

SOLIDGEO: Measuring Multimodal Spatial Math Reasoning in Solid Geometry

Peijie Wang^{*1,2}, Chao Yang^{*3}, Zhong-Zhi Li^{1,2}, Fei Yin^{1,2}, Dekang Ran^{1,2}
Mi Tian⁴, Zhilong Ji⁴, Jinfeng Bai⁴, Cheng-Lin Liu^{†1,2}

¹MAIS, Institute of Automation of Chinese Academy of Sciences

²School of Artificial Intelligence, University of Chinese Academy of Sciences

³University of Electronic Science and Technology of China ⁴TAL

{wangpei jie2023, lizhongzhi2022, randekang2025}@ia.ac.cn
{jizhilong, baijinfeng, tianmi}@tal.com {fyin, liucl}@nlpr.ia.ac.cn

Abstract

Geometry is a fundamental branch of mathematics and plays a crucial role in evaluating the reasoning capabilities of multimodal large language models (MLLMs). However, existing multimodal mathematics benchmarks mainly focus on plane geometry and largely ignore solid geometry, which requires spatial reasoning and is more challenging than plane geometry. To address this critical gap, we introduce **SOLIDGEO**, the first large-scale benchmark specifically designed to evaluate the performance of MLLMs on mathematical reasoning tasks in solid geometry. SOLIDGEO consists of 3,113 real-world K–12 and competition-level problems, each paired with visual context and annotated with difficulty levels and fine-grained solid geometry categories. Our benchmark covers a wide range of 3D reasoning subjects such as projection, unfolding, spatial measurement, and spatial vector, offering a rigorous testbed for assessing solid geometry. Through extensive experiments, we observe that MLLMs encounter substantial challenges in solid geometry math tasks, with a considerable performance gap relative to human capabilities on SOLIDGEO. Moreover, we analyze the performance, inference efficiency and error patterns of various models, offering insights into the solid geometric mathematical reasoning capabilities of MLLMs. We hope SOLIDGEO serves as a catalyst for advancing MLLMs toward deeper geometric reasoning and spatial intelligence. The dataset is released at [SOLIDGEO](#).

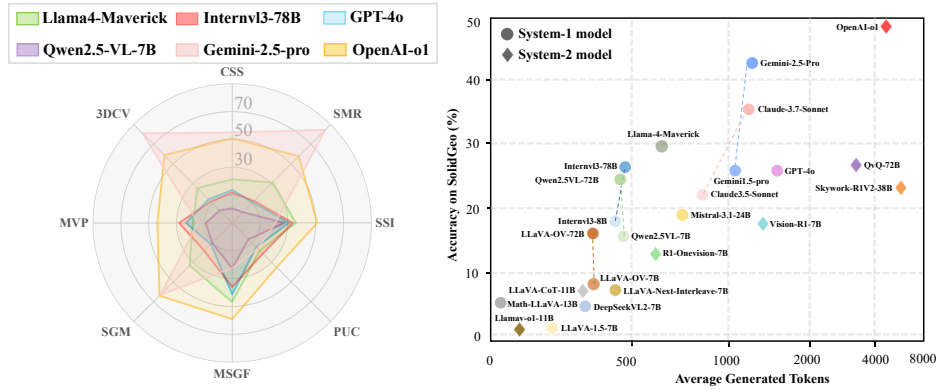


Figure 1: Performance of six MLLMs on SOLIDGEO benchmark across 8 solid geometry subjects (left), and trade-off between accuracy and average generated token length across 25 MLLMs (right).

* Equal contribution † Corresponding author

1 Introduction

"There is no royal road to geometry." — Euclid

Geometric problems hold a vital position in mathematics and are widely regarded as the foundation and core of mathematics [1, 2]. The challenge of geometric problem solving lies in the integration of complex visual information and symbolic reasoning. Based on the structural properties of geometric figures, geometry can be categorized into plane geometry and solid geometry [3]. Compared to plane geometry, solid geometry involves understanding three-dimensional structures and spatial relationships, making it inherently more complex. Requiring advanced spatial reasoning and the integration of visual and textual modalities, solid geometry represents a particularly challenging class of problems for artificial intelligence systems. [4, 5, 6, 7].

In recent years, advances in Large Language Models (LLMs) [8, 9, 10, 11, 12, 13] and Multimodal Large Language Models (MLLMs) [14, 15, 16, 17, 12, 18] have led to impressive performance across a wide range of language and visual-language tasks, such as natural language understanding, code generation, image captioning and visual question answering [19, 20, 21, 22, 23, 24, 25, 26, 27]. With the growing capabilities of MLLMs, their mathematical reasoning abilities have emerged as a critical focus of recent research. Notably, recent models such as GPT-4o [28], Gemini [29], Qwen-VL [30] and InternVL [16] have surpassed the average human performance on MathVista [31], one of the most widely used benchmarks for multimodal mathematical reasoning.

To better understand the limitations of current multimodal mathematical reasoning benchmarks, we conducted a detailed analysis of existing datasets with a focus on geometry-related content. Our investigation reveals two key issues:

Firstly, although geometry problems are among the most prevalent types of mathematical tasks—as evidenced by the abundance of benchmarks such as Geometry3K [32], GeoQA [33], PGPS9K [34], UniGeo [35] and GeomRel [36]—most existing geometry benchmarks overwhelmingly concentrate on *plane geometry*. In contrast, *solid geometry*, which entails reasoning about three-dimensional structures and their spatial properties such as projection and spatial measurement, remains severely underrepresented. This imbalance persists despite the fundamental role that solid geometry plays in both human curricula and machine reasoning. Crucially, solving solid geometry problems demands advanced spatial reasoning—an essential aspect of spatial intelligence, which has been identified as a key competency on the road toward artificial general intelligence (AGI) [37].

Secondly, although the visual data in existing benchmarks span a wide range of sources and modalities, the scope and depth of solid geometry problems remain notably limited. For instance, MathVista contains only 62 solid geometry questions, all involving simple object counting tasks [31]. MathVision includes 244 such problems [6], and MathVerse provides 119 [7], but we found that the majority focus on recognizing object structures, shapes, or performing basic volume calculations. Critically, more advanced problem types such as projection analysis, spatial transformations, and complex spatial relationship reasoning are rarely represented. This imbalance underscores a significant gap in the current landscape of multimodal mathematical benchmarks: the absence of rich, diverse, and cognitively demanding tasks targeting solid geometry reasoning.

Despite recent progress in multimodal mathematical reasoning, the current benchmarks for solid geometry remain narrow in both topical coverage and problem diversity. As a result, the spatial reasoning capabilities of MLLMs—particularly in 3D geometric contexts—have not been adequately assessed. Solid geometry inherently integrates symbolic mathematical reasoning with spatial intelligence and is regarded as essential for achieving AGI [2, 37]. Yet, this critical domain remains largely overlooked in existing evaluations. To bridge this gap, we introduce the **SOLIDGEO** dataset. Unlike prior datasets that only sparsely include solid geometry content and focus on basic recognition or counting tasks, **SOLIDGEO** offers a comprehensive collection of diverse and challenging problems that span a wide range of solid geometry concepts. Our goal is to establish a rigorous benchmark that better reflects the demands of real-world solid geometry reasoning and pushes the frontier of spatial intelligence in multimodal AI systems.

SOLIDGEO comprises solid geometry problems sourced from six existing multimodal math datasets, along with a collection of high-quality questions that we newly curated and annotated from real-world K–12 and high school competition educational scenarios. In total, SOLIDGEO contains **3,113** solid geometry problems, each accompanied by at least one image. Among these, 1,737 samples are derived from existing datasets, while 1,376 are newly collected and verified by our team. The dataset

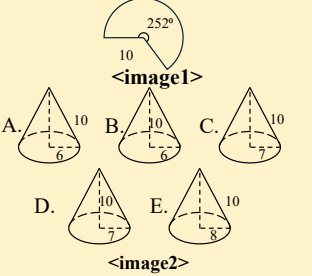
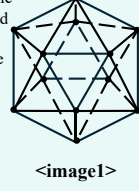
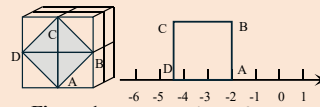
Multiple Choice	Open-ended: Single-Step	Open-ended: Multi-Step
Question: Which of the cones listed below can be formed from a 252° sector of a circle of radius 10 by aligning the two straight sides? <image1> <image2> A. A B. B C. C D. D E. E  <image2>	Question: A regular icosahedron is a 20-faced solid where each face is an equilateral triangle and five triangles meet at every vertex. The regular icosahedron shown below has one vertex at the top, one vertex at the bottom, an upper pentagon of five vertices all adjacent to the top vertex and all in the same horizontal plane, and a lower pentagon of five vertices all adjacent to the bottom vertex and all in another horizontal plane. Find the number of paths from the top vertex to the bottom vertex such that each part of a path goes downward or horizontally along an edge of the icosahedron.  <image1>	Question: As shown in Figure 1, this is a Rubik's Cube composed of 8 equally sized smaller cubes, with a total volume of 64. (1) Determine the edge length of the Rubik's Cube. (2) In the figure, quadrilateral ABCD is a square. Find the area and the side length of this square. (3) Place the square ABCD on the number line as shown in Figure 2, such that point A coincides with -1, point E coincides with 1, and point F is symmetric to point D with respect to point E. Then, the number represented by point D on the number line is $______$. <ImageHere> Figure1 <ImageHere> Figure 2  Figure 1 Figure 2
Answer: C Difficulty Level: 2 Problem Type: PUC	Answer: 810 Difficulty Level: 3 Problem Type: CSS	Answer: (1) 4 (2) $8\sqrt{2}$ (3) $-1-2\sqrt{2}$ Difficulty Level: 2 Problem Type: CSS

Figure 2: Sampled SOLIDGEO question examples from each question type. Each sample contains a visual context, difficulty levels and fine-grained solid geometry categories.

covers three problem formats: multiple-choice, open-ended single-step, and open-ended multi-step questions. As shown in Figure 2, every question in SOLIDGEO contains at least one image input, annotated with difficulty levels and fine-grained solid geometry categories. Specifically, we introduce three major improvements in SOLIDGEO:

1. Fine-grained Category. We present the first fine-grained categorization of solid geometry problems, dividing all questions into eight reasoning-based subcategories. This classification captures core aspects of spatial intelligence and enables a more structured evaluation of model capabilities.

2. Real-world Problem. All problems and images in SOLIDGEO are sourced from authentic scenarios, resulting in naturally phrased and diverse questions. Notably, SOLIDGEO has a much higher average question length (77.2) compared to MathVista (15.6) and MathVision (42.3), offering richer contextual information and posing greater challenges.

3. Difficulty level. Each problem is labeled with a difficulty level from 1 to 3, verified by domain experts. This enables fine-grained model analysis and helps identify reasoning bottlenecks, particularly for o1-like models whose efficiency may be sensitive to problem complexity.

We conduct extensive experiments on SOLIDGEO to comprehensively assess model performance in solid geometry reasoning tasks. Among the evaluated models, OpenAI-o1 achieves the highest accuracy with a score of 49.5%. Notably, the open-source model Llama 4 [38] performs competitively, scoring 29.6% and outperforming GPT-4o, while ranking just below Claude-3.7-Sonnet. However, all models still fall significantly short of human-level performance. Our analysis provides insights into the strengths and limitations of current open-source MLLMs in handling complex solid geometry spatial reasoning. In summary, our main contributions are as follows:

- We carefully reviewed existing datasets and found that solid geometry problems are insufficiently covered. Despite being crucial for spatial reasoning, a core capability for achieving AGI, solid geometry has been underestimated in previous benchmarks.
- We present **SOLIDGEO**, the first benchmark dedicated to solid geometry mathematical reasoning. It comprises 3,113 problems with visual context, drawn from real-world K-12 and competition sources. Problems are categorized into 3 difficulty levels and 8 fine-grained categories.
- Leveraging fine-grained annotations, we conduct a detailed evaluation of 27 MLLMs, providing insights into their geometric spatial reasoning capabilities and offering analysis of inference efficiency for o1-like models, while identifying key limitations for future research.

