6 Plus	0.000	0.109	0.000	0.215	0.957	0.123	0.000	0.171
Qwen3.7 Flash	0.000	0.000	0.000	0.072	1.435	0.000	0.000	0.129

G.3 Model Performance On Refusal Rate

As per Table 20, refusal rates are also generally very low, remaining below **0.60%** overall for all models. Kimi K2.5 records no refusals in any category, while Qwen3.7 Flash has the lowest nonzero overall rate at **0.010%**. Claude Sonnet 4.6, Qwen3.6 Plus, and GLM-4.5V also show minimal refusal behaviour, with overall rates of **0.049%**, **0.049%**, and **0.059%**, respectively. Gemini 2.5 Flash exhibits the highest overall refusal rate at **0.594%**, driven mainly by Applied Sciences (**3.597%**) and Socio Economic Mathematics (**2.103%**). The remaining models show occasional category specific refusals, particularly on Applied Sciences and Socio Economic Mathematics. Overall, refusals are too infrequent to explain major performance differences, suggesting that errors primarily arise from reasoning and visual interpretation rather than response avoidance.

Table 20: Refusal response rate (%)(↓) across mathematical reasoning categories.

Model	Cross-Dim. (#3322)	Discrete (#1842)	Solid-3D (#1565)	Continuous (#1398)	Applied (#836)	Plane-2D (#812)	Socio-Econ. (#523)	Overall (#10298)
Proprietary Models								
Claude Sonnet 4.6	0.030	0.109	0.064	0.000	0.120	0.000	0.000	0.049
Gemini 2.5 Flash	0.211	0.218	0.449	0.072	3.597	0.124	2.103	0.594
Gemini 3.1 Flash Lite Preview	0.060	0.163	0.129	0.143	0.838	0.000	0.577	0.185
Gemini 3.5 Flash	0.060	0.271	0.256	0.000	0.120	0.000	0.191	0.126
GPT-5.4 mini	0.030	0.109	0.000	0.143	0.479	0.493	0.956	0.175
Open-Source Models								
Gemma 4 31B it	0.000	0.000	0.000	0.286	0.239	0.000	0.956	0.107
GLM-4.5V	0.000	0.055	0.000	0.000	0.605	0.000	0.000	0.059
Kimi K2.5	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
Qwen3.6 Plus	0.000	0.000	0.000	0.000	0.600	0.000	0.000	0.049
Qwen3.7 Flash	0.000	0.000	0.000	0.000	0.120	0.000	0.000	0.010

H CATEGORY-WISE MODEL PERFORMANCE OF DEEPVISUALMATH

H.1 CLOSED-SOURCE MODELS

Table 21: Open-ended evaluation accuracy on Discrete Mathematics & Formal Logic, by sub-category and task, for closed-source models. Best score per row is bold and underlined.

Sub-Category	Task	Claude Sonnet 4.6	Gemini 2.5 Flash	Gemini 3.1 Flash Lite Preview	Gemini 3.5 Flash	GPT-5.4 mini
Abstract Logic & Procedural Puzzles	Logic	**48.65**	22.08	32.05	42.34	47.75
	Puzzle	1.52	0.00	0.00	**4.55**	1.56
	Logical Reasoning	38.00	24.00	**44.00**	**44.00**	38.00
	Game	6.42	6.49	2.60	5.50	**11.93**
	Block Moving	34.00	20.00	**38.00**	8.00	30.00
	Puzzle Test	**94.00**	72.00	58.00	86.00	72.00
	Arrow Moving	38.00	28.00	**56.00**	42.00	36.00
Arithmetic & Concrete Number Theory	Arithmetic	79.09	60.25	68.75	85.99	**86.85**
	Math – Number Theory	60.00	38.00	40.00	54.00	**70.00**
	Divisibility / Factorization	40.82	14.89	23.40	**44.90**	29.79
	Arithmetic Reasoning	82.00	60.00	74.00	**86.00**	74.00
Combinatorics & Graph Structures	Graph Theory	**68.66**	35.64	44.15	66.67	65.43
	Combinatorics	55.37	47.60	36.40	60.54	**71.69**
	Graphs	36.00	**56.00**	52.00	52.00	44.00
	Counting	39.90	15.83	25.83	**54.19**	42.86
	Combinatorial Geometry	49.10	26.23	33.68	42.58	**54.93**
	Math – Combinatorics	**64.00**	22.00	38.00	50.00	46.00

Table 22: Open-ended evaluation accuracy on Plane (2D) Geometry, by sub-category and task, for closed-source models. Best score per row is bold and underlined.

Sub-Category	Task	Claude Sonnet 4.6	Gemini 2.5 Flash	Gemini 3.1 Flash Lite Preview	Gemini 3.5 Flash	GPT-5.4 mini
Plane Coordinate & Transformation	2D-Geometry	19.68	19.47	27.43	**38.30**	29.41
	2D Rotation	31.37	30.00	14.00	58.82	**70.59**
	Plane Geometry – Analytic	86.00	90.00	92.00	**94.00**	**94.00**
Plane Measurement	Plane Geometry – Applied	80.33	70.91	85.19	**90.16**	80.33
	Plane Geometry – Angle	66.67	55.74	73.55	**85.51**	80.19
	Plane Geometry	42.73	13.55	19.27	**58.18**	50.28
	Plane Geometry – Area	76.00	76.00	74.00	**86.00**	74.00
	Plane Geometry – Length	81.88	62.11	67.37	**87.92**	79.19

Table 23: Open-ended evaluation accuracy on Solid (3D) Geometry, by sub-category and task, for closed-source models. Best score per row is bold and underlined.

Sub-Category	Task	Claude Sonnet 4.6	Gemini 2.5 Flash	Gemini 3.1 Flash Lite Preview	Gemini 3.5 Flash	GPT-5.4 mini
Solid Measurement	Solid Geometry	62.95	29.16	34.97	66.92	**68.76**
	Solid Geometry – Area	58.00	32.00	50.00	**72.00**	62.00
	Solid Geometry – Volume	72.00	61.22	65.31	**82.00**	74.00
	Solid Geometry – Length	**70.00**	60.00	62.00	**70.00**	66.00
Solid Visualization & Projection	3D-Geometry	**26.01**	16.13	22.04	18.21	22.32
	Chart Projection	95.05	79.52	80.72	**97.36**	94.72
	Descriptive Geometry	33.33	29.55	30.30	27.19	**48.25**
	CrossSection	25.00	12.33	17.81	**34.00**	28.00
	3D Rotation	**37.50**	25.00	27.08	8.33	33.33
	CubeCounting	38.00	48.98	42.86	36.00	**58.00**

H.2 OPEN-SOURCE MODELS

Table 24: Open-ended evaluation accuracy on Cross-Dimensional & Relational Geometry, by sub-category and task, for closed-source models. Best score per row is bold and underlined.

Sub-Category	Task	Claude Sonnet 4.6	Gemini 2.5 Flash	Gemini 3.1 Flash Lite Preview	Gemini 3.5 Flash	GPT-5.4 mini
Analytic & Transformation	Analytic Geometry	68.72	48.96	57.07	**80.45**	79.61
	Transformation Geometry	**48.00**	21.61	38.19	45.07	46.93
	Functions – Coordinate	**84.00**	82.00	**84.00**	82.00	**84.00**
Comparative Relations	Angle Comparison	80.00	71.09	80.57	88.00	**90.50**
	Perpendicular Comparison	61.46	27.82	11.28	**93.83**	79.00
	Parallel Comparison	90.78	84.58	85.02	89.17	**94.47**
Foundational & Topological Geometry	Geometry	71.30	55.69	47.81	**85.61**	81.27
	Math – Geometry	69.47	38.57	57.97	68.42	**74.47**
	Topology	21.57	42.00	**50.00**	45.10	19.61
Metric Measurement	Metric Geometry – Length	74.75	50.53	62.57	80.59	**80.76**
	Metric Geometry – Angle	58.73	22.47	56.39	**65.99**	63.84
	Metric Geometry – Area	76.04	44.91	58.16	**76.53**	76.48

Table 25: Open-ended evaluation accuracy on Continuous Mathematics & Analysis, by sub-category and task, for closed-source models. Best score per row is bold and underlined.

Sub-Category	Task	Claude Sonnet 4.6	Gemini 2.5 Flash	Gemini 3.1 Flash Lite Preview	Gemini 3.5 Flash	GPT-5.4 mini
Calculus & Advanced Analysis	Calculus	74.81	69.77	71.26	**82.44**	77.86
	Differential Calculus	38.78	42.86	53.06	63.27	**75.51**
	Integral Calculus	44.68	38.30	44.68	51.06	**65.96**
	Analysis and Calculus	90.00	82.00	92.00	**96.00**	86.00
Foundations of Algebra	Algebra	79.73	49.93	61.25	**81.01**	80.43
	Geometry and Algebraic Reasoning	83.43	77.14	86.67	**92.31**	**92.31**
	Algebraic Expressions	30.00	6.52	30.43	42.00	**43.48**
	Equations and Inequalities	30.43	11.11	27.27	**32.61**	22.22
	Math – Algebra	**72.00**	44.00	48.00	70.00	60.00
	Algebraic Reasoning	88.00	86.00	86.00	**90.00**	86.00
Functional Analysis & Precalculus	Functions – Property	78.00	78.00	**86.00**	84.00	74.00
	Functions – Expression	64.00	73.47	72.00	80.00	**86.00**
	Functions – Applied	76.00	77.55	**78.00**	76.00	70.00
Probability, Statistics & Analytics	Statistical	52.08	81.25	54.17	62.50	**83.33**

Table 26: Open-ended evaluation accuracy on Socio-Economic & Contextual Mathematics, by sub-category and task, for closed-source models. Best score per row is bold and underlined.