2 Related Works

Benchmark for Mathematical Reasoning. Various benchmarks have been proposed to evaluate the mathematical reasoning capabilities of MLLMs. Early multimodal math datasets such as GEOS [39], GeoQA [33], Geometry3K [32], and UniGeo [35] cover only a narrow range of topics and primarily focus on plane geometry. More recent benchmarks like MMMU [40] include only a small subset of math-related questions and lack any coverage of solid geometry. MathVista [31] includes just 62 solid geometry problems, all of which are basic object counting tasks that do not require complex spatial reasoning. Although MATH-Vision [6], WE-MATH [41], MV-MATH [42], and GeoSense [43] present more diverse and rigorous math problems, they contain only a limited number of solid geometry samples and lack fine-grained categorization. In contrast, SOLIDGEO offers a comprehensive and diverse set of solid geometry problems. It encompasses both visual recognition and complex spatial reasoning, and features explicit difficulty annotations and 8 fine-grained categories—enabling more systematic evaluation of model performance on solid geometry tasks.

Multimodal Models. With the advancement of LLMs and vision-language alignment techniques, early multimodal models such as MiniGPT-4 [44], LLaMA-Adapter [45], and LLaVA [14] demonstrated promising capabilities in visual understanding. Recent research also explores parameter-efficient fine-tuning (PEFT) approaches such as LoRA [46] and its improved orthogonal reinitialization strategies UORA [47]. More recent models such as closed-source GPT-4o [28], Claude [48], Gemini [29] and open-source Qwen-VL [30], InternVL [16], and LLaVA-Onevision [49] have further pushed the boundaries of general-purpose visual reasoning. The recent release of OpenAI’s o1 model has highlighted the effectiveness of long Chain-of-Thought in improving reasoning performance, this has inspired a wave of o1-style models [50, 51, 52, 53, 54, 55, 56], many of which show strong performance on multimodal mathematical tasks. Beyond step-wise reasoning, Chain-of-Reasoning [57] advocates for a unified multi-paradigm approach, integrating symbolic, visual, and CoT-based reasoning for mathematical problem solving.

Several multimodal models have been specifically developed for mathematical reasoning, including G-LLaVA [58], UniMath [59], LANS [60] and GeoUni [61]. However, these approaches primarily focus on plane geometry tasks. Even AlphaGeometry [2], a geometry model that achieves IMO-level, remains ineffective in handling solid geometry. Despite its importance for spatial reasoning, solid geometry remains an underexplored area for current multimodal models.

3 The SOLIDGEO Benchmark

3.1 Overview

We present **SOLIDGEO**, a carefully curated benchmark designed to evaluate the multimodal spatial and mathematical reasoning capabilities of foundation models in the domain of solid geometry. Unlike previous benchmarks that primarily emphasize plane geometry or provide only limited coverage of solid geometry, SolidGeo targets the unique challenges of understanding and reasoning over three-dimensional structures and their spatial relationships. Solving such problems requires advanced spatial intelligence such as interpreting projections and analyzing 3D configurations, which are essential for assessing a model’s ability to integrate visual perception with symbolic reasoning.

SOLIDGEO comprises 3,113 real-world solid geometry problems, collected from K–12 curricula and high school mathematics competitions. Each problem is paired with at least one image and annotated with a difficulty level ranging from 1 (easy) to 3 (hard). The dataset is categorized into eight fine-grained domains: *Composite Solid Structures*, *Spatial Metric Relations*, *Solid Shape Identification*, *Planar Unfolding and Configuration*, *Measurement of Solid Geometric Forms*, *Solid Geometry Modeling*, *Multi-view Projection*, and *3D Coordinate and Vector Reasoning*. These annotations support more granular analysis of model strengths and weaknesses across different task types and reasoning complexities. Compared to existing multimodal math benchmarks, SOLIDGEO offers a richer, more diverse, and cognitively demanding collection of problems. The average question length is substantially higher, reflecting the depth of contextual information and the complexity of reasoning required. Detailed statistics and coverage of SOLIDGEO are presented in Table 1 and Figure 3.

3.2 Data Construction

Data Collection. To construct SOLIDGEO, we adopt a hybrid data collection strategy that combines solid geometry problems from existing multimodal mathematical reasoning datasets with newly

Statistic	Number
Total questions	3,113
- Multiple-choice questions	969
- Open-ended questions	2144
- Single-step questions	1936
- Multi-step questions	208
Difficulties (Easy: Medium: Hard)	32%:61%:7%
Newly collected questions	1,376 (44.2%)
Existing-dataset questions	1,737 (55.8%)
Newly collected images	3,555 (66.1%)
Existing-dataset images	1,825 (33.9%)
Language	
- English (EN)	2,192 (70.4%)
- Chinese (CN)	921 (29.6%)
Maximum question length	679
Maximum answer length	2833
Average question length	77.2
Average answer length	312.2

Table 1: Key Statistics of SOLIDGEO.

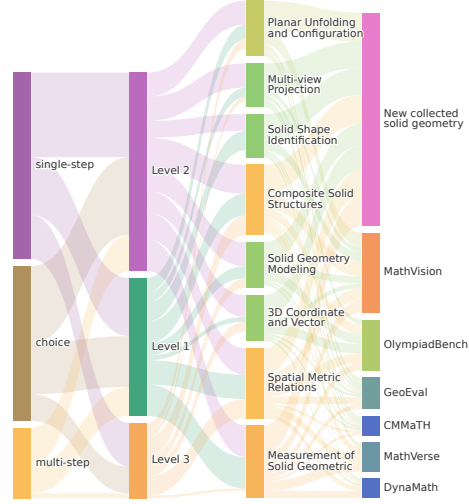


Figure 3: Distribution of SOLIDGEO.

gathered problems from real-world K–12 educational sources. Specifically, we first extracted solid geometry samples from six existing benchmarks based on their provided category labels. To supplement these, we further collected problems from the Zujian platform, which provides a large-scale repository of K–12 math problems in PDF format. Using the Mathpix API², we extracted structured content including questions, answers, analyses and associated diagrams, resulting in an initial pool of 10,932 candidate problems identified via keyword filtering.

Data Filtering. To ensure the quality and relevance of the collected solid geometric data, we designed a four-stage filtering pipeline:

- Stage 1: Structural Filtering. We excluded purely textual problems and retained only multimodal samples with at least one associated image. We also removed samples missing essential fields such as answers or diagrams.
- Stage 2: Image Quality Filtering. Using OpenCV, we computed sharpness metrics to eliminate samples containing low-resolution or blurry images, ensuring the retained diagrams were visually clear and suitable for spatial reasoning tasks.
- Stage 3: Semantic Filtering. To refine topic relevance beyond keyword matching, we used the DeepSeek API [13] to classify whether each candidate problem truly belonged to the solid geometry domain, filtering out unrelated samples.
- Stage 4: Cross-Set Deduplication. To prevent redundancy, we applied n-gram [62] based similarity checks between the newly collected samples and those from existing datasets. This ensured that SOLIDGEO contains no near-duplicate or overly similar problems across sources.

Following this pipeline, we retained 1,376 high-quality new problems, which were combined with 1,737 problems from existing datasets, yielding the final SOLIDGEO dataset with 3,113 unique samples. The dataset covers three problem formats: 969 multiple-choice, 1,936 open-ended single-step, and 208 open-ended multi-step questions. Further details on dataset statistics and the filtering process are provided in Appendix A and Appendix D.

Data Labeling. The SOLIDGEO dataset incorporates two types of annotations: fine-grained subject categorization and difficulty level assignment. To achieve accurate labeling, we first invited subject-matter experts to define eight representative subjects of solid geometry, covering the full spectrum of spatial reasoning skills required. Each problem was then categorized using a majority voting scheme based on predictions from three advanced MLLMs: GPT-4o, Claude-3.7-Sonnet, and Qwen-VL-Max. Some problems may belong to multiple subjects when applicable. Similarly, difficulty levels ranging from 1 (easy) to 3 (hard) were determined using model-based voting among the same three MLLMs. In cases where consensus could not be reached, final decisions were made by experienced human

²<https://mathpix.com/convert>

Table 2: Comparison with existing multimodal math benchmarks. SG: Solid Geometry, PG: Plane Geometry. **Level:** K=K-12, U=University, C=Competition. **Source:** S=Self-sourced, P=Collected from Public Dataset. MC: Multiple Choice, SS: Single-Step, MS: Multi-Step.

Benchmarks	Language	SG Size	SG Proportion	PG Proportion	Level	Source	SG category	MC	SS	MS
GeoQA [33]	CN	0	0.0%	100.0%	K	S	X	✓	X	X
Geometry3K [32]	EN	0	0.0%	100.0%	K	S	X	✓	✓	X
UniGeo [35]	EN	0	0.0%	100.0%	K	S	X	X	✓	X
PGPS9K [34]	EN	0	0.0%	100.0%	K	S	X	✓	✓	X
MMM-U-MATH [40]	EN	0	0.0%	29.3%	U	S	X	✓	✓	X
GeoEval [5]	EN	100	2.0%	94.1%	K	P	X	✓	X	X
DynaMath [63]	EN	150	3.0%	15.4%	K	S	X	✓	✓	X
MATH-Vision [6]	EN	263	8.7%	58.7%	K U	S	X	✓	✓	X
OlympiadBench [64]	EN/CN	784	9.2%	15.6%	C	S	X	X	✓	X
MathVerse [7]	EN	119	15.1%	64.7%	K	P	X	✓	✓	X
GeoSense [43]	EN/CN	350	20.0%	80.0%	K	P	2	✓	✓	X
SOLIDGEO (Ours)	EN/CN	3113	100.0%	0.0%	K C	P	8	✓	✓	✓

annotators. To ensure annotation quality and consistency, all 3,113 problems finally underwent independent review by three expert annotators, who verified both the subjects and difficulty labels.

3.3 Comparison with Existing Benchmarks

Most existing mathematical reasoning benchmarks include only a limited number of solid geometry problems. For example, MathVista [31] contains just 62 solid geometry questions, all following a single templated format: "*Subtract all ... objects. How many objects are left?*", offering minimal diversity or reasoning depth. MathVerse [7] expands its dataset by restating problems, but its original solid geometry set comprises only 119 unique examples. MathVision [6] provides 244 solid geometry questions, though most of them are concentrated in the domain of structural analysis. Benchmarks such as GeoQA [33], Geometry3K [32], UniGeo [35], and PGPS9K [34] focus almost exclusively on plane geometry, with little or no coverage of 3D reasoning.

In contrast, SOLIDGEO is the first large-scale benchmark dedicated to solid geometry. Each problem is paired with at least one visual input and has been manually verified for correctness. Unlike prior datasets that use broad or ambiguous labels, SOLIDGEO introduces a refined taxonomy with eight fine-grained categories, explicitly capturing the diversity of solid geometry subdomains. Furthermore, SOLIDGEO features a significantly longer average question length (77.2 words), compared to MathVista (15.6) and MathVision (42.3), indicating higher contextual richness and greater reasoning complexity. Together, these attributes make SOLIDGEO a comprehensive and challenging benchmark for evaluating multimodal mathematical reasoning in 3D spatial contexts. A detailed comparison with existing benchmarks is presented in Table 2. See Appendix C for details.

4 Experiments

In this section, we conduct a systematic evaluation of existing LLMs and MLLMs on SOLIDGEO. We first introduce the experimental setup in Section 4.1. Then, we detail the quantitative results in Section 4.2 and narrate the error analysis in Section 4.3. The experimental outcomes show that most of the current MLLMs are still far behind the human level in solid geometry.

4.1 Experimental Setup

Evaluation Models. We evaluate a diverse set of models on SOLIDGEO, spanning both LLMs and MLLMs, including 20 open-source and 7 closed-source models. These models fall into two categories: 19 **System-1 models**, which follow a fast, single-pass reasoning paradigm, and 8 **System-2 models**, which adopt a slow, iterative long CoT reasoning style inspired by o1-type designs [78].

Implementation Details. Our evaluation is conducted under three settings: zero-shot direct answering, Chain-of-Thought (CoT), and CoT with 2-shot examples. To establish a human performance baseline, we recruit high school students to independently complete the questions as detailed in Appendix F. For multiple-choice, single-step, and multi-step open-ended formats, we carefully design tailored prompts to ensure models generate responses in the correct structure. To assess the contribution of visual information, we additionally evaluate performance using LLM (text-only) and GPT-4o with text-only inputs. This enables a controlled comparison between purely textual reasoning and full multimodal inputs, thereby quantifying the role of images in solid geometry reasoning. Answer

Table 3: Comparison of model performances across 8 fine-grained solid geometry subjects and average output tokens. CSS: Composite Solid Structures, SMR: Spatial Metric Relations, SSI: Solid Shape Identification, PUC: Planar Unfolding and Configuration, MSGF: Measurement of Solid Geometric Forms, SGM: Solid Geometry Modeling, MVP: Multi-view Projection, 3DCV: 3D Coordinate and Vector Reasoning. The **first** and **second** highest accuracy of LLMs are marked in red and blue, respectively.

Model	Overall	CSS	SMR	SSI	PUC	MSGF	SGM	MVP	3DCV	Avg.tokens
Text-only, zero-shot direct answering										
Deepseek-V3[13] (LLM)	9.3	10.7	8.1	8.3	12.7	6.3	7.8	10.3	12.2	787.2
GPT-4o[28] (MLLM)	9.1	10.0	10.4	10.6	6.8	12.1	8.6	7.3	9.6	692.6
Open-source MLLMs (Text + Image, zero-shot direct answering)										
<i>System-1 Models</i>										
LLaVA-v1.5-7B[14]	1.8	1.1	1.1	6.7	2.2	0.6	0.0	4.6	0.0	246.2
InternLM-XComposer2.5-VL-7B[65]	4.4	2.5	1.8	6.7	8.9	0.6	0.0	9.4	1.2	151.8
DeepSeek-VL2-7B[66]	5.1	2.8	2.6	11.1	5.1	1.4	1.8	11.7	1.8	338.2
Math-LLaVA-13B[67]	5.9	4.2	4.1	7.6	11.7	2.7	4.2	12.6	6.2	7.4
LLaVA-NeXT-Interleave-7B[68]	7.7	2.5	2.3	21.5	13.5	2.3	7.3	16.7	0.6	486.3
LLaVA-OneVision-Chat-7B[49]	8.6	4.3	2.9	19.3	15.2	3.5	6.4	17.9	0.0	353.2
Qwen2.5-VL-Instruct-7B[30]	15.5	8.4	8.8	30.1	13.3	26.2	16.2	15.2	10.2	490.2
LLaVA-OneVision-Chat-72B[49]	15.9	13.2	9.5	31.9	18.1	12.9	11.8	23.7	8.4	396.3
InternVL3-8B[16]	17.7	11.8	10.0	24.4	17.4	28.0	19.1	19.9	7.2	488.8
Mistral-small-3.1-24b-instruct[69]	19.6	15.2	15.8	27.4	17.1	28.9	10.9	17.0	16.8	769.7
Qwen2.5-VL-Instruct-72B[30]	24.2	19.7	18.8	29.6	21.5	35.4	16.4	22.5	18.0	485.0
InternVL3-78B[16]	26.2	17.4	17.9	34.8	24.9	36.8	22.7	30.5	17.4	493.2
Llama-4-Maverick-17B-128E[38]	29.6	25.1	30.9	34.6	20.5	43.4	32.6	20.7	26.3	605.6
<i>System-2 Models</i>										
LlamaV-o1-11B [70]	1.5	0.6	0.7	1.5	0.5	5.0	2.7	0.1	0.0	106.1
LLaVA-CoT-11B [71]	7.3	4.2	2.5	7.4	6.5	15.1	8.2	7.4	1.8	401.7
VLM-R1-3B [72]	9.6	6.3	4.4	11.1	8.7	19.6	4.5	8.3	2.4	453.0
R1-Onevision-7B [73]	13.2	7.7	9.7	25.2	10.1	23.3	11.8	12.3	9.0	522.3
Vision-R1-7B [74]	18.1	11.7	11.3	28.6	17.8	26.9	13.9	19.3	12.0	1498.7
Skywork-R1V2-38B [75]	23.0	18.4	29.5	13.3	11.6	31.2	30.0	12.3	26.9	5682.9
QvQ-72B-Preview [76]	26.6	17.9	28.1	37.0	22.9	34.7	20.9	20.3	22.8	3622.2
Closed-source MLLMs (Text + Image, zero-shot direct answering)										
<i>System-1 Models</i>										
Claude-3.5-Sonnet[48]	22.2	16.9	9.8	42.2	24.2	36.5	25.5	23.5	9.6	992.1
GPT-4V[15]	25.3	16.6	15.8	35.6	21.5	41.5	25.5	25.9	18.0	1433.5
Gemini-1.5-pro[29]	25.3	18.5	16.8	34.8	19.6	41.6	17.3	25.6	19.2	1003.5
GPT-4o[28]	25.5	18.9	16.8	32.6	19.6	41.0	17.3	26.5	19.2	1344.9
Claude-3.7-Sonnet[48]	34.1	27.7	28.2	43.0	32.9	46.8	43.6	28.5	26.3	1217.4
Gemini-2.5-pro[29]	42.7	52.0	75.7	24.8	20.9	26.0	58.4	19.6	72.9	1263.9
<i>System-2 Models</i>										
OpenAI-o1[77]	49.5	48.7	54.2	48.9	36.1	55.3	59.1	43.0	55.1	4942.6
Human Performance										
Human	77.5	88.2	70.9	90.2	77.2	87.4	71.2	78.5	69.2	-

evaluation is performed using the DeepSeek API³, with customized prompts tailored to each question type to ensure accurate and consistent scoring. Experiments are conducted on NVIDIA A800 GPUs. Additional implementation details are provided in the Appendix G.

4.2 Experimental Results

We show the evaluation results of various models on SOLIDGEO in Table 3 and report the model performance by difficulty level, question type, and under different prompts in Table 4.

Challenging Nature of SOLIDGEO. The results presented in Table 2 highlight the inherent difficulty of SOLIDGEO. The best-performing model, OpenAI-o1, achieves an overall accuracy of 49.5%, followed by Gemini-2.5-pro at 42.7%, and Claude-3.7-Sonnet at 34.1%. Notably, all other models fall below the 30% threshold, underscoring the substantial challenge that solid geometry reasoning poses even for advanced MLLMs. Among the open-source system-1 models, the strongest model is Llama4, a 400B MoE model with 128 experts and 17B active parameters, which achieves 29.6% accuracy. In the System-2 category, the best-performing open-source model is QvQ-72B, reaching 26.6%. Despite these results, even the most advanced model OpenAI-o1 still falls significantly short of human performance, highlighting the limitations of current MLLMs in spatial reasoning. Additionally, there remains a notable performance gap between open-source and closed-source models, which shows the need for continued advancements in training strategies, data quality, and architectural innovations.

³<https://www.deepseek.com/>

Table 4: Model Performance across Different Prompt Settings, Difficulty Levels, and Question Types. MC: Multiple Choice, SS: Single-Step, MS: Multi-Step.

Models	Original	CoT	CoT & 2-shot	Level 1	Level 2	Level 3	MC	SS	MS
Claude-3.5-sonnet [48]	22.2	21.7(-0.5)	19.3(-1.9)	37.7	15.7	4.5	32.8	17.8	13.9
Gemini-1.5-pro [29]	25.3	26.1(+0.8)	27.5(+2.2)	39.0	19.7	8.5	32.1	23.1	14.4
GPT-4V [15]	25.3	24.1(-1.2)	23.4(-1.9)	41.6	18.5	6.8	35.2	22.1	10.6
GPT-4o [28]	25.5	24.9(-0.6)	22.9(-2.6)	38.9	20.1	7.9	32.7	23.2	13.0
Claude-3.7-Sonnet [48]	34.1	-	-	42.5	31.2	16.4	39.9	32.5	21.2
Gemini-2.5-pro [29]	42.7	-	-	22.1	50.8	80.7	35.2	49.0	19.3
OpenAI-o1 [77]	49.6	-	-	46.6	50.4	57.4	50.8	51.2	28.8
LLaVA-OneVision-Chat-7B [49]	8.6	10.5(+1.9)	11.4(+2.8)	16.1	5.4	1.1	22.1	2.6	1.4
Qwen2.5-VL-Instruct-7B [30]	15.5	16.0(+0.5)	15.0(-0.5)	28.1	10.0	2.8	22.2	13.1	6.2
LLaVA-OneVision-Chat-72B [49]	15.9	14.8(-1.1)	15.6(-0.3)	23.5	13.0	4.0	29.2	10.2	6.2
InternVL3-8B [16]	17.7	18.1(+0.4)	18.3(+0.6)	30.7	11.9	6.2	27.3	14.1	6.2
Qwen2.5-VL-Instruct-72B [30]	24.2	28.8(+4.6)	26.1(+1.9)	33.5	20.6	10.6	28.5	22.8	17.3
InternVL3-78B [16]	26.2	25.8(-0.4)	25.9(-0.3)	42.2	19.7	6.2	35.1	23.0	15.4
QvQ-72B-Preview [76]	26.6	-	-	32.6	24.5	14.7	31.5	26.3	6.7

Comparison among different Subjects. As shown in Table 2, models generally underperform on tasks requiring complex spatial reasoning, such as Planar Unfolding and Configuration (PUC), Multi-view Projection (MVP), 3D Coordinate and Vector Reasoning (3DCV). Even the strongest model, OpenAI-o1, achieves only 36.1% on PUC and 43.0% on MVP. Interestingly, Gemini-2.5-pro exhibits a notable performance pattern—surpassing human-level accuracy on 3DCV and Spatial Metric Relations (SMR), the most cognitively demanding categories for humans, while underperforming on comparatively simpler tasks where humans excel, such as Solid Shape Identification (SSI).

Failure of CoT/few-shot. As shown in Table 4, Chain-of-Thought (CoT) and few-shot prompting do not consistently improve model performance on SOLIDGEO. Models like Qwen2.5-VL-72B and Gemini-1.5-pro benefit from CoT and 2-shot settings, with accuracy increasing from 24.2% to 28.8% and 26.1%, and from 25.3% to 26.1% and 27.5%, respectively. However, other models such as GPT-4o and LLaVA-OV-72B exhibit performance degradation under the same conditions. To further understand this, we analyze the effect of prompting strategies on the average number of generated tokens, as shown in Figure 4. For many models, neither CoT nor 2-shot leads to longer outputs, suggesting such reasoning capabilities may already be learned during pretraining. Notably, we observe that only when CoT increases token count by over 10% does it yield tangible gains—e.g., Qwen2.5-VL-72B: tokens $\uparrow$ 21.9%, accuracy $\uparrow$ 4.6%; LLaVA-OV-7B: tokens $\uparrow$ 20.6%, accuracy $\uparrow$ 1.9%. In contrast, 2-shot prompting shows no consistent correlation with performance improvement, indicating it is not a universally effective strategy. Efficient long-context modeling remains a challenge, despite recent advances in scalable context generalization techniques [79].