Sub-Category	Task	Claude Sonnet 4.6	Gemini 2.5 Flash	Gemini 3.1 Flash Lite Preview	Gemini 3.5 Flash	GPT-5.4 mini
Arithmetic & Concrete Number Theory	Numeric Commonsense and Arithmetic Reasoning	48.00	46.00	54.00	**56.00**	32.00
Operational & Financial Analytics	Investment	78.95	66.13	82.26	**86.84**	**86.84**
	Managerial Accounting	**85.96**	67.92	71.70	84.21	73.68
	Operations Research	38.00	48.98	**62.50**	60.00	55.10
	Financial Accounting	75.68	52.46	57.38	**82.43**	79.73
	Economics	**44.00**	10.20	31.91	**44.00**	42.86
	Cost Management	64.00	40.00	38.00	**66.00**	60.00
Semantic & Grounded Representation	Mathematical	22.68	11.41	18.79	24.16	**28.30**

Table 27: Open-ended evaluation accuracy on Applied Sciences & Grounded Reasoning, by sub-category and task, for closed-source models. Best score per row is bold and underlined.

Sub-Category	Task	Claude Sonnet 4.6	Gemini 2.5 Flash	Gemini 3.1 Flash Lite Preview	Gemini 3.5 Flash	GPT-5.4 mini
Physical Systems Reasoning	Physics – Mechanics	10.00	10.00	20.00	22.00	**32.00**
	Physics – Electromagnetism	18.00	12.00	**20.00**	16.00	**20.00**
	Physics – Thermodynamics	24.49	12.24	20.41	**26.53**	20.41
	Physics	18.30	7.75	10.85	**25.00**	21.27
	Physics – Optics	18.00	10.87	13.33	**26.00**	23.91
Probability, Statistics & Analytics	Statistics	82.25	70.64	69.86	**88.49**	87.53
	Scientific Reasoning	88.00	78.00	82.00	86.00	**92.00**
	Arithmetic and Statistical Reasoning	88.00	96.00	92.00	**98.00**	94.00
	Statistical Reasoning	84.34	73.85	76.92	**86.75**	84.34
	Scientific Figure	80.00	70.51	60.26	85.45	**87.27**
	Probability and Statistics	**90.00**	80.00	86.00	**90.00**	88.00

Table 28: Open-ended evaluation accuracy on Discrete Mathematics & Formal Logic, by sub-category and task. Best score per row is bold and underlined.

Sub-Category	Task	Gemma-4-31B-IT	GLM-4.5V	Kimi-K2.5	Qwen3.6-Plus	Qwen3.7-Flash
Abstract Logic & Procedural Puzzles	Logic	20.51	14.10	**61.54**	3.85	5.13
	Puzzle	0.00	0.00	**1.82**	0.00	0.00
	Logical Reasoning	18.00	12.24	**24.00**	2.00	0.00
	Game	0.00	**1.33**	1.30	0.00	0.00
	Block Moving	0.00	0.00	**14.29**	0.00	0.00
	Puzzle Test	54.00	66.00	**84.00**	44.00	58.00
	Arrow Moving	12.00	2.13	**52.00**	0.00	2.00
Arithmetic & Concrete Number Theory	Arithmetic	60.83	52.54	**76.67**	36.67	31.67
	Math – Number Theory	42.00	36.00	**86.00**	26.00	22.00
	Divisibility / Factorization	6.38	10.87	**40.00**	6.38	14.89
	Arithmetic Reasoning	72.00	88.00	**91.84**	60.00	36.00
Combinatorics & Graph Structures	Graph Theory	18.09	15.96	**45.21**	7.45	6.38
	Combinatorics	28.40	21.95	**58.00**	11.60	13.20
	Graphs	48.00	**56.00**	50.00	42.00	30.00
	Counting	13.33	6.67	**27.50**	5.00	0.83
	Combinatorial Geometry	2.07	0.78	**27.98**	0.26	0.00
	Math – Combinatorics	18.00	14.29	**36.73**	14.00	6.00

Table 29: Open-ended evaluation accuracy on Plane (2D) Geometry, by sub-category and task. Best score per row is bold and underlined.

Sub-Category	Task	Gemma-4-31B-IT	GLM-4.5V	Kimi-K2.5	Qwen3.6-Plus	Qwen3.7-Flash
Plane Coordinate & Transformation	2D-Geometry	0.88	1.77	**16.81**	0.88	0.00
	2D Rotation	4.00	6.38	**48.00**	10.00	4.00
	Plane Geometry – Analytic	80.00	79.59	**86.00**	66.00	64.00
Plane Measurement	Plane Geometry – Applied	54.55	57.41	**83.64**	36.36	23.64
	Plane Geometry – Angle	39.34	26.27	**60.33**	16.39	5.74
	Plane Geometry	11.55	8.36	**23.36**	3.61	2.53
	Plane Geometry – Area	70.00	72.00	**78.00**	46.00	44.00
	Plane Geometry – Length	51.58	56.38	**74.19**	37.89	26.32

Table 30: Open-ended evaluation accuracy on Solid (3D) Geometry, by sub-category and task. Best score per row is bold and underlined.

Sub-Category	Task	Gemma-4-31B-IT	GLM-4.5V	Kimi-K2.5	Qwen3.6-Plus	Qwen3.7-Flash
Solid Measurement	Solid Geometry	21.08	13.74	**39.34**	7.11	6.45
	Solid Geometry – Area	56.00	52.00	**72.92**	32.00	16.00
	Solid Geometry – Volume	66.00	68.00	**86.00**	42.00	34.00
	Solid Geometry – Length	44.00	34.69	**65.31**	26.00	20.00
Solid Visualization & Projection	3D-Geometry	1.61	0.00	**13.98**	0.54	0.00
	Chart Projection	81.33	**85.45**	82.53	78.31	72.89
	Descriptive Geometry	1.52	3.05	**21.21**	0.00	0.00
	CrossSection	6.85	2.78	**19.18**	2.74	0.00
	3D Rotation	4.17	0.00	**8.33**	2.08	0.00
	CubeCounting	10.00	8.00	**38.78**	2.00	0.00

Table 31: Open-ended evaluation accuracy on Cross-Dimensional & Relational Geometry, by sub-category and task. Best score per row is bold and underlined.

Sub-Category	Task	Gemma-4-31B-IT	GLM-4.5V	Kimi-K2.5	Qwen3.6-Plus	Qwen3.7-Flash
Analytic & Transformation	Analytic Geometry	41.67	32.81	**50.52**	22.40	18.75
	Transformation Geometry	5.53	2.51	**35.18**	0.50	0.00
	Functions – Coordinate	**78.00**	73.47	**78.00**	60.00	46.00
Comparative Relations	Angle Comparison	**87.20**	81.52	62.09	23.33	1.90
	Perpendicular Comparison	25.19	22.18	**56.39**	8.27	3.01
	Parallel Comparison	84.14	**86.34**	85.46	59.47	35.68
Foundational & Topological Geometry	Geometry	45.45	43.03	**59.19**	31.29	26.92
	Math – Geometry	34.29	37.68	**64.29**	20.00	20.00
	Topology	20.00	2.00	**44.00**	4.00	0.00
Metric Measurement	Metric Geometry – Length	34.27	27.66	**62.85**	17.61	13.18
	Metric Geometry – Angle	17.11	6.64	**44.25**	8.33	7.02
	Metric Geometry – Area	25.00	21.99	**63.27**	13.23	11.77

Table 32: Open-ended evaluation accuracy on Continuous Mathematics & Analysis, by sub-category and task. Best score per row is bold and underlined.

Sub-Category	Task	Gemma-4-31B-IT	GLM-4.5V	Kimi-K2.5	Qwen3.6-Plus	Qwen3.7-Flash
Calculus & Advanced Analysis	Calculus	41.38	34.88	**47.13**	17.24	17.24
	Differential Calculus	**44.90**	32.65	34.69	36.73	16.33
	Integral Calculus	29.79	28.26	**45.65**	10.64	12.77
	Analysis and Calculus	70.00	**75.00**	46.00	12.00	0.00
Foundations of Algebra	Algebra	46.80	41.23	**65.72**	22.54	22.35
	Geometry and Algebraic Reasoning	60.00	60.00	**78.85**	41.90	44.76
	Algebraic Expressions	17.39	13.04	**40.00**	2.17	2.17
	Equations and Inequalities	15.56	6.67	**44.44**	11.11	6.67
	Math – Algebra	36.00	32.65	**64.00**	30.00	28.00
	Algebraic Reasoning	**82.00**	80.00	74.00	72.00	36.00
Functional Analysis & Precalculus	Functions – Property	80.00	**83.67**	44.90	50.00	30.00
	Functions – Expression	60.00	40.82	**72.00**	28.00	24.00
	Functions – Applied	66.00	62.00	**81.63**	40.00	18.00
Probability, Statistics & Analytics	Statistical	27.08	62.50	**75.00**	37.50	27.08

Table 33: Open-ended evaluation accuracy on Applied Sciences & Grounded Reasoning, by sub-category and task. Best score per row is bold and underlined.