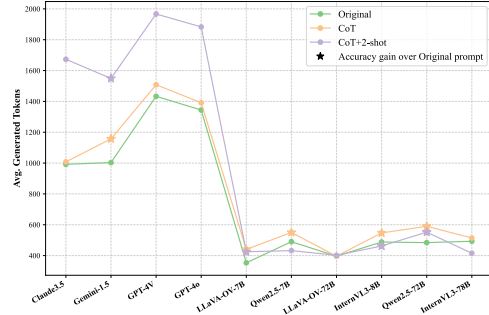


Figure 4: Average number of generated tokens for different prompting strategies across various models on SOLIDGEO.

Difficulty Levels and Question Types. Analyses in this paragraph are conducted under the original prompt setting. As shown in Table 4, most models exhibit a clear decline in accuracy as task difficulty increases. For instance, InternVL3-78B achieves 42.2% on Level 1 but drops sharply to just 6.2% on Level 3, underscoring the increasing complexity of harder solid geometry problems. Interestingly, Gemini-2.5-pro and OpenAI-o1 defy this trend—their performance improves with higher difficulty levels. Gemini-2.5-pro reaches an impressive 80.7% on Level 3 while scoring only 22.1% on Level 1, suggesting a potential overfitting to complex problem structures or a struggle with generalizing across simpler formats. Regarding question types, most models perform better on multiple-choice (MC) questions, likely due to reduced ambiguity from predefined options. However, Gemini-2.5-pro and OpenAI-o1 once again diverge from this pattern, achieving their highest scores on single-step (SS) questions. This counterintuitive behavior may reflect an internal ability to dynamically adjust reasoning strategies based on task complexity.

Model Inference Efficiency. As shown in Figure 5, System-2 models generally generate much longer outputs than System-1 models. While this often improves accuracy, it reduces inference

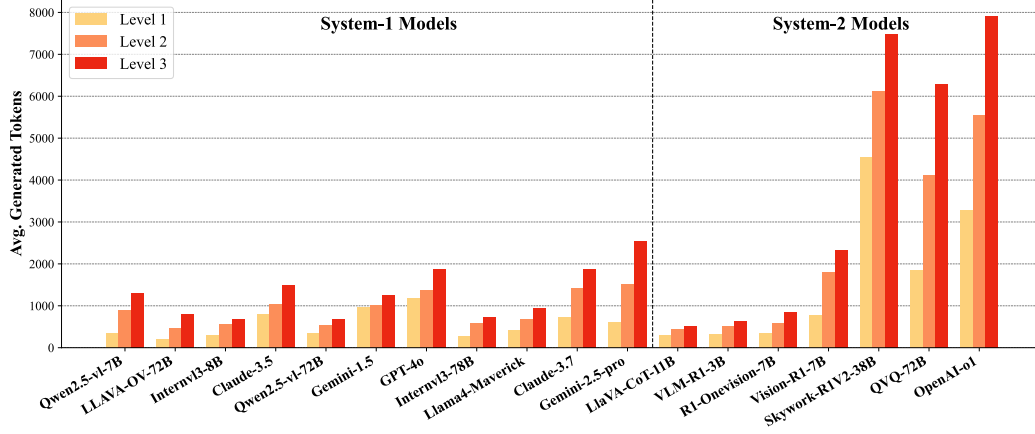


Figure 5: Average number of generated tokens by difficulty level of various models on SOLIDGEO.

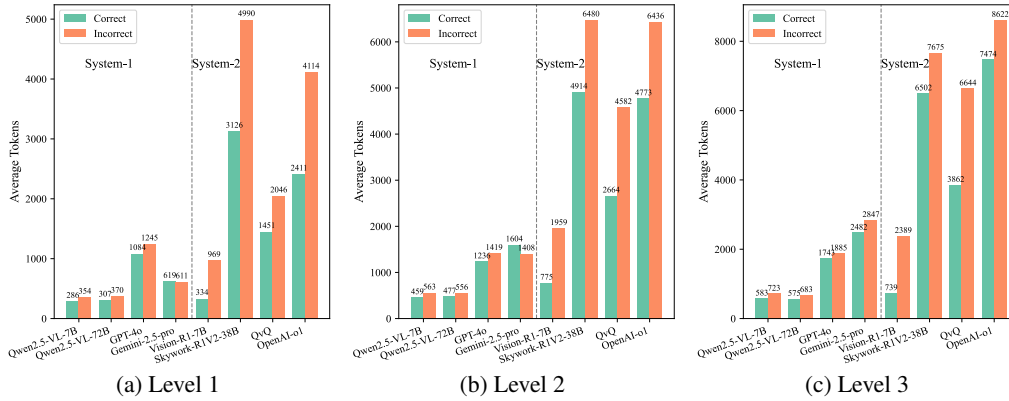


Figure 6: Average generated tokens by correctness across different complexity levels.

efficiency due to longer reasoning and latency. We also observe that output token counts increase with problem difficulty, suggesting that models require longer reasoning chains for more complex tasks. To better understand the link between output length and correctness, we analyze token usage by answer correctness across difficulty levels in Figure 6. Results show that models—especially System-2—tend to consume more tokens for incorrect answers than correct ones, without accuracy gains. This implies potential overthinking, where excessive reasoning fails to improve outcomes. Reducing unnecessary reasoning steps while maintaining performance remains a key challenge for building efficient and capable MLLMs. We provide further analysis in Appendix E.

4.3 Error Analysis

We conduct a fine-grained error analysis on two representative models: Gemini-2.5-pro and OpenAI-o1. Specifically, we randomly sample 300 incorrect predictions from each model, classify the errors into five categories, and analyze their distribution. The error taxonomy and proportions are visualized in Figure 7, with more details provided in the Appendix H.

Among all error types, visual perception errors and reasoning errors dominate, jointly accounting for over 70% of total failures. OpenAI-o1 exhibits a lower reasoning error rate (38%) compared to Gemini-2.5-pro (44%), suggesting that System-2 models with long CoT reasoning paradigms may partially mitigate faulty logical inference. However, OpenAI-o1 suffers from a higher visual perception error rate (41% vs. 34%), indicating that even advanced models struggle to accurately interpret spatial geometric elements. Moreover, we observe that OpenAI-o1 produces hallucinated content in 6% of cases—double the rate observed in Gemini-2.5-pro. This highlights hallucination as an emerging issue in complex long CoT generation settings, especially for System-2 models. An illustrative reasoning error is shown on the right side of Figure 7, where the model incorrectly applies logical reasoning without accounting for spatial structure.

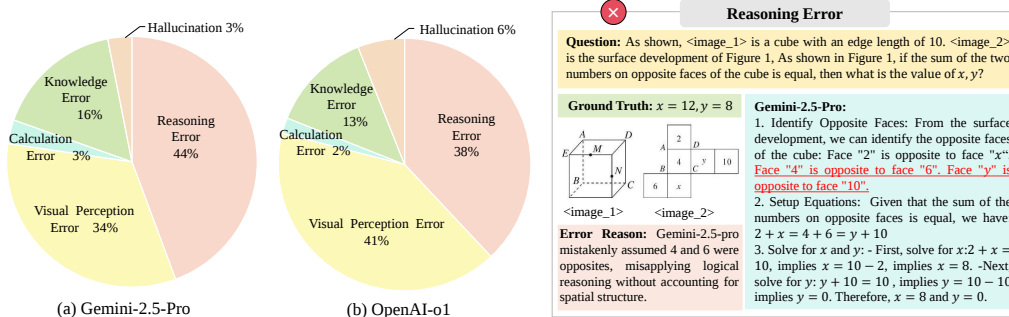


Figure 7: Error distribution of Gemini-2.5-pro and OpenAI-o1 and an example of reasoning error.

5 Conclusion

We introduce SOLIDGEO, the first large-scale benchmark designed to evaluate the solid geometry mathematical reasoning capabilities of MLLMs, addressing a critical gap in current multimodal benchmarks that overlook spatial and three-dimensional mathematical reasoning tasks. SOLIDGEO comprises a diverse and challenging set of real-world K-12 and competition-level problems enriched with visual contexts, difficulty levels, and fine-grained categorical annotations. Using this benchmark, we conduct a comprehensive evaluation of a wide range of open-source and closed-source models, revealing a significant performance gap between current MLLMs and human performance on solid geometry tasks. Furthermore, we analyze model inference efficiency across difficulty levels and response lengths, offering valuable insights into the current limitations and potential directions for future research. We hope SOLIDGEO will serve as a foundation for advancing the complex spatial reasoning capabilities of next-generation MLLMs.

Limitations and Future Works. SOLIDGEO has certain limitations. First, the dataset includes only English and Chinese problems, which may limit its applicability for evaluating multilingual reasoning capabilities [80]. Second, there currently exists no widely adopted formal language to represent solid geometry problems, making it difficult to standardize symbolic reasoning in this domain. Future work could explore designing a formal representation framework for solid geometry, which may further facilitate precise modeling and programmatic evaluation.

6 Acknowledgements

This work has been supported by the National Natural Science Foundation of China (NSFC) Grant 62436009.

References

- [1] Peter Petersen. *Riemannian geometry*, volume 171. Springer, 2006.
- [2] Trieu H Trinh, Yuhuai Wu, Quoc V Le, He He, and Thang Luong. Solving olympiad geometry without human demonstrations. *Nature*, 625(7995):476–482, 2024.
- [3] Andrew Arana and Paolo Mancosu. On the relationship between plane and solid geometry. *The Review of Symbolic Logic*, 5(2):294–353, 2012.
- [4] Shang-Ching Chou, Xiao-Shan Gao, and Jing-Zhong Zhang. Automated generation of readable proofs with geometric invariants: I. multiple and shortest proof generation. *Journal of Automated Reasoning*, 17(3):325–347, 1996.
- [5] Jiaxin Zhang, Zhong-Zhi Li, Ming-Liang Zhang, Fei Yin, Cheng-Lin Liu, and Yashar Moshfeghi. Geoeval: Benchmark for evaluating llms and multi-modal models on geometry problem-solving. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 1258–1276, 2024.
- [6] Ke Wang, Juntong Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. Measuring multimodal mathematical reasoning with math-vision dataset. *Advances in Neural Information Processing Systems*, 37:95095–95169, 2024.