Sub-Category	Task	Gemma-4-31B-IT	GLM-4.5V	Kimi-K2.5	Qwen3.6-Plus	Qwen3.7-Flash
Physical Systems Reasoning	Physics – Mechanics	8.00	4.08	**32.00**	2.00	4.00
	Physics – Electromagnetism	10.00	8.00	**24.00**	2.00	0.00
	Physics – Thermodynamics	4.08	6.25	**16.33**	0.00	0.00
	Physics	10.85	6.20	**16.41**	2.33	3.88
	Physics – Optics	17.39	8.89	**32.61**	4.35	4.35
Probability, Statistics & Analytics	Statistics	65.75	65.44	**77.98**	39.27	35.62
	Scientific Reasoning	70.00	68.00	**74.00**	56.00	40.00
	Arithmetic and Statistical Reasoning	94.00	**95.83**	94.00	90.00	86.00
	Statistical Reasoning	75.38	79.69	**84.62**	60.00	40.00
	Scientific Figure	75.64	59.74	**79.49**	53.85	28.21
	Probability and Statistics	70.00	65.31	**89.80**	40.00	22.00

Table 34: Open-ended evaluation accuracy on Socio-Economic & Contextual Mathematics, by sub-category and task. Best score per row is bold and underlined.

Sub-Category	Task	Gemma-4-31B-IT	GLM-4.5V	Kimi-K2.5	Qwen3.6-Plus	Qwen3.7-Flash
Arithmetic & Concrete Number Theory	Numeric Commonsense and Arithmetic Reasoning	32.00	**44.00**	33.33	30.00	16.00
Operational & Financial Analytics	Investment	45.16	41.94	**69.35**	30.65	16.13
	Managerial Accounting	49.06	52.83	**64.15**	32.08	33.96
	Operations Research	26.53	14.58	**57.14**	4.08	2.04
	Financial Accounting	32.79	42.62	**47.54**	21.31	16.39
	Economics	**30.61**	21.28	24.49	6.12	0.00
	Cost Management	54.00	48.98	**66.00**	32.00	28.00
Semantic & Grounded Representation	Mathematical	0.67	1.35	**16.78**	0.67	0.00

I AGENT-BASED EVALUATION

I.1 MATHSOLVER: MULTI-AGENT SYSTEM FOR VISUAL MATHEMATICAL REASONING

We developed MATHSOLVER, a multi agent system for structured visual mathematical reasoning, based on Xolver Hosain et al. (2025). The system decomposes each problem into specialized stages that separately process visual content, mathematical objects, symbolic relations, quantitative constraints, and solution procedures. This design reduces the burden on a single model to simultaneously perceive a diagram, translate it into mathematical language, select an appropriate method, and produce a verifiable answer. The resulting architecture supports both multiple choice and open ended questions across geometry, algebra, calculus, discrete mathematics, applied sciences, and contextual mathematical reasoning. The complete agent prompts are provided in Section K.1.

I.1.1 AGENTS

Visual Mathematical Analysis Agent. The Visual Mathematical Analysis agent initiates the pipeline by producing a detailed description of the input image. It identifies relevant diagrams, geometric objects, coordinate axes, labels, symbols, numbers, equations, tables, graphs, units, annotations, and visual conventions. For geometric figures, the agent records shape type, orientation, relative position, parallelism, perpendicularity, incidence, intersection, symmetry, and visible measurements. For plots and scientific illustrations, it identifies axes, scales, trends, extrema, legends, curves, and marked points. The prompt requires exhaustive observation without solving the problem or omitting apparently minor details. The image and question are provided to a vision language model, and the response is converted into a structured textual representation for downstream processing. This stage establishes a perceptual record while separating visual evidence from mathematical interpretation.

Mathematical Entity Extraction Agent. The Mathematical Entity Extraction agent converts the visual description and question into a discrete inventory of mathematical entities. These entities include points, line segments, rays, angles, polygons, circles, solids, vectors, functions, variables, constants, equations, inequalities, sets, sequences, graphs, tables, physical quantities, and answer choices. Each entity receives a stable identifier, such as `PointA`, `TriangleABC`, `FunctionF`, or `VariableX`. The agent also records aliases and textual references so that the same entity can be tracked consistently across the image, question, intermediate derivation, and final answer. This representation reduces ambiguity when multiple objects share similar visual properties or labels.

Symbol and Measurement Recognition Agent. The Symbol and Measurement Recognition agent extracts numerical and symbolic information from the image. It records exact values, approximate values, units, signs, subscripts, superscripts, fractions, exponents, angle marks, congruence indicators, equality symbols, inequality symbols, and diagram annotations. Each extracted item is associated with its corresponding mathematical entity and its location or visual source. The agent distinguishes between explicitly provided values and quantities that must be inferred. Unclear or partially visible symbols are marked with an uncertainty flag rather than silently replaced. The resulting representation provides a machine readable account of the quantitative evidence available in the problem.

Mathematical Relation Agent. The Mathematical Relation agent translates the extracted entities and symbols into formal relations. These include spatial relations such as `left_of`, `inside`, `parallel_to`, and `perpendicular_to`, as well as mathematical relations such as `equal_to`, `greater_than`, `congruent_to`, `similar_to`, `divides`, `adjacent_to`, and `incident_with`. Relations are represented as structured triples or equations, for example `(AB, perpendicular_to, CD)` or `(angle A, equal_to, angle B)`. The agent distinguishes directly observed relations from relations derived from definitions or theorems. Consistency checks are then applied to identify contradictory measurements, impossible configurations, or relations unsupported by the visual evidence.

Problem Type and Strategy Selection Agent. The Problem Type and Strategy Selection agent classifies the problem according to its dominant mathematical structure and proposes one or more valid solution strategies. Possible strategies include angle chasing, similarity, congruence, coordinate geometry, proportional reasoning, algebraic substitution, equation solving, induction, counting, graph analysis, probability, optimization, dimensional analysis, or direct interpretation of a table or

plot. The agent maps the extracted entities and relations to the requirements of each candidate strategy and identifies the assumptions needed for its application. When multiple strategies are possible, it records them as alternative solution paths rather than prematurely committing to one method. This stage helps evaluate whether the selected reasoning procedure is appropriate for the mathematical structure of the problem.

Symbolic Derivation Agent. The Symbolic Derivation agent applies the selected strategy to construct a step by step mathematical solution. It converts visual relations into equations, applies definitions and theorems, performs algebraic transformations, and derives intermediate quantities before computing the final result. Every step is linked to an input fact, a recognized relation, or a standard mathematical rule. The agent is instructed not to introduce unstated assumptions, such as treating a diagram as drawn to scale, unless the problem explicitly supports that assumption. Intermediate expressions, units, domains, and sign conventions are retained to make the derivation auditable. For multiple choice questions, the agent may use the answer choices as a verification aid, but it must first establish the underlying mathematical result.

Quantitative Verification Agent. The Quantitative Verification agent independently checks the numerical and symbolic validity of the proposed derivation. It recomputes arithmetic, simplifies expressions, verifies substitutions, checks units and dimensions, and tests whether the final value satisfies the original constraints. For geometric problems, it verifies angle sums, lengths, areas, volumes, coordinates, and consistency with the stated configuration. For graphs and tables, it checks whether the selected answer agrees with the relevant values, trend, scale, or extremum. When an exact symbolic solution is available, the agent compares it with any numerical approximation and records the expected precision. Disagreement between the derivation and verification stages triggers a revision request rather than being resolved by majority voting alone.

Theorem and Constraint Checking Agent. The Theorem and Constraint Checking agent examines whether the reasoning respects the applicable mathematical assumptions. It checks conditions such as nonzero denominators, valid square root domains, triangle inequalities, parallelism requirements for proportionality, differentiability requirements, probability bounds, graph connectivity assumptions, and unit compatibility. It also determines whether theorems such as Pythagoras, the angle sum property, similarity, congruence, Bayes' rule, or conservation laws have been applied under valid conditions. This agent identifies logically unsupported steps even when the final numerical answer happens to match an answer choice.

Error Diagnosis and Solution Revision Agent. The Error Diagnosis and Solution Revision agent integrates feedback from the verification and constraint checking stages. It categorizes failures as visual misidentification, symbol recognition error, incorrect entity association, relation extraction error, invalid theorem selection, missing case, algebraic error, arithmetic error, unit inconsistency, unsupported assumption, or answer formatting error. When a failure is detected, the agent returns to the earliest affected stage and revises the corresponding representation or derivation. This targeted revision mechanism prevents the system from merely regenerating the same incorrect solution and provides a structured explanation of why a candidate answer was rejected.

Answer Selection and Explanation Agent. The Answer Selection and Explanation agent produces the final response from the verified solution. For multiple choice questions, it selects the option whose value or statement is supported by the derivation and reports the corresponding answer label. For open ended questions, it presents the final quantity, expression, or conclusion together with the essential reasoning steps. The response is checked for consistency with the requested format, including numerical precision, units, notation, and the distinction between an exact answer and an approximation. The agent is explicitly instructed to avoid adding conclusions that are not established by the visual evidence or mathematical derivation.

Unified Mathematical Representation Agent. The Unified Mathematical Representation agent consolidates the outputs of all previous stages into a machine readable representation containing `visual_entities`, `symbols`, `measurements`, `relations`, `constraints`, `problem_type`, `candidate_strategies`, `derivation`, `verification`, `error_diagnosis`, and `final_answer`. Each field preserves the distinction between observed evidence, inferred information, and derived mathematical conclusions. Empty or uncertain fields are retained explicitly rather than silently discarded. This unified representation enables

Table 35: Aggregate correctness and available judge tags. "Valid" is the number of non-missing item scores; E1/E7 are raw stored tags, not mutually exclusive error counts.

Model	Valid	Accuracy	Errors	E1 tags	E7 tags
GPT-5.4 mini	10,298	50.27%	5,121	0	987
Gemini 3.5 Flash	10,298	51.36%	5,009	618	2,189
Gemini 3.1 Flash Lite	10,276	49.95%	5,143	0	1,113
Claude Sonnet 4.6	10,297	43.04%	5,865	182	737
Gemini 2.5 Flash	10,280	43.84%	5,773	430	917
Gemma 4 31B it	10,297	34.50%	6,745	155	1,139
GLM-4.5V	10,213	31.17%	7,030	0	3,653
Qwen3.6 Plus	10,294	19.78%	8,258	0	2,556
Qwen3.7 Flash	10,298	15.50%	8,702	447	844
Kimi K2.5	1,796	66.65%	599	0	799

reproducible evaluation, targeted error analysis, and comparison of failures across categories, subcategories, and fine grained task types in DEEPVISUALMATH.

Open Ended Mathematical Reasoning Agent. The final reasoning agent receives the unified representation together with the original question and answer choices, when available. It solves the problem using the verified entities, relations, constraints, and derivation, while revisiting earlier stages when the available evidence is incomplete or inconsistent. The agent supports both direct answer prediction and structured solution generation. Its output includes the selected answer, the final mathematical expression or value, a concise derivation, and, when applicable, an error or uncertainty flag. This separation between perception, formalization, derivation, verification, and explanation allows MATHSOLVER to evaluate not only whether a model reaches the correct answer, but also which component of visual mathematical reasoning fails.

I.2 CLAUDE CODE SETUP

We evaluated the Claude Code agent using **Claude Sonnet 6**. The agent received the same questions and images as the other systems and operated with the default Claude Code configuration. We did not add task specific prompts, custom tools, or manual intervention. All remaining settings, including the agent instructions and execution behavior, were left unchanged.

I.3 CODEX SETUP

We evaluated the Codex agent using **GPT 5.6 Luna Max**. It processed the same hard case instances under the default Codex configuration and was allowed to use its standard program assisted reasoning workflow. We added no task specific instructions, tools, or manual corrections. All other model and execution settings remained at their default values.

J ERROR ANALYSIS

We analysed the item-level outputs and judge records in the ten model result directories. The judge files contain the binary correctness decision, per-dimension scores, a short rationale, and (when available) an `error_type` tag. We use the binary decisions for quantitative claims and the rationales for qualitative diagnosis. The analysis follows a first-break principle: when a response contains several mistakes, we assign the failure to the earliest point at which the solution ceases to be grounded in the image and the question.

Scope and data quality. The current result files contain 10,298 questions and ten model runs. Most runs cover the full set; `Kimi K2.5` has 1,796 valid scored rows, while several other runs have a small number of missing rows. The stored `error_type` field is not a complete mutually exclusive annotation: it is populated mainly for E1 and E7, and E7 sometimes appears on correct rows. We therefore do not interpret the raw E1/E7 tag counts as a prevalence estimate and do not report invented frequencies for E2–E6. Instead, the seven-category analysis below is a compact qualitative synthesis of the incorrect responses and judge rationales, with the available tag counts reported separately as audit evidence.

Table 36: Seven first-level error categories and their operational meaning.

Code	Name	Operational definition
E1	Perceptual failure	A visual element is missed, misread, or incorrectly measured.
E2	Representation grounding failure	A detected element is assigned the wrong mathematical meaning.
E3	Conceptual/procedural failure	The model selects the wrong theorem, formula, rule, or procedure.
E4	Quantitative computation failure	The procedure is appropriate but arithmetic or algebraic execution fails.
E5	Multi-step reasoning failure	Correct intermediate work is not combined into the correct final conclusion.
E6	Language–mathematics mapping failure	A relation, quantifier, comparison, or operator in the question is misread.
E7	Format/refusal failure	The answer is missing, unusable, truncated, refused, or violates the required format.

High-level pattern. The dominant failure is not a single mathematical theorem. It is the loss of visual or task grounding before the final calculation. This is visible in the judge rationales: many wrong responses contain fluent algebra but operate on an incorrectly read chart value, geometric relation, answer option, or question polarity. Stronger models reduce gross perceptual and formatting failures, but they remain vulnerable to visually ambiguous configurations and long chains of dependent operations. The weakest models additionally show truncation, malformed output, or answers that never commit to the requested choice.

J.1 Taxonomy

J.2 E1: Perceptual failure

E1 is the earliest visual breakdown. The response may be mathematically coherent, but it starts from the wrong pixels, labels, or geometric configuration. We identify five recurring subcategories:

E1.1 Chart and graph value extraction. The model reads the wrong bar, point, curve, axis label, or interval. A representative judge rationale notes that a model correctly added two chart values but compared them with an incorrectly read maximum.

E1.2 Symbol and digit recognition. Small digits, signs, fractions, subscripts, tick labels, and answer-option symbols are confused or omitted. This failure is especially damaging because a single digit error propagates through an otherwise valid derivation.

E1.3 Geometric relation perception. The model misreads parallelism, perpendicularity, angle equality, adjacency, or incidence. The judge records contain direct examples in which the response answers "No" to a relation whose visual ground truth is "Yes."

E1.4 3-D structure and projection. The model identifies the objects but uses the wrong visible face, viewpoint, hidden edge, or projection. These errors are common in solid-visualisation and unfolding questions.

E1.5 Occlusion, clutter, and low-contrast detection. Relevant marks are present but are ignored when the image contains overlapping shapes, dense annotations, weak contrast, or multiple distractors.

Interpretation. E1 errors are not well described as "reasoning mistakes." They indicate that the model is solving a different internal problem from the one shown. The large spread in accuracy across visual categories, and the presence of E1 tags in only some runs, also suggest that perception is sensitive to model resolution and preprocessing rather than only to parameter count.

J.3 E2: Representation grounding failure

E2 begins after the relevant visual content has been detected. The model sees the mark or object but maps it to the wrong mathematical representation. Four subcategories recur:

E2.1 Coordinate and reference-frame grounding. Image coordinates, axes, origin, orientation, or viewpoint are assigned incorrectly; a visually rightward movement is treated as a negative coordinate change, for example.

E2.2 Symbol-to-quantity grounding. A mark is treated as a point rather than a value, a shaded region as a boundary, or a label as a data value rather than an identifier.

E2.3 Entity correspondence. The model associates a label, side, curve, or object with the wrong counterpart. This produces internally consistent calculations over the wrong pair of entities.

E2.4 Unit and scale grounding. The model reads a scale, unit, or normalisation convention incorrectly, such as treating percentages as raw counts or confusing a diagram's visual length with its labelled quantity.

Interpretation. E2 is a cross-modal alignment failure: the image is not absent from the response, but its semantics are unstable. It often appears as a plausible explanation with the wrong referent, making surface fluency a poor indicator of correctness.

J.4 E3: Conceptual or procedural failure

E3 occurs when the model has a usable representation but chooses an invalid mathematical operation. We identify four forms:

E3.1 Wrong theorem or formula. A familiar formula is applied because it looks similar to the task, even though the required conditions do not hold.

E3.2 Wrong case decomposition. In combinatorics, probability, or logic, the model omits a case, double-counts an arrangement, or treats dependent events as independent.

E3.3 Invalid geometric transformation. Rotation, reflection, translation, similarity, or projection is applied with the wrong invariant or direction.

E3.4 Incorrect statistical or functional interpretation. The model uses an unweighted rather than weighted quantity, confuses a rate with a count, or substitutes a local value for a global functional relation.

Interpretation. E3 is the main distinction between visual recognition and mathematical reasoning. A model can identify every relevant element yet fail because it selects a memorised template instead of checking the assumptions of the procedure.

J.5 E4: Quantitative computation failure

E4 is a procedural-execution error: the chosen method is appropriate, but the calculation is not. Four subcategories are useful:

E4.1 Arithmetic slip. Addition, subtraction, multiplication, division, sign handling, or fraction simplification is incorrect.

E4.2 Algebraic manipulation error. Terms are expanded, factored, substituted, or rearranged incorrectly even though the target equation is understood.

E4.3 Counting and enumeration error. The model loses or duplicates an item while counting visible objects, cases, paths, or arrangements.

E4.4 Precision and rounding error. The model truncates too early, misreads a requested tolerance, or selects an answer option using an inconsistent rounding convention.

Interpretation. The judge rationales frequently distinguish E4 from E3: the model's setup is recognisable, but an intermediate number or final arithmetic step is wrong. This distinction matters because calculator-style verification can target E4, whereas more mathematical supervision is needed for E3.

J.6 E5: Multi-step reasoning failure

E5 appears when local steps are reasonable but the chain does not survive to the final answer. We identify four patterns:

E5.1 Intermediate-result propagation. An early wrong value is carried through all later steps without checking whether it remains compatible with the image or constraints.

E5.2 Lost branch or condition. A case restriction, inequality, domain, or geometric condition is dropped after the first transformation.

E5.3 Cross-step inconsistency. The final answer does not follow from the model's own intermediate values, even when those values are individually correct.

E5.4 Premature termination. The model performs the setup or first calculation but stops before comparing candidates, selecting an option, or stating the requested conclusion.

Interpretation. E5 is particularly visible in long combinatorial, geometric, and applied questions. It explains why a response can look competent to a shallow judge: most local steps are present, but the final dependency structure is broken.

J.7 E6: Language-to-mathematics mapping failure

E6 is caused by misinterpreting the question rather than the picture or the calculation. Four subcategories cover the observed forms:

E6.1 Polarity and comparison reversal. "greater than," "less than," "same," "different," or a yes/no relation is reversed.

E6.2 Quantifier and inequality error. "At least," "at most," "exactly," "between," and inclusive endpoints are mapped to the wrong inequality.