- [7] Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Yu Qiao, et al. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? In *European Conference on Computer Vision*, pages 169–186. Springer, 2024.
- [8] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.
- [9] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.
- [10] Tianxiang Sun, Xiaotian Zhang, Zhengfu He, Peng Li, Qinyuan Cheng, Xiangyang Liu, Hang Yan, Yunfan Shao, Qiong Tang, Shiduo Zhang, et al. Moss: An open conversational large language model. *Machine Intelligence Research*, 21(5):888–905, 2024.
- [11] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.
- [12] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.
- [13] Aixiu Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.
- [14] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.
- [15] OpenAI. Gpt-4v(ision) system card. 2023.
- [16] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24185–24198, 2024.
- [17] Xiao Wang, Guangyao Chen, Guangwu Qian, Pengcheng Gao, Xiao-Yong Wei, Yaowei Wang, Yonghong Tian, and Wen Gao. Large-scale multi-modal pre-trained models: A comprehensive survey. *Machine Intelligence Research*, 20(4):447–482, 2023.
- [18] Guoqing Ma, Haoyang Huang, Kun Yan, Liangyu Chen, Nan Duan, Shengming Yin, Changyi Wan, Ranchen Ming, Xiaoni Song, Xing Chen, et al. Step-video-t2v technical report: The practice, challenges, and future of video foundation model. *arXiv preprint arXiv:2502.10248*, 2025.
- [19] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 1(2), 2023.
- [20] Muhammad Usman Hadi, Rizwan Qureshi, Abbas Shah, Muhammad Irfan, Anas Zafar, Muhammad Bilal Shaikh, Naveed Akhtar, Jia Wu, Seyedali Mirjalili, et al. A survey on large language models: Applications, challenges, limitations, and practical usage. *Authorea Preprints*, 3, 2023.
- [21] Likang Wu, Zhi Zheng, Zhaopeng Qiu, Hao Wang, Hongchao Gu, Tingjia Shen, Chuan Qin, Chen Zhu, Hengshu Zhu, Qi Liu, et al. A survey on large language models for recommendation. *World Wide Web*, 27(5):60, 2024.
- [22] Juyong Jiang, Fan Wang, Jiasi Shen, Sungju Kim, and Sunghun Kim. A survey on large language models for code generation. *arXiv preprint arXiv:2406.00515*, 2024.
- [23] Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. A comprehensive overview of large language models. *arXiv preprint arXiv:2307.06435*, 2023.
- [24] Ben Mann, N Ryder, M Subbiah, J Kaplan, P Dhariwal, A Neelakantan, P Shyam, G Sastry, A Askell, S Agarwal, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 1:3, 2020.
- [25] Zijiang Liang, Yanjie Xu, Yifan Hong, Penghui Shang, Qi Wang, Qiang Fu, and Ke Liu. A survey of multimodal large language models. In *Proceedings of the 3rd International Conference on Computer, Artificial Intelligence and Control Engineering*, pages 405–409, 2024.

- [26] Can Cui, Yunsheng Ma, Xu Cao, Wenqian Ye, Yang Zhou, Kaizhao Liang, Jintai Chen, Juanwu Lu, Zichong Yang, Kuei-Da Liao, et al. A survey on multimodal large language models for autonomous driving. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 958–979, 2024.
- [27] Jiayang Wu, Wensheng Gan, Zefeng Chen, Shicheng Wan, and Philip S Yu. Multimodal large language models: A survey. In *2023 IEEE International Conference on Big Data (BigData)*, pages 2247–2256. IEEE, 2023.
- [28] OpenAI. Hello gpt-4o, 2024.
- [29] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.
- [30] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.
- [31] 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. *arXiv preprint arXiv:2310.02255*, 2023.
- [32] Pan Lu, Ran Gong, Shibiao Jiang, Liang Qiu, Siyuan Huang, Xiaodan Liang, and Song-Chun Zhu. Interps: Interpretable geometry problem solving with formal language and symbolic reasoning. *arXiv preprint arXiv:2105.04165*, 2021.
- [33] Jiaqi Chen, Jianheng Tang, Jinghui Qin, Xiaodan Liang, Lingbo Liu, Eric Xing, and Liang Lin. Geoqa: A geometric question answering benchmark towards multimodal numerical reasoning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 513–523, 2021.
- [34] Ming-Liang Zhang, Fei Yin, and Cheng-Lin Liu. A multi-modal neural geometric solver with textual clauses parsed from diagram. *arXiv preprint arXiv:2302.11097*, 2023.
- [35] Jiaqi Chen, Tong Li, Jinghui Qin, Pan Lu, Liang Lin, Chongyu Chen, and Xiaodan Liang. Unigeo: Unifying geometry logical reasoning via reformulating mathematical expression. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3313–3323, 2022.
- [36] Xiaofeng Wang, Yiming Wang, Wenhong Zhu, and Rui Wang. Do large language models truly understand geometric structures? *arXiv preprint arXiv:2501.13773*, 2025.
- [37] Jihan Yang, Shusheng Yang, Anjali W Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. *arXiv preprint arXiv:2412.14171*, 2024.
- [38] Ajit Singh. Meta llama 4: The future of multimodal ai. *Available at SSRN 5208228*, 2025.
- [39] Minjoon Seo, Hannaneh Hajishirzi, Ali Farhadi, Oren Etzioni, and Clint Malcolm. Solving geometry problems: Combining text and diagram interpretation. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1466–1476, 2015.
- [40] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9556–9567, 2024.
- [41] Runqi Qiao, Qiuna Tan, Guanting Dong, Minhui Wu, Chong Sun, Xiaoshuai Song, Zhuoma GongQue, Shanglin Lei, Zhe Wei, Miaoxuan Zhang, et al. We-math: Does your large multimodal model achieve human-like mathematical reasoning? *arXiv preprint arXiv:2407.01284*, 2024.
- [42] 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. *arXiv preprint arXiv:2502.20808*, 2025.
- [43] Liangyu Xu, Yingxiu Zhao, Jingyun Wang, Yingyao Wang, Bu Pi, Chen Wang, Mingliang Zhang, Jihao Gu, Xiang Li, Xiaoyong Zhu, et al. Geosense: Evaluating identification and application of geometric principles in multimodal reasoning. *arXiv preprint arXiv:2504.12597*, 2025.
- [44] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.

- [45] Renrui Zhang, Jiaming Han, Chris Liu, Aojun Zhou, Pan Lu, Yu Qiao, Hongsheng Li, and Peng Gao. Llama-adapter: Efficient fine-tuning of large language models with zero-initialized attention. In *The Twelfth International Conference on Learning Representations*, 2024.
- [46] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.
- [47] Xueyan Zhang, Jinman Zhao, Zhifei Yang, Yibo Zhong, Shuhao Guan, Linbo Cao, and Yining Wang. Uora: Uniform orthogonal reinitialization adaptation in parameter-efficient fine-tuning of large models. *arXiv preprint arXiv:2505.20154*, 2025.
- [48] Anthropic. The claude 3 model family: Opus, sonnet, haiku. <https://www.anthropic.com/claude-3-model-card>, 2024. Claude-3 Model Card.
- [49] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.
- [50] Yifan Du, Zikang Liu, Yifan Li, Wayne Xin Zhao, Yuqi Huo, Bingning Wang, Weipeng Chen, Zheng Liu, Zhongyuan Wang, and Ji-Rong Wen. Virgo: A preliminary exploration on reproducing o1-like mllm. *arXiv preprint arXiv:2501.01904*, 2025.
- [51] Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi Wang. Visual-rft: Visual reinforcement fine-tuning. *arXiv preprint arXiv:2503.01785*, 2025.
- [52] Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhui Chen. VI-rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. *arXiv preprint arXiv:2504.08837*, 2025.
- [53] Wenyi Xiao, Leilei Gan, Weilong Dai, Wanggui He, Ziwei Huang, Haoyuan Li, Fangxun Shu, Zhelun Yu, Peng Zhang, Hao Jiang, et al. Fast-slow thinking for large vision-language model reasoning. *arXiv preprint arXiv:2504.18458*, 2025.
- [54] Jingyi Zhang, Jiaying Huang, Huanjin Yao, Shunyu Liu, Xikun Zhang, Shijian Lu, and Dacheng Tao. R1-vl: Learning to reason with multimodal large language models via step-wise group relative policy optimization. *arXiv preprint arXiv:2503.12937*, 2025.
- [55] Huajie Tan, Yuheng Ji, Xiaoshuai Hao, Minglan Lin, Pengwei Wang, Zhongyuan Wang, and Shanghang Zhang. Reason-rft: Reinforcement fine-tuning for visual reasoning. *arXiv preprint arXiv:2503.20752*, 2025.
- [56] Yihe Deng, Hritik Bansal, Fan Yin, Nanyun Peng, Wei Wang, and Kai-Wei Chang. Openvlthinker: An early exploration to complex vision-language reasoning via iterative self-improvement. *arXiv preprint arXiv:2503.17352*, 2025.
- [57] Yiyao Yu, Yuxiang Zhang, Dongdong Zhang, Xiao Liang, Hengyuan Zhang, Xingxing Zhang, Ziyi Yang, Mahmoud Khademi, Hany Awadalla, Junjie Wang, et al. Chain-of-reasoning: Towards unified mathematical reasoning in large language models via a multi-paradigm perspective. *arXiv preprint arXiv:2501.11110*, 2025.
- [58] Jiahui Gao, Renjie Pi, Jipeng Zhang, Jiacheng Ye, Wanjuan Zhong, Yufei Wang, Lanqing Hong, Jianhua Han, Hang Xu, Zhenguo Li, et al. G-llava: Solving geometric problem with multi-modal large language model. *arXiv preprint arXiv:2312.11370*, 2023.
- [59] Zhenwen Liang, Tianyu Yang, Jipeng Zhang, and Xiangliang Zhang. Unimath: A foundational and multimodal mathematical reasoner. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7126–7133, 2023.
- [60] Zhong-Zhi Li, Ming-Liang Zhang, Fei Yin, and Cheng-Lin Liu. Lans: A layout-aware neural solver for plane geometry problem. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 2596–2608, 2024.
- [61] Jo-Ku Cheng, Zeren Zhang, Ran Chen, Jingyang Deng, Ziran Qin, and Jinwen Ma. Geouni: A unified model for generating geometry diagrams, problems and problem solutions. *arXiv preprint arXiv:2504.10146*, 2025.
- [62] William B Cavnar, John M Trenkle, et al. N-gram-based text categorization. In *Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval*, volume 161175, page 14. Ann Arbor, Michigan, 1994.