E6.3 Operator-word substitution. "Difference," "sum," "product," "ratio," or "change" is translated into another operation.

E6.4 Reference and scope error. A pronoun, modifier, or phrase such as "the remaining," "both," or "only" is attached to the wrong object or subset.

Interpretation. E6 is easy to miss because the resulting mathematics may be flawless for the wrong question. It is therefore important to separate language grounding from perception and computation when diagnosing model behaviour.

J.8 E7: Format or refusal failure

E7 is a usability failure: the model does not return a valid final answer, even when its hidden or partial reasoning may be adequate. Four forms occur:

E7.1 Truncation and incomplete completion. The response ends during a derivation or before the requested choice. Judge rationales explicitly flag responses that stop mid-sentence without a final "Yes" or "No."

E7.2 Refusal or missing-image response. The model claims that it cannot access the image, declines the task, or asks for information already supplied.

E7.3 Format non-compliance. The answer includes explanation when only a literal choice is allowed, omits the required prefix, or returns a value in a form that cannot be parsed.

E7.4 Malformed or contaminated output. The response contains random characters, file-path-like strings, broken markup, or unrelated text. One judge record contains exactly this pattern rather than a mathematical answer.

Interpretation. E7 is disproportionately costly on exact-match tasks. It is also the only failure category that can turn correct internal reasoning into a scored failure without a mathematical error. The raw stored E7 tags are therefore useful for locating output-instability hotspots, but they must not be read as clean failure counts because the current annotation is noisy and sometimes appears on correct rows.

K PROMPTS USED IN DEEPVISUALMATH

DEEPVISUALMATH: Prompt for Answering Question

Answer the question using clear visual and mathematical reasoning.
Question:
`{question}`
Context (Some possible answers):
`{clean_options(choices)}`
Give only the final answer in one sentence.

DEEPVISUALMATH: Prompt for Judging Model

You are an expert evaluation agent for spatial reasoning tasks. Your job is to rigorously evaluate a model's response.
Return ONLY a valid JSON object. Do NOT include any explanation, markdown, or extra text.
========== INPUT ==========
Question:
`{question}`
Ground Truth Answer:
`{ground_truth}`
Model Response:
`{model_output}`
========== INSTRUCTIONS ==========
Follow these rules strictly:

1. **Accuracy Score (0/1):**
   - 1 if the final answer is correct.
   - 0 otherwise.
   - For OE: use semantic equivalence.
2. **Refusal Score (0/1):**
   - 1 if the model refuses (e.g., "I cannot answer", "insufficient information").
   - 0 otherwise.
3. **Off Topic Score (0/1):**
   - 1 if response is irrelevant.
   - 0 if relevant.
4. **Extract Reasoning:**
   - Extract and summarize the reasoning for the Score.

========== OUTPUT FORMAT ==========
Return **STRICT JSON only** (no trailing commas, no Python types, use `null` not `None`):

```
{
  "accuracy_score": 0,
  "refusal_score": 0,
  "off_topic_score": 0,
  "reasoning": ""
}
```

DEEPVISUALMATH: Prompt for Error Cluster Naming

You are an expert in Mathematical error analysis.
Your task is to assign a concise, reusable name to an error cluster based on the provided error examples.
Instructions:

- Examine all examples and identify the dominant shared failure pattern.
- Focus on the underlying root cause, not the specific content of the examples.
- Ignore outliers unless they represent the majority.
- The name should be generic enough to apply to future errors of the same type.
- Use standard ML error analysis terminology when appropriate.
- Keep the name between 3 and 8 words.
- Avoid generic names like "Prediction Error" or "Wrong Answer."

Error Examples:
{ERROR_EXAMPLES}
Output:
<cluster_name_only>

DEEPVISUALMATH: Prompt for Error Taxonomy Classification

You are an expert error taxonomy classifier for spatial reasoning models.
You will be given a set of clustered model failures. Your task is to assign each cluster to exactly ONE primary error type from a fixed taxonomy.
Return ONLY valid JSON. Do NOT include explanation, markdown, or extra text.
========== FIXED ERROR TAXONOMY ==========
{error_taxonomies}
========== INPUT ==========
Cluster ID:
{cluster_id}
Cluster Examples:
{list_of_failure_samples}
========== TASK ==========

1. Carefully analyze the cluster behavior.
2. Identify the SINGLE best matching error type.
3. Do NOT assign multiple labels.
4. Prefer the dominant failure mode in the cluster.

========== OUTPUT FORMAT ==========
Return **STRICT JSON only** (no trailing commas, no Python types, use `null` not `None`):

```
{
  "cluster_id": "{cluster_id}",
  "error_type": "",
  "error_name": "string label of selected type"
}
```

K.1 PROMPTS FOR MATHSOLVER

DEEPVISUALMATH: Prompt for Visual Mathematical Analysis

You are a visual mathematical analysis agent.
Your task is to inspect the image and question carefully and produce an exhaustive description of all visually relevant mathematical evidence. Do not solve the problem.
Return ONLY a valid JSON object. Do not include markdown, explanations, or extra text.
========== INPUT ==========
Question:
{question}
Image:
{image}
========== INSTRUCTIONS ==========
Record all visible information relevant to solving the problem:

1. Identify every diagram, graph, table, chart, equation, annotation, and visual object.
2. Describe geometric objects, including points, lines, rays, segments, angles, polygons, circles, and solids.
3. Record labels, numbers, symbols, units, arrows, tick marks, angle marks, shading, and colour.
4. Describe relative positions, alignment, intersection, containment, symmetry, parallelism, and perpendicularity.
5. For graphs and tables, identify axes, scales, legends, marked points, trends, extrema, and relevant values.
6. Distinguish explicitly visible information from information that is uncertain or inferred.
7. Do not assume that a diagram is drawn to scale unless this is explicitly stated.
8. Do not calculate, interpret, or solve the question.

========== OUTPUT FORMAT ==========

```
{
  "visual_objects": [],
  "labels_and_symbols": [],
  "measurements": [],
  "spatial_properties": [],
  "graph_or_table_information": [],
  "uncertain_observations": []
}
```

DEEPVISUALMATH: Prompt for Mathematical Entity Extraction

You are a mathematical entity extraction agent.
Your task is to convert the visual description and question into a structured inventory of all mathematical entities. Do not solve the problem.
Return ONLY a valid JSON object.
========== INPUT ==========
Question:
{question}
Visual Analysis:
{visual_analysis}
========== INSTRUCTIONS ==========

1. Extract every mathematical object mentioned or visible in the input.
2. Assign a unique and stable identifier to each entity.
3. Include points, lines, segments, rays, angles, polygons, circles, solids, vectors, functions, variables, constants, sets, sequences, graphs, tables, and physical quantities.
4. Preserve labels and record aliases when the same entity has multiple references.
5. Separate explicitly observed entities from inferred entities.
6. Do not combine distinct objects merely because they have similar appearance.
7. Do not introduce entities that are not supported by the image or question.

========== OUTPUT FORMAT ==========

```
{
  "entities": [
    {
      "id": "",
      "type": "",
      "label": "",
      "aliases": [],
      "description": "",
      "source": "observed|inferred",
      "confidence": 0.0
    }
  ]
}
```

DEEPVISUALMATH: Prompt for Symbol and Measurement Recognition

You are a mathematical symbol and measurement recognition agent.
Your task is to extract every numerical and symbolic item from the image and question. Do not solve the problem.
Return ONLY a valid JSON object.
========== INPUT ==========
Question:
{question}
Visual Analysis:
{visual_analysis}
Entities:
{entities}
========== INSTRUCTIONS ==========

1. Extract exact values, approximate values, variables, constants, fractions, exponents, subscripts, signs, units, and equations.
2. Extract visual mathematical notation such as angle marks, congruence marks, tick marks, arrows, equality symbols, and inequality symbols.
3. Associate each value or symbol with its relevant entity whenever possible.
4. Distinguish exact values from approximate values.
5. Distinguish explicitly provided values from quantities that must be inferred.
6. Mark unclear, ambiguous, or partially visible symbols as uncertain.
7. Do not perform calculations or infer unstated numerical values.

========== OUTPUT FORMAT ==========

```
{
  "symbols": [
    {
      "symbol": "",
      "meaning": "",
      "associated_entity": "",
      "value": null,
      "unit": "",
      "exactness": "exact|approximate|unknown",
      "source": "explicit|inferred|uncertain",
      "confidence": 0.0
    }
  ],
  "equations": [],
  "inequalities": [],
  "measurements": []
}
```

DEEPVISUALMATH: Prompt for Mathematical Relation Extraction

You are a mathematical relation extraction agent.
Your task is to convert the extracted entities, symbols, and visual evidence into formal mathematical relations. Do not solve the problem.
Return ONLY a valid JSON object.
========== INPUT ==========
Question:
{question}
Visual Analysis:
{visual_analysis}
Entities:
{entities}
Symbols and Measurements:
{symbols}
========== INSTRUCTIONS ==========

1. Extract spatial, geometric, algebraic, numerical, logical, and contextual relations.
2. Represent each relation using a subject, relation, and object.
3. Include relations such as equality, inequality, parallelism, perpendicularity, incidence, containment, adjacency, similarity, congruence, ordering, divisibility, and functional dependence.
4. Include equations and constraints when they involve multiple entities.
5. Distinguish directly observed relations from mathematically inferred relations.
6. Do not assume unmarked properties such as equal lengths, right angles, or parallel lines.
7. Identify contradictions or relations that cannot be verified from the available evidence.

========== OUTPUT FORMAT ==========

```
{
  "relations": [
    {
      "subject": "",
      "relation": "",
      "object": "",
      "source": "observed|inferred|uncertain",
      "justification": "",
      "confidence": 0.0
    }
  ],
  "equations": [],
  "constraints": [],
  "possible_contradictions": []
}
```

DEEPVISUALMATH: Prompt for Problem Type and Strategy Selection

You are a mathematical problem classification and strategy selection agent.
Your task is to identify the mathematical structure of the problem and propose appropriate solution strategies. Do not complete the final solution.
Return ONLY a valid JSON object.
========== INPUT ==========
Question:
{question}
Entities:
{entities}
Symbols and Measurements:
{symbols}
Relations and Constraints:
{relations}
========== INSTRUCTIONS ==========

1. Classify the problem into one or more mathematical domains.
2. Possible domains include geometry, algebra, calculus, discrete mathematics, probability, statistics, number theory, coordinate geometry, applied science, optimization, graph theory, and contextual mathematics.
3. Identify the main task, such as finding a value, proving a statement, comparing quantities, identifying a pattern, interpreting a graph, or selecting an option.
4. Propose one or more valid solution strategies.
5. For each strategy, list the facts, definitions, theorems, equations, or assumptions required.
6. Rank the strategies by suitability.
7. Do not calculate the final answer.

========== OUTPUT FORMAT ==========

```
{
  "problem_domains": [],
  "task_type": "",
  "known_quantities": [],
  "unknown_quantities": [],
  "candidate_strategies": [
    {
      "name": "",
      "description": "",
      "required_facts": [],
      "required_theorems": [],
      "required_assumptions": [],
      "suitability": 0.0
    }
  ],
  "recommended_strategy": ""
}
```

DEEPVISUALMATH: Prompt for Symbolic Mathematical Derivation

You are a symbolic mathematical derivation agent.
Your task is to solve the problem step by step using only the provided evidence and a valid mathematical strategy.
Return ONLY a valid JSON object.
========== INPUT ==========
Question:
{question}
Entities:
{entities}
Symbols and Measurements:
{symbols}
Relations and Constraints:
{relations}
Recommended Strategy:
{strategy}
Answer Choices:
{answer_choices}
========== INSTRUCTIONS ==========

1. Solve the problem using a logically valid sequence of steps.
2. Link every step to an observed fact, mathematical relation, definition, theorem, or valid algebraic operation.
3. Clearly distinguish known quantities, intermediate quantities, and the final quantity.
4. Do not assume that a diagram is drawn to scale.
5. Do not introduce unstated values or properties.
6. Preserve exact expressions before giving decimal approximations.
7. Track units and signs throughout the derivation.
8. Consider all relevant cases when the problem allows multiple solutions.
9. If the available information is insufficient, state this explicitly.

========== OUTPUT FORMAT ==========

```
{
  "solution_steps": [
    {
      "step": 1,
      "expression": "",
      "operation": "",
      "justification": "",
      "result": ""
    }
  ],
  "intermediate_results": [],
  "candidate_final_answers": [],
  "assumptions_used": [],
  "insufficient_information": false
}
```

DEEPVISUALMATH: Prompt for Quantitative Verification

You are an independent mathematical verification agent.
Your task is to check the proposed derivation and final answer independently. Do not accept the derivation merely because it appears plausible.
Return ONLY a valid JSON object.
========== INPUT ==========
Question:
{question}
Proposed Derivation:
{derivation}
Proposed Answer:
{proposed_answer}
Original Evidence:
{evidence}
========== INSTRUCTIONS ==========

1. Recompute all important arithmetic and algebraic transformations.
2. Verify substitutions and simplifications.
3. Check units, dimensions, signs, domains, and numerical precision.
4. For geometry, verify angle sums, lengths, areas, volumes, coordinates, and stated configurations.
5. For graphs and tables, verify that the answer agrees with the relevant values and scales.
6. Check whether the final answer satisfies the original equations and constraints.
7. Identify the first incorrect step if an error exists.
8. Do not change the answer silently. Report every disagreement.

========== OUTPUT FORMAT ==========

```
{
  "verification_status": "verified|partially_verified|rejected",
  "recomputed_results": [],
  "consistent_steps": [],
  "incorrect_steps": [],
  "unit_check": "passed|failed|not_applicable",
  "constraint_check": "passed|failed|not_applicable",
  "final_answer_check": "passed|failed|uncertain",
  "verification_notes": ""
}
```

DEEPVISUALMATH: Prompt for Theorem and Constraint Checking

You are a mathematical theorem and constraint checking agent.
Your task is to determine whether the proposed reasoning uses valid mathematical assumptions and theorems.
Return ONLY a valid JSON object.
========== INPUT ==========
Question:
{question}
Entities and Relations:
{relations}
Proposed Derivation:
{derivation}
========== INSTRUCTIONS ==========

1. Identify every theorem, definition, formula, and assumption used.
2. Check whether each theorem is applicable under the stated conditions.
3. Check triangle inequalities, angle conditions, domain restrictions, denominator restrictions, and probability bounds when relevant.
4. Check whether proportionality, similarity, congruence, symmetry, or conservation laws are properly justified.
5. Check whether the reasoning treats a visual appearance as a mathematical fact without evidence.
6. Check whether all cases and boundary conditions have been considered.
7. Report unsupported assumptions and invalid theorem applications explicitly.

========== OUTPUT FORMAT ==========

```
{
  "theorems_and_definitions_used": [],
  "valid_applications": [],
  "invalid_applications": [],
  "unsupported_assumptions": [],
  "missing_constraints": [],
  "boundary_cases": [],
  "overall_status": "valid|partially_valid|invalid",
  "explanation": ""
}
```

DEEPVISUALMATH: Prompt for Mathematical Error Diagnosis

You are a mathematical error diagnosis and revision agent.
Your task is to identify the source of an incorrect or uncertain solution and provide a corrected reasoning path.
Return ONLY a valid JSON object.
========== INPUT ==========
Question:
{question}
Evidence:
{evidence}
Derivation:
{derivation}
Verification:
{verification}
Constraint Check:
{constraint_check}
========== INSTRUCTIONS ==========

1. Identify the earliest incorrect or unsupported step.
2. Classify each error using one or more of the following categories:
   - visual misidentification
   - symbol recognition error
   - entity association error
   - relation extraction error
   - invalid theorem selection
   - unsupported assumption
   - missing case
   - algebraic error
   - arithmetic error
   - unit inconsistency
   - answer formatting error
3. Explain why the step is incorrect.
4. Replace the incorrect step with a valid step.
5. Recompute all downstream results affected by the correction.
6. If the problem cannot be solved reliably, state the exact source of uncertainty.

========== OUTPUT FORMAT ==========

```
{
  "error_found": true,
  "error_categories": [],
  "first_incorrect_step": null,
  "error_explanation": "",
  "corrected_steps": [],
  "corrected_final_answer": null,
  "remaining_uncertainty": ""
}
```

DEEPVISUALMATH: Prompt for Unified Mathematical Representation

You are a mathematical representation standardization agent.
Your task is to consolidate all intermediate outputs into one consistent, machine readable representation.
Return ONLY a valid JSON object.
========== INPUT ==========
Visual Analysis:
{visual_analysis}
Entities:
{entities}
Symbols:
{symbols}
Relations:
{relations}
Strategy:
{strategy}
Derivation:
{derivation}
Verification:
{verification}
Error Diagnosis:
{error_diagnosis}
========== INSTRUCTIONS ==========

1. Consolidate all information without losing source attribution.
2. Preserve the distinction between observed, inferred, derived, and uncertain information.
3. Use consistent entity identifiers throughout the representation.
4. Remove duplicated entries while preserving meaningful alternatives.
5. Preserve all relevant equations, constraints, assumptions, derivation steps, and verification results.
6. Do not introduce new mathematical facts.
7. If two agents disagree, retain both claims and record the disagreement explicitly.

========== OUTPUT FORMAT ==========

```
{
  "visual_entities": [],
  "symbols": [],
  "measurements": [],
  "relations": [],
  "equations": [],
  "constraints": [],
  "problem_type": {},
  "candidate_strategies": [],
  "derivation": [],
  "verification": {},
  "error_diagnosis": {},
  "disagreements": [],
  "uncertainties": []
}
```

DEEPVISUALMATH: Prompt for Final Mathematical Reasoning

You are the final mathematical reasoning agent.
Your task is to provide the most accurate answer using the standardized mathematical representation and the original question.
Return ONLY a valid JSON object.
========== INPUT ==========
Question:
{question}
Answer Choices:
{answer_choices}
Unified Representation:
{unified_representation}
========== INSTRUCTIONS ==========

1. Use the visual evidence, extracted entities, equations, relations, constraints, and verified derivation.
2. Select the correct answer for multiple choice questions.
3. Produce the final mathematical value, expression, statement, or classification for open ended questions.
4. Use semantic equivalence when the response is expressed differently from the expected answer.
5. Include only the essential reasoning needed to support the answer.
6. Preserve exact forms when possible and provide approximations only when appropriate.
7. Include units when they are required.
8. Do not rely on unsupported visual assumptions.
9. If the evidence is insufficient or contradictory, report the uncertainty rather than inventing an answer.