- [63] 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. *arXiv preprint arXiv:2411.00836*, 2024.
- [64] Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3828–3850, 2024.
- [65] Pan Zhang, Xiaoyi Dong, Yuhang Zang, Yuhang Cao, Rui Qian, Lin Chen, Qipeng Guo, Haodong Duan, Bin Wang, Linke Ouyang, et al. Internlm-xcomposer-2.5: A versatile large vision language model supporting long-contextual input and output. *arXiv preprint arXiv:2407.03320*, 2024.
- [66] Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang Ma, Chengyue Wu, Bingxuan Wang, et al. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. *arXiv preprint arXiv:2412.10302*, 2024.
- [67] Wenhao Shi, Zhiqiang Hu, Yi Bin, Junhua Liu, Yang Yang, See Kiong Ng, Lidong Bing, and Roy Lee. Math-llava: Bootstrapping mathematical reasoning for multimodal large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 4663–4680, 2024.
- [68] Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv preprint arXiv:2407.07895*, 2024.
- [69] Mistral Team. Mistral small 3.1, Mar 2025.
- [70] Omkar Thawakar, Dinura Dissanayake, Ketan More, Ritesh Thawkar, Ahmed Heakl, Noor Ahsan, Yuhao Li, Mohammed Zumri, Jean Lahoud, Rao Muhammad Anwer, et al. Llamav-o1: Rethinking step-by-step visual reasoning in llms. *arXiv preprint arXiv:2501.06186*, 2025.
- [71] Guowei Xu, Peng Jin, Li Hao, Yibing Song, Lichao Sun, and Li Yuan. Llava-o1: Let vision language models reason step-by-step. *arXiv preprint arXiv:2411.10440*, 2024.
- [72] Haozhan Shen, Peng Liu, Jingcheng Li, Chunxin Fang, Yibo Ma, Jiajia Liao, Qiaoli Shen, Zilun Zhang, Kangjia Zhao, Qianqian Zhang, et al. Vlm-r1: A stable and generalizable r1-style large vision-language model. *arXiv preprint arXiv:2504.07615*, 2025.
- [73] Yi Yang, Xiaoxuan He, Hongkun Pan, Xiyan Jiang, Yan Deng, Xingtao Yang, Haoyu Lu, Dacheng Yin, Fengyun Rao, Minfeng Zhu, et al. R1-onevision: Advancing generalized multimodal reasoning through cross-modal formalization. *arXiv preprint arXiv:2503.10615*, 2025.
- [74] Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models. *arXiv preprint arXiv:2503.06749*, 2025.
- [75] Yichen Wei, Yi Peng, Xiaokun Wang, Weijie Qiu, Wei Shen, Tianyidan Xie, Jiangbo Pei, Jianhao Zhang, Yunzhuo Hao, Xuchen Song, et al. Skywork r1v2: Multimodal hybrid reinforcement learning for reasoning. *arXiv preprint arXiv:2504.16656*, 2025.
- [76] Qwen Team. Qvq: To see the world with wisdom, December 2024.
- [77] Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Hel-yar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.
- [78] Zhong-Zhi Li, Duzhen Zhang, Ming-Liang Zhang, Jiabin Zhang, Zengyan Liu, Yuxuan Yao, Haotian Xu, Junhao Zheng, Pei-Jie Wang, Xiuyi Chen, et al. From system 1 to system 2: A survey of reasoning large language models. *arXiv preprint arXiv:2502.17419*, 2025.
- [79] Zhiyuan Hu, Yuliang Liu, Jinman Zhao, Suyuchen Wang, Yan Wang, Wei Shen, Qing Gu, Anh Tuan Luu, See-Kiong Ng, Zhiwei Jiang, et al. Longrecipe: Recipe for efficient long context generalization in large language models. *arXiv preprint arXiv:2409.00509*, 2024.
- [80] Jinman Zhao and Xueyan Zhang. Large language model is not a (multilingual) compositional relation reasoner. In *First Conference on Language Modeling*, 2024.

- [81] Zhongzhi Li, Ming-Liang Zhang, Pei-Jie Wang, Jian Xu, Rui-Song Zhang, Yin Fei, Zhi-Long Ji, Jin-Feng Bai, Zhen-Ru Pan, Jiaxin Zhang, et al. Cmmath: A chinese multi-modal math skill evaluation benchmark for foundation models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 2690–2726, 2025.
- [82] Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, et al. Do not think that much for $2+3=?$ on the overthinking of o1-like llms. *arXiv preprint arXiv:2412.21187*, 2024.
- [83] Yue Wang, Qiuzhi Liu, Jiahao Xu, Tian Liang, Xingyu Chen, Zhiwei He, Linfeng Song, Dian Yu, Juntao Li, Zhuosheng Zhang, et al. Thoughts are all over the place: On the underthinking of o1-like llms. *arXiv preprint arXiv:2501.18585*, 2025.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: see Section 1

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: see Section 5

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[NA\]](#)

Justification: Our paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: see Section 4.1

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We will provide complete code and documentation

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: see Section 4.1 and Section 3

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: For large-scale evaluations, we did not conduct error analysis. However, for 300 small-scale evaluations such as hallucination rejection ability assessment, we classify the errors into five categories, and analyze their distribution(see Section 4.3).

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: see Section 4.1

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: We comply with the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: see Section 5

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The paper poses no such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The MLLMs we evaluated are all open-source by the author and the dataset is created by us.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset’s creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [\[Yes\]](#)

Justification: : We provide the details of our dataset in Section 3

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [\[Yes\]](#)

Justification: see Section F

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [\[Yes\]](#)

Justification: see Section F

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: Our job is to evaluate the performance of MLLMs in the field of solid geometry, and we described the purpose and methods of use in the article

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.

Appendix

A More Detailed Statistics about SOLIDGEO

In this chapter, we will introduce more statistics about our SOLIDGEO. As shown in 2, every question in SOLIDGEO contains at least one image input, annotated with difficulty levels and fine-grained solid geometry categories.

A.1 Distribution of Text Length

Questions in SOLIDGEO are presented in English or Chinese. As shown in Table 1, the longest question in SOLIDGEO spans 679 words, with an average length of 77.2 words. Figure 12 further illustrates the distribution of text lengths, highlighting the diversity of SOLIDGEO. The length of English and Chinese questions is counted in words and Chinese characters, respectively.

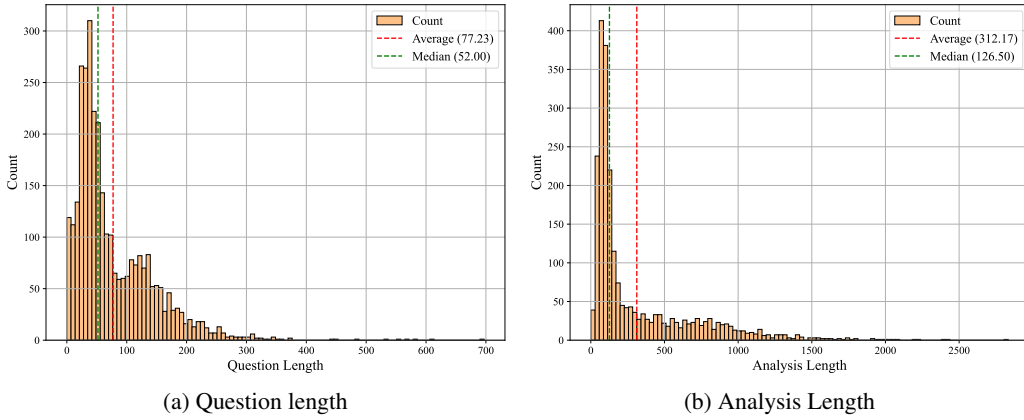


Figure 8: The distribution of the length per question in SOLIDGEO.

A.2 Distribution of Image Number per question

As shown in Figure 9, the majority of questions in the SOLIDGEO dataset (74.5%) are accompanied by a single image. The remaining questions are associated with multiple images, with the number of images ranging from two to eight.

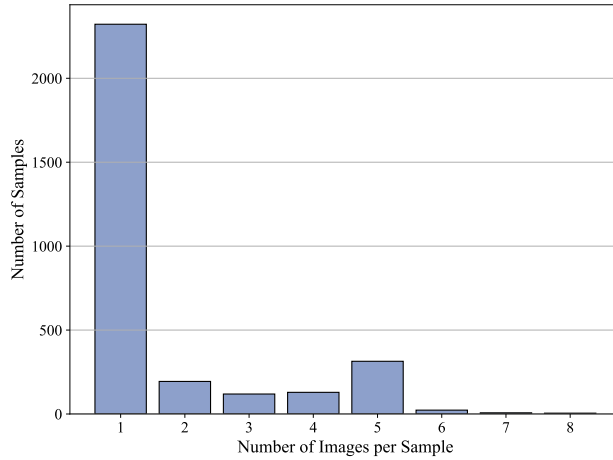


Figure 9: The distribution of the number of images per question in SOLIDGEO.

A.3 Distribution of Solid Geometry Subjects

Figure 10 presents the distribution of subject categories within the solid geometry portion of the SOLIDGEO. The questions are categorized into eight distinct geometry-related topics. Among them, the most dominant subject is *Spatial Metric Relations*, accounting for 857 samples, followed by *Composite Solid Structures* (794) and *Multi-view Projection* (699).

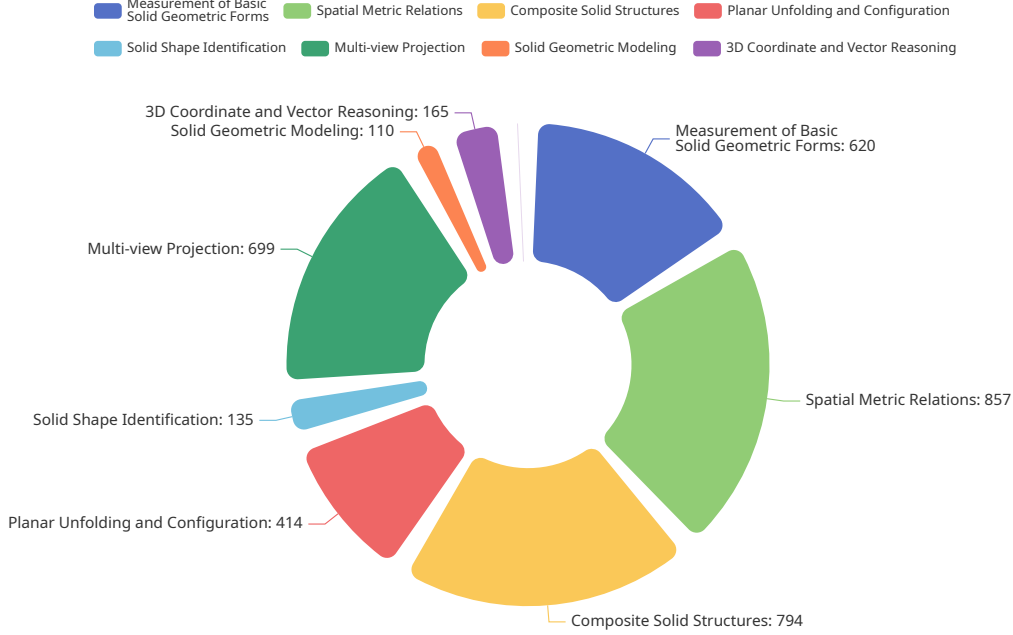


Figure 10: The distribution of 8 fine-grained solid geometry subjects in SOLIDGEO.

Measurement of Basic Solid Geometric Forms also comprises a substantial portion with 620 samples. Mid-sized categories include *Planar Unfolding and Configuration* (414) and *3D Coordinate and Vector Reasoning* (165). Notably, even the smallest categories—*Solid Shape Identification* (135) and *Solid Geometric Modeling* (110)—still contain more than 100 samples each, which is sufficient to ensure evaluation reliability and statistical stability across all sub-domains.

This distribution demonstrates that while the dataset exhibits a certain degree of imbalance, all categories are adequately represented to support meaningful performance comparisons for models across different solid geometry skills.

A.4 Distribution of Sources

Figure 11 illustrates the composition of our dataset in terms of source origin. The dataset integrates solid geometry problems from seven distinct benchmarks, including MathVerse (11.47%), MathVision (7.84%), GeoEval (2.41%), DynaMath (4.49%), OlympiadBench (24.93%), and CMMaTH (4.66%). Notably, 44.2% of the samples are newly collected and curated solid geometry problems, which significantly expand the diversity and coverage of our dataset.

To ensure the quality and uniqueness of each problem, we performed rigorous filtering and deduplication across all sources. Redundant or low-quality items were removed, and only samples with well-defined geometric settings, unambiguous language, and reliable answer annotations were retained. This refinement process guarantees that each question in the final SolidGeo dataset is both high-quality and non-overlapping, providing a consistent and trustworthy benchmark for evaluating spatial reasoning capabilities. For the specific process, see Appendix D.

This broad integration of sources—combined with strict curation—ensures that the dataset covers a wide range of solid geometry problem types while maintaining a high standard of clarity, correctness, and diversity.

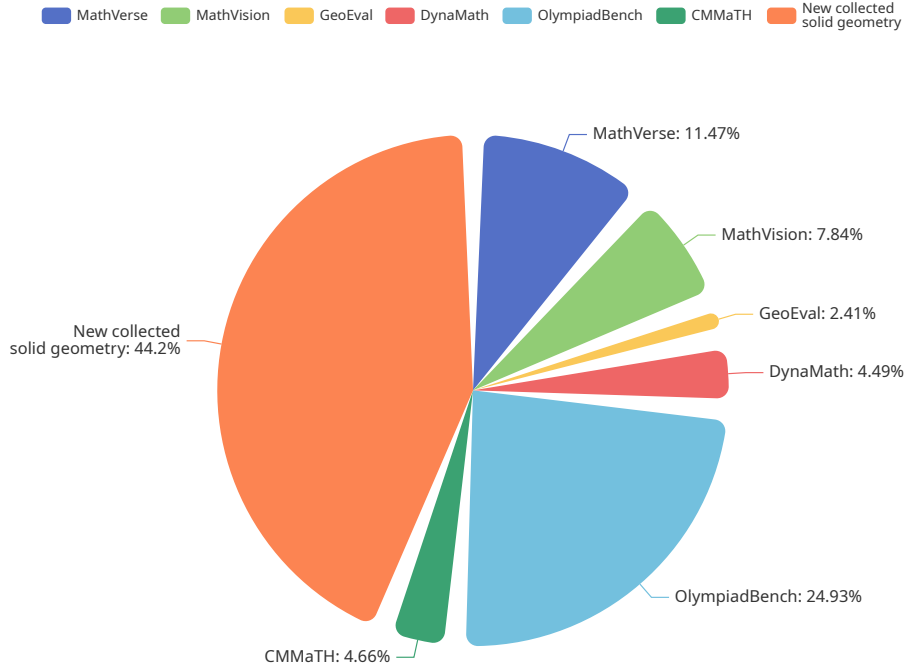


Figure 11: The distribution of different sources in SOLIDGEO.

B Introduction of our Fine-grained Solid Geometry Subjects

In this section, we introduce the eight fine-grained subject categories defined in our solid geometry benchmark. These categories are designed to reflect the diverse reasoning skills required for real-world spatial understanding tasks. Rather than treating solid geometry as a monolithic domain, we present a structured taxonomy that decomposes the problem space into conceptually distinct yet complementary components. Figure 22 to Figure 29 illustrate representative image examples corresponding to each category.

Our classification is based on both the cognitive processes involved (e.g., spatial metric reasoning, 3D transformation) and the structural features of the geometric objects (e.g., solid composition, projection, unfolding). This allows for more interpretable performance diagnostics and supports targeted model evaluation. The eight categories span a broad range of solid geometry competencies, ensuring coverage of foundational as well as advanced visual-spatial skills.

1. Measurement of Solid Geometric Forms. This category focuses on fundamental formula-based computations involving standard three-dimensional shapes such as cubes, cuboids, cylinders, cones, spheres, and regular polyhedrons. Problems typically involve direct application of geometric formulas for volume, surface area, or edge length based on given dimensions. These problems often serve as entry points in spatial reasoning tasks and require little to no shape manipulation or visualization beyond understanding the formula’s variables. *Example task:* Compute the volume of a right circular cone with radius 4 cm and height 9 cm.

2. Solid Shape Identification. This subject class targets the recognition and naming of 3D geometric solids or their components (such as faces, edges, or vertices), based on visual or structural cues. The focus lies in spatial visualization rather than computation. Problems typically include diagrams and ask students to match shapes to names or count features (e.g., number of faces). *Example task:* Identify the name of a 3D shape with 8 faces and 12 edges, or label the edges in a net diagram.

3. Spatial Metric Relations. This category involves reasoning about geometric measurements in 3D space, particularly those that rely on theorems and spatial relationships such as distances, angles, and relative positions. Solutions typically require applying properties of perpendicularity, intersection, or symmetry. These problems demand an understanding of spatial configurations and their mathematical implications. *Example task:* Determine the angle between a diagonal of a cube and one of its faces, or calculate the shortest distance from a point to a plane.

4. Multi-view Projection. This class includes problems involving orthographic projections and their interpretation. It emphasizes the ability to switch between 2D projections (front, top, side views) and 3D spatial understanding. Learners must mentally reconstruct 3D solids from multiple projections or generate accurate views based on a given model. *Example task:* Infer the 3D object described by its top and front views, or draw the side view of a given solid.

5. Planar Unfolding and Configuration. This subject covers the analysis of how 3D solids unfold into 2D nets and vice versa. It includes determining valid unfoldings, reasoning about face connectivity, and reconstructing solid forms from nets. It also addresses spatial folding logic and pathfinding over surfaces (e.g., shortest path problems on a cube’s surface). *Example task:* Complete the missing face in the net of a cube, or determine if a given net corresponds to a regular dodecahedron.

6. Composite Solid Structure. Problems in this category deal with complex solids formed by combining, intersecting, or modifying standard geometric shapes. It requires understanding how operations such as union, subtraction, and intersection affect volume and surface area. This type reflects realistic modeling scenarios involving multi-body systems. *Example task:* Find the volume of the intersection between a sphere and a cube, or compute the surface area of a cylinder with a conical cutout.

7. 3D Coordinate and Vector Reasoning. This subject category utilizes algebraic methods to solve geometric problems in three-dimensional coordinate systems. It includes vector-based calculations of angles, distances, projections, and normal vectors. Problems often translate geometric relationships into equations and exploit coordinate geometry principles. *Example task:* Calculate the angle between two vectors or find the shortest distance from a point to a line in space.

8. Solid Geometric Modeling. This is the most application-oriented category, featuring problems that simulate real-world use of solid geometry in architecture, engineering, and design. It involves optimizing parameters, validating geometric constraints, or modeling complex surfaces. This class bridges pure geometry with practical problem-solving. *Example task:* Design an optimized water tank shape with minimal surface area or verify whether a given structure satisfies balance and symmetry constraints.

C Comparison

As shown in Section 3.3, we have already compared SOLIDGEO with existing datasets in terms of coverage and task diversity. In this section, we further extend the comparison from two additional perspectives: (1) the average length of questions, and (2) the performance of the best-performing model on each dataset.

As shown in Figure 12, our SOLIDGEO exhibits the longest average question length among all compared benchmarks, reaching 77.2 words. This is significantly higher than datasets such as MathVista (15.6), GeoQA (37.1), and MMMU-MATH (40.8). The increased question length in SOLIDGEO directly reflects the need for more precise problem understanding and more complex reasoning processes, which distinguishes it from benchmarks that primarily involve shorter and more formulaic tasks. Moreover, the right subfigure highlights a striking performance gap: despite the state-of-the-art models achieving over 80% accuracy on other benchmarks, their best performance on SOLIDGEO drops sharply to 49.5%. This further validates the higher reasoning difficulty and challenge posed by our benchmark. It underscores the necessity of more robust spatial understanding and compositional reasoning to succeed on SOLIDGEO.

D More Detailed Construction of SOLIDGEO

To further clarify the construction process of SOLIDGEO, this section provides additional details on the data collection strategy, filtering pipeline, and annotation workflow introduced in Section 3. These

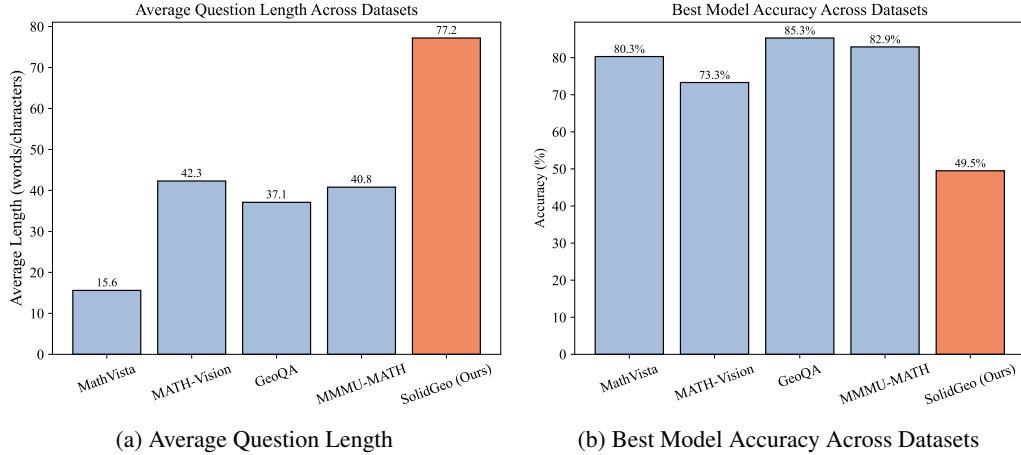


Figure 12: Average question length and best-performing model accuracy across datasets

components jointly ensure that each problem in SOLIDGEO is high-quality, topically relevant, and uniquely represented. We first describe the hybrid data sourcing approach, followed by our four-stage filtering procedure and the multi-phase annotation protocol used to assign difficulty levels and subject categories.

D.1 Data Collection Details

To construct SOLIDGEO, we adopted a hybrid data collection strategy that combines both existing benchmarks and newly collected real-world data. Specifically, we extracted solid geometry problems from six publicly available multimodal math datasets: **MathVerse** [7], **MATH-Vision** [6], **DynaMath** [63], **GeoEval** [5], **CMMaTH** [81], and **OlympiadBench** [64], based on their official topic annotations or keyword filtering. These samples account for 1,737 problems in our final dataset.

To enrich the dataset with more diverse and realistic problem styles, we further collected **1,376 new solid geometry problems** from K–12 education sources. Most of these were obtained from the Zujian platform⁴, a widely used educational resource that provides large-scale math problems in PDF format. We employed the Mathpix OCR API⁵ to extract structured problem components, including question texts, diagrams, and answer analyses.

The initial candidate pool contained 10,932 problems after solid geometry keyword filtering, which were later filtered and refined through a four-stage pipeline (see Section 3.2). These newly collected samples contribute 44.2% to the final dataset, significantly enhancing its diversity in language, format, and reasoning complexity.

D.2 Data Filtering Details

Recap of the Four-Stage Filtering. As described in Section 3.2, we employed a systematic four-stage filtering pipeline: (1) structural filtering to ensure multimodal completeness, (2) image quality filtering using OpenCV, (3) semantic filtering using large model-based topic classification, and (4) n-gram-based deduplication across datasets.

Image Quality Filtering with OpenCV. To remove blurry or low-resolution diagrams, we computed the variance of the Laplacian for each image as a sharpness metric. A conservative threshold of **1000** was adopted; images with a variance below this threshold were excluded from the dataset. This ensured that all retained images contain sufficiently clear visual information to support accurate spatial reasoning.

Cross-Set Deduplication via Post-hoc n-gram Similarity Analysis. Instead of applying rule-based filtering, we conducted a comprehensive word-level n -gram similarity analysis between our newly

⁴<https://zujian.xkw.com/>

⁵<https://mathpix.com/convert>

collected `solidgeo` problems and existing benchmark samples. Specifically, we computed the cosine similarity between each `solidgeo` question and all external questions based on their n -gram count vectors, with $n = 2, 3, 4, 5, 8$.

Let $S_{i,j}$ denote the cosine similarity computed between the word-level n -gram feature vectors of the i -th `solidgeo` question and the j -th existing question. The average cross-source similarity is then calculated as:

$$\text{Average Similarity} = \frac{1}{N_{\text{solidgeo}} \times N_{\text{existing}}} \sum_{i=1}^{N_{\text{solidgeo}}} \sum_{j=1}^{N_{\text{existing}}} S_{i,j}$$

We observed that the average similarity drops rapidly as n increases—e.g., from 2.53% at $n = 2$ to near-zero at $n = 8$, validating the lexical independence of `SOLIDGEO`. This fully demonstrates that there is no overlap between our newly collected `solidgeo` and existing samples. The detailed statistics of word-level n -gram similarity between `SolidGeo` and existing sources are summarized in Table 5.