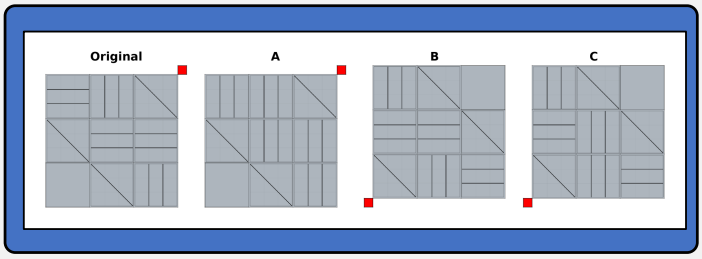
========== OUTPUT FORMAT ==========

```
{
  "final_answer": "",
  "selected_option": null,
  "answer_type": "multiple_choice|open_ended|uncertain",
  "reasoning": "",
  "supporting_steps": [],
  "confidence": 0.0,
  "uncertainty": ""
}
```

L Representative Benchmark Samples Across All Tasks

Here (including supplementary PDF) we include at least one example (with responses from different models).

Plane (2D) Geometry : Plane Coordinate & Transformation : 2D Rotation



Question: The left image shows a patterned grid with a red square marking one corner. Which grid can be obtained by rotating the left grid only? Please answer from options A, B, C, or D.
Possible Choices: ['A', 'B', 'C', 'All three other options are incorrect']

Gold Answer: B

claude_sonnet_4.6: The answer is C

gemini_2.5_flash: Grid B can be obtained by rotating the left grid only

gemini_3.1_flash_lite_preview: The correct answer is B as rotating the original grid 180 degrees moves the red square to the bottomleft corner and correctly reorients all the internal patterns to match option B

gemini_3.5_flash: 31 diagonal TLBR Matches 180 deg 13 was diagonal TLBR rotated 180 deg remains

gemma_4_31b_it: The objective is to identify which of the grids A B or C can be obtained by rotating the Original grid The grid is a 3x3 arrangement of larger squares Each larger square contains internal lines There is a red square marker at the topright corner Topleft Two horizontal lines dividing it into three rectangles Topmiddle Two vertical lines dividing it into three rectangles Topright One diagonal line from topleft to bottomright Middleleft One diagonal line from topleft to bottomright Middlemiddle Two horizontal lines dividing it into three rectangles Middleright Two horizontal lines

glm_4.5v: This question is asking me to identify which of the options A B or C can be obtained by rotating the original grid Let me analyze the original image and each option First lets look at the original The red square is at the top right corner The grid is a 3x3 arrangement of larger cells Let me identify the pattern in each cell Top left horizontal lines 3 rectangles stacked vertically Top middle vertical lines 3 rectangles side by side Top right diagonal line from topleft to bottomright Middle left diagonal line from topleft to bottomright Middle center horizontal lines 3 rectangles stacked vertically Middle right

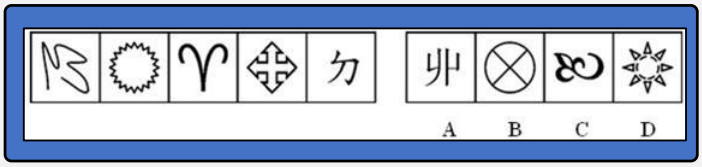
gpt_5.4_mini: B

qwen3.6_plus: The user wants me to identify the correct rotation of the Original grid from the options A B and C 1 Analyze the Original Grid Its a 3x3 grid of squares Lets describe the pattern in each cell row col where 00 is topleft 00 Two horizontal lines 01 Two vertical lines 02 Diagonal line from topleft to bottomright 10 Diagonal line from topleft to bottomright 11 Two horizontal lines 12 Two horizontal lines 20 Blank solid grey 21 Diagonal line from topleft to bottomright 22 Two vertical lines There is a red square at the topright corner This is a crucial marker 2 Analyze the Rotation The question as

qwen3.7_flash: The user wants me to identify which of the grids A B or C is a rotation of the original grid 1 Analyze the Original Grid Lets define the grid as a 3x3 matrix of cells Lets denote the cells by row col where row 0 is top col 0 is left 0 0 Horizontal lines 3 lines Wait lets look closer It has 3 horizontal lines Lets call it Hlines 0 1 Vertical lines 3 lines Lets call it Vlines 0 2 Diagonal line from topleft to bottomright Lets call it D1 1 0 Diagonal line from topleft to bottomright D1 1 1 Horizontal lines Hlines 1 2 Horizontal lines Hlines 2 0 Blank square Lets call it Blank 2 1 Diagonal line from topleft to bottomright D1 2 2 Vertica

Figure 7: Example from the benchmark and evaluation results. Red denotes incorrect and Green denotes correct answer marked by the judge model.

Plane (2D) Geometry : Plane Coordinate & Transformation : 2D-Geometry



Question: Choose the most appropriate option from the given four options to present a certain regularity:

Gold Answer: B

claude_sonnet_4.6: Let me analyze the pattern in the sequence of symbols The sequence shows 1 A wavycurved symbol looks like a cursive letter or wave 2 A spiky starsun shape 3 Aries symbol 4 A cross with arrowscompass rose 5 A partial symbol 勿 or similar Let me think about this differently The symbols might represent something in terms of their visual complexity or number of elements Looking at the symbols 1 First symbol wavy lines appears to have curves 2 Second symbol spiky star many points 3 Third symbol Aries like two curved horns 4 Fourth symbol compasscross with 4 directions 5 Fifth symbol appears to be 勿 Chinese character

gemini_2.5_flash: The pattern is based on the number of strokes or distinct parts in each figure decreasing by one in each step the first figure has 3 strokes the second has 2 the two horns the third has 1 the central cross and the fourth has 0 the single stroke of the character Following this pattern the next figure should have 3 strokes which is represented by option A The final answer is boxedA

gemini_3.1_flash_lite_preview: The correct option is B because the sequence follows a pattern of alternating between symbols with curved lines and symbols with straight lines and the next symbol must be one composed of straight lines

gemini_3.5_flash: The most appropriate option to continue the pattern is B as the sequence alternates between open figures having 0 closed regions and closed figures having 1 or more closed

gemma_4_31b_it: The user wants me to identify the pattern in the first sequence of images and choose the most appropriate option A B C or D to continue or complete the regularity 1 Analyze the first sequence of images Image 1 A stylized shape looks like a character or a squiggle Image 2 A jagged starlike or sunlike shape a circle with many small spikes Image 3 A shape resembling the Aries zodiac symbol Image 4 A diamond shape with a cross inside it Image 5 A stylized character looks like a simplified Chinese character or a symbol

glm_4.5v: This question is asking me to identify a pattern in the first row of images and then choose which option A B C or D continues that pattern Let me analyze the first set of images 1 First image Looks like a stylized B or some curved lines 2 Second image A starburst or sun shape with many points 3 Third image The Aries zodiac symbol 4 Fourth image A geometric shape with fourfold symmetry looks like a square with extensions 5 Fifth image The Chinese character 力 lì meaning power or force

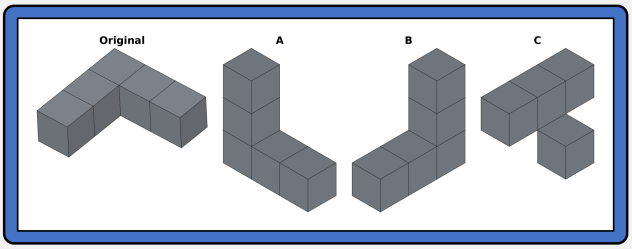
gpt_5.4_mini: The most appropriate option is D

qwen3.6_plus: The user wants me to identify the pattern in the first row of images and select the correct option from the second row to complete the sequence 1 Analyze the first row the sequence Image 1 Looks like a squiggly line or a stylized B or 13 Lets count the number of separate parts or strokes It looks like one continuous shape but lets look for other properties Maybe its about symmetry No obvious symmetry Maybe its about the number of ends or points Image 2 A jagged circle like a sun or a gear It has many points Image 3 Looks like the Greek letter Gamma gamma or a rams horn It has two ends Image 4 A geometric shape with arrows pointing outwar

qwen3.7_flash: The user wants me to identify the pattern in the first row of images and select the correct option from the second row to continue the pattern 1 Analyze the first row Context Image 1 Looks like a stylized 13 or B or Chinese character part Wait lets look at strokes or components It looks like the Chinese character 十三 13 combined or stylized Actually it looks like 13 written cursively Lets look at the number of strokes or parts Image 2 A sunlike shape with many points It looks like a circle with jagged edges Image 3 Looks like the Aries zodiac symbol Upsilon Or the Chinese character 羊 sheepgoat top part Wait Aries is the 3rd sign Image 4 A

Figure 8: Example from the benchmark and evaluation results. Red denotes incorrect and Green denotes correct answer marked by the judge model.

Solid (3D) Geometry : Solid Visualization & Projection : 3D Rotation



Question: The left image shows the original cube stack made of equal-sized small cubes. Which of the options on the right cannot be obtained by rotating the original cube stack?
Possible Choices: ['A', 'B', 'C', 'All three other options are incorrect']

Gold Answer: C

claude_sonnet_4.6: The answer is C

gemini_2.5_flash: Option C cannot be obtained by rotating the original cube stack because the original stack has a cube stacked vertically on one of the corner cubes while option C has a cube extending horizontally from the middle of a threecube line

gemini_3.1_flash_lite_preview: The correct answer is A

gemini_3.5_flash: Lets count the cubes in C Topleft to bottomright 3 cubes From the middle cube of this row there is a cube extending to the

gemma_4_31b_it: The goal is to identify which of the options A B or C cannot be obtained by rotating the original cube stack Its an Lshaped arrangement of small cubes Lets count the cubes One arm has 3 cubes The other arm has 3 cubes Wait lets look closer The corner cube is shared One arm left to right has 3 cubes The other arm back to front has 3 cubes Total cubes 3 3 1 5 cubes Wait let me reexamine the image Original Left arm 3 cubes including corner Right arm 3 cubes including corner Total 5 cubes Wait look

glm_4.5v: Let me analyze the original cube stack and the three options Original Cube Stack Looking at the original image I can see its a shape made of 6 small cubes Let me count and identify the structure It looks like an L shape or a bent line Let me trace it there are cubes arranged in a way that forms a corner From the perspective shown I can see 3 cubes in one direction going backleft then it turns and there are 3 more cubes in another direction going right Actually looking more carefully Theres a front cube on the left Behind it is another cube Behind that is a third cube To the right of the third cube is a fourth cube

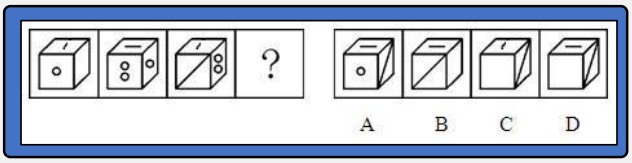
gpt_5.4_mini: C

qwen3.6_plus: The user wants me to identify which of the options A B or C cannot be obtained by rotating the Original cube stack 1 Analyze the Original Object Its an Lshaped structure made of cubes Lets count the cubes One arm has 3 cubes in a row The other arm has 3 cubes in a row They share a corner cube Wait lets look closer Left arm 3 cubes long Right arm 3 cubes long They meet at a corner Total cubes It looks like a flat Lshape lying on the ground Lets count carefully Left side 1 2 3 cubes Right side 1 2 3 cubes The corner cube is shared So total cubes 3 3 1 5 cubes Wait lets reex

qwen3.7_flash: The user wants me to identify which of the options A B C cannot be obtained by rotating the original cube stack 1 Analyze the Original Object Its an Lshaped structure made of cubes Lets count the cubes One arm has 3 cubes including the corner Wait lets look closer Left arm 3 cubes in a row Lets call them 00 10 20 Right arm 3 cubes in a row Lets call them 00 01 02 Wait lets trace the path Start at the far left end cube 1 Next is cube 2 Next is the corner cube 3 Then cube 4 Then cube 5 So its a long Lshape Lets count Left branch 3 cubes Right branch 3 cubes They share the corner cube Total cubes 3 3 1 5 cubes

Figure 9: Example from the benchmark and evaluation results. Red denotes incorrect and Green denotes correct answer marked by the judge model.

Solid (3D) Geometry : Solid Visualization & Projection : 3D-Geometry



Question: Which of the following options conforms to the given pattern of changes in the figures?

Gold Answer: D

claude_sonnet_4.6: The answer is C

gemini_2.5_flash: The pattern shows that the top face alternates between a single dash and a diagonal line while the front face alternates between one dot and two dots and the right face alternates between being empty and having a diagonal line Following this pattern the missing figure should have a diagonal line on the top face one dot on the front face and an empty right face which corresponds to option C The final answer is boxedC

gemini_3.1_flash_lite_preview: The correct option is B because the pattern involves the sequential removal of the dots from the front face and then the side face followed by the addition of a diagonal line across the side face which continues the progression of simplifying and modifying the cubes features

gemini_3.5_flash: Based on the pattern of changes in the figures the elements on the front face 1 circle 2 circles diagonal line shift to the right face in the subsequent step while a

gemma_4_31b_it: The user wants me to identify the next figure in a sequence based on a pattern 1 Analyze the sequence of figures Figure 1 Top face A single short horizontal line dash Front face A single small circle Right face Blank Figure 2 Top face A single short horizontal line dash Front face Two small circles stacked vertically Right face Blank Figure 3 Top face A single short horizontal line dash

glm_4.5v: This question is about identifying the pattern in a sequence of dice figures and determining which option fits the pattern Let me analyze the given figures 1 First figure A die with one dot center and a line on top face 2 Second figure A die with five dots like a 5 pattern and no line visible or different orientation 3 Third figure A die with three dots diagonal and a diagonal line dividing the face 4 Fourth figure Question mark we need to find what comes next Let me look more carefully at the pattern It seems like there might be two attributes changing The number of dots The presenceposition of lines Let me list the figur

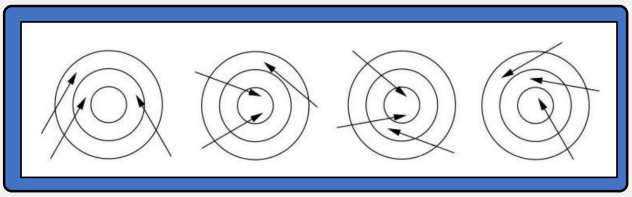
gpt_5.4_mini: Identifying cube patterns Im trying to infer a pattern from the first three cubes In Figure 1 the front has one dot circle the top has a short diagonal mark and the right face is blank Figure 2 shows two circles vertically in front a right face with one dot and the top with a horizontal line In Figure 3 the front has a diagonal line the right face has two circles and the top still has a horizontal line It seems like the front content shifts to the right face with each figure

qwen3.6_plus: The user wants me to identify the next figure in a sequence 1 Analyze the input There are three figures in a sequence followed by a question mark There are four options A B C D The figures are cubes with markings on their faces 2 Examine the sequence Figure 1 Top face A small diagonal line looks like a 1 or a slash Front face A single dot looks like a 1 on a die Right face Blank Figure 2 Top face Two small diagonal lines looks like a 2 or two slashes Front face Three dots arranged vertically looks like a 3 on a die Right face Two dots arranged vertically looks like a 2 on a die Figure 3 Top face A sm

qwen3.7_flash: The user wants me to identify the next figure in a sequence 1 Analyze the sequence Figure 1 A cube Top face A diagonal line looks like a slash from bottomleft to topright or maybe just a line Wait lets look closer It looks like a diagonal line on the top face Actually looking at Figure 3 theres a diagonal line on the front face Lets reevaluate Front face A circle dot in the center Right face Blank Top face A diagonal line Wait lets look at the orientation In Fig 1 top face has a line Front face has a circle Right face is blank Figure 2 A cube Top face Two horizontal lines or a rectangle with lines Actually it look

Figure 10: Example from the benchmark and evaluation results. Red denotes incorrect and Green denotes correct answer marked by the judge model.

Continuous Mathematics & Analysis : Foundations of Algebra : Algebra



Question: Jeffrey shoots three arrows at each of four identical targets. He scores 29 points on the first target, 43 on the second and 47 on the third. How many points does Jeffrey score on the last target?
<image1>
Possible Choices: []

Gold Answer: 36

claude_sonnet_4.6: Let me analyze the targets Each target has three rings outer middle innerbullseye Let me assign point values to each ring Looking at a typical archery target with 3 rings lets say the values are Outer ring a points Middle ring b points Inner ring bullseye c points

gemini_2.5_flash: Jeffrey scores 39 points on the last target

gemini_3.1_flash_lite_preview: Based on the visual arrangement of the arrows let x y and z represent the point values for the outer middle and inner rings respectively 1 Target 1 29 points Two arrows in the outer ring and one in the middle ring 2x y 29 2 Target 2 43 points One arrow in the outer ring one in the middle ring and one in the inner ring x y z 43 3 Target 3 47 points One arrow in the outer ring and two in the inner ring x 2z 47

gemini_3.5_flash: By analyzing the positions of the arrows on each target we can set up a system of equations where O M and I represent the point values of the outer middle and inner rings respectively 1 First target O 2M 29 2 Second target O 2I 43 3 Third target M 2I 47 Subtracting the second equation from the third gives M O 4 or M O 4 Substituting this into the first equation yields O 2O 4 29 which simplifies to 3O 21 so O 7 This gives M 11 and I 18 For the fourth target which has one arrow in each of the three regions the score is O

gemma_4_31b_it: The user wants to find the score of the fourth target based on the provided image and information 1 Analyze the image There are four targets Each target has three concentric circles Each target has three arrows Lets denote the ringsSSC center circle M middle ring and O outer ring as the scoring zones

glm_4.5v: The user is asking me to solve a problem about archery targets Let me analyze the image and the problem The problem states Jeffrey shoots three arrows at each of four identical targets First target 29 points Second target 43 points Third target 47 points Question How many points on the last target Looking at the image there are four circular targets with concentric rings Each target has 3 arrows Let me count how many arrows are in each ring for each target

gpt_5.4_mini: Jeffrey scores 36 points on the last target

kimi_k2.5: Looking at the image I identify the arrow placements on each target across the three concentric scoring rings center middle outer Determining the ring values Target 1 0 center 2 middle 1 outer 29 points Target 2 2 center 0 middle 1 outer 43 points Target 3 2 center 1 middle 0 outer 47 points Let center a middle b outer c From the equations 2b c 29 2a c 43 2a b 47

qwen3.6_plus: The user wants me to solve a puzzle based on an image of four targets 1 Analyze the input Image Four targets each with three concentric circles inner middle outer Each target has three arrows hitting it Text Jeffrey shoots three arrows at each of four identical targets Target 1 score 29 points Target 2 score 43 points Target 3 score 47 points

qwen3.7_flash: The user wants me to solve a math puzzle based on an image 1 Analyze the input Text Jeffrey shoots three arrows at each of four identical targets He scores 29 points on the first target 43 on the second and 47 on the third How many points does Jeffrey score on the last target Image Four targets Each target has three concentric circles rings Lets call the regions from inside out Inner Circle center Middle Ring Outer Ring Wait lets look closer Theres a central circle Theres a ring around it Theres an outer ring So there are 3 distinct scoring zones Lets define variables for the points in each zone Let x be the points

Figure 11: Example from the benchmark and evaluation results. Red denotes incorrect and Green denotes correct answer marked by the judge model.