Table 5: Word-level n -gram Average Similarity Between `SolidGeo` and Other Sources

Source Pair	2-grams	3-grams	4-grams	5-grams	8-grams
SolidGeo vs Others	2.53%	0.60%	0.29%	0.10%	0.00%

D.3 Data Annotation Protocol

Difficulty Labeling via Prompted Model Voting. To label question difficulty, we constructed a tailored prompt based on expert-provided heuristics. This prompt guides LLMs to judge the complexity of geometric reasoning required (e.g., direct formula use vs. multi-step deduction). Three advanced MLLMs—**GPT-4o**, **Claude-3.7-Sonnet**, and **Qwen-VL-Max**—were queried in parallel, and a majority vote was used to assign a difficulty level from 1 (easy) to 3 (hard). For cases where model votes disagreed, the final label was manually assigned by experienced annotators. Detailed prompt for difficulty level is shown in Table 6.

Subject Categorization Based on Expert Taxonomy. To define meaningful subject categories, we invited two domain experts in geometry education to review a representative subset of our data. Based on their analysis of geometry curricula and problem-solving cognitive patterns, they proposed an 8-category taxonomy: *Measurement of Solid Geometric Forms*, *Solid Shape Identification*, *Spatial Metric Relations*, *Multi-view Projection*, *Planar Unfolding and Configuration*, *Composite Solid Structure*, *3D Coordinate and Vector Reasoning* and *Solid Geometric Modeling*. These categories aim to reflect essential dimensions of solid geometry reasoning, ensuring interpretability and task-specific performance analysis. See Table 6 for detailed prompt.

Given the taxonomy, the same three MLLMs (GPT-4o, Claude-3.7-Sonnet, and Qwen-VL-Max) were prompted to predict the subject of each problem. Again, majority voting was used to assign a label, and human annotators resolved any disagreements.

Annotator Review via Interactive Annotation Platform. After automated labeling, we developed an internal web-based interface to visualize problem texts, diagrams, and model predictions. Using this tool, three trained annotators independently reviewed all 3,113 samples. Each problem was checked for both subject and difficulty correctness. In cases of disagreement, consensus was reached through discussion. This final human verification stage ensures label consistency and semantic integrity across the dataset.

E Generated Token Distribution

Figures 13–16 show the token distributions of System-1 models, while Figures 17–21 correspond to System-2 models. In general, System-2 models tend to produce longer outputs. Across most models, the generated token distributions exhibit long-tailed characteristics.

Notably, **Gemini-2.5-pro** and **OpenAI-o1** demonstrate the most evident long-tail distributions, suggesting their ability to dynamically adjust output length based on problem complexity. Such adaptive behavior is crucial for efficient reasoning and future inference scalability.

In contrast, models like **Skywork-R1V2-38B** and **Vision-R1-7B** show highly concentrated output lengths, indicating a lack of dynamic adjustment in response to varying problem difficulty—i.e., these models tend to *overthinking* [82] even for simpler questions.

Interestingly, although **R1-Onevision-7B** is a System-2 model, its token distribution is relatively narrow. This may be due to model size limitations that prevent the expression of extended multi-step reasoning, indicating possible *underthinking* [83] behavior.

Overall, both overthinking and underthinking behaviors are emerging concerns in current models. Developing mechanisms to dynamically adapt reasoning depth based on problem complexity is essential for achieving robust and efficient multimodal reasoning.

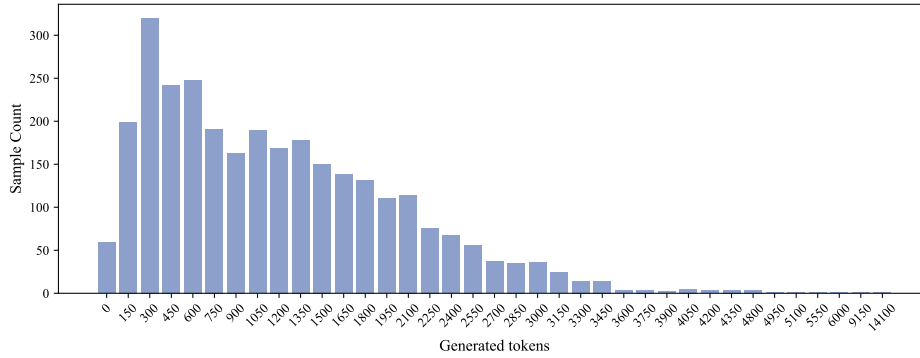


Figure 13: The generated token distribution of Gemini-2.5-pro on SOLIDGEO.

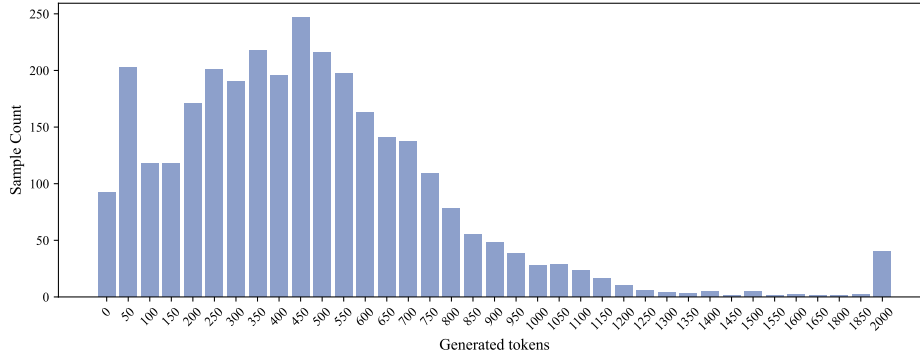


Figure 14: The generated token distribution of Internv13-78B on SOLIDGEO.

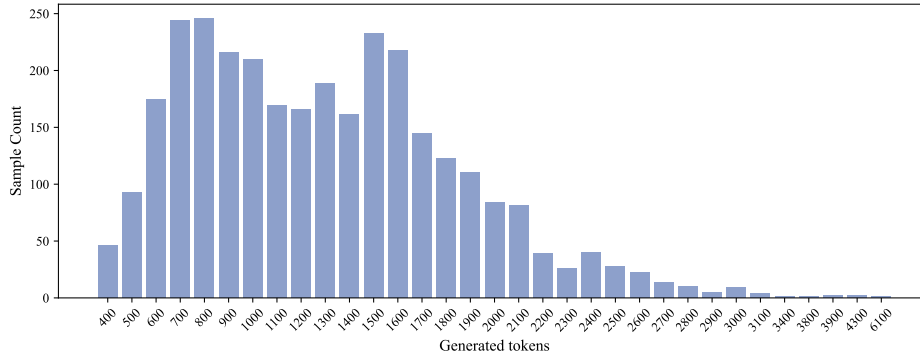


Figure 15: The generated token distribution of GPT-4o on SOLIDGEO.

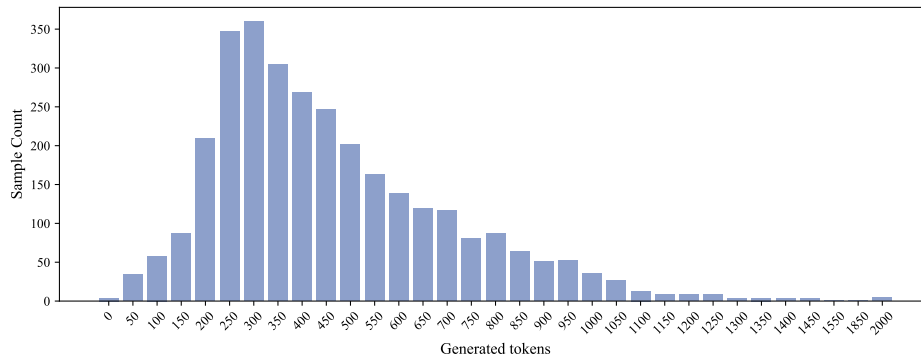


Figure 16: The generated token distribution of Qwen2.5-VL-72B on SOLIDGEO.

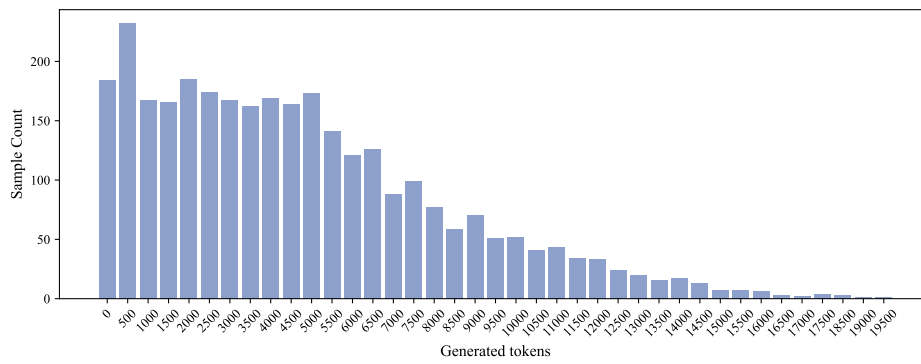


Figure 17: The generated token distribution of OpenAI-o1 on SOLIDGEO.

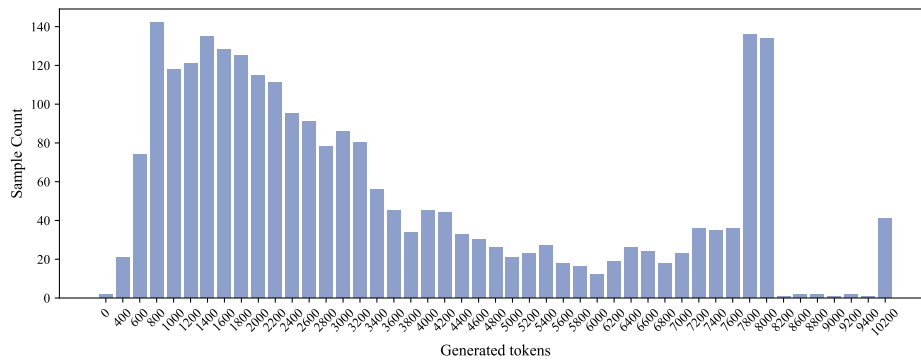


Figure 18: The generated token distribution of QvQ-72B on SOLIDGEO.

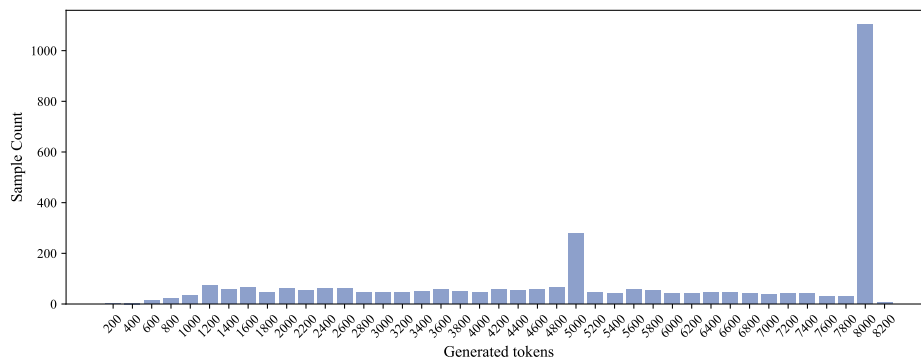


Figure 19: The generated token distribution of Skywork-R1V2-38B on SOLIDGEO.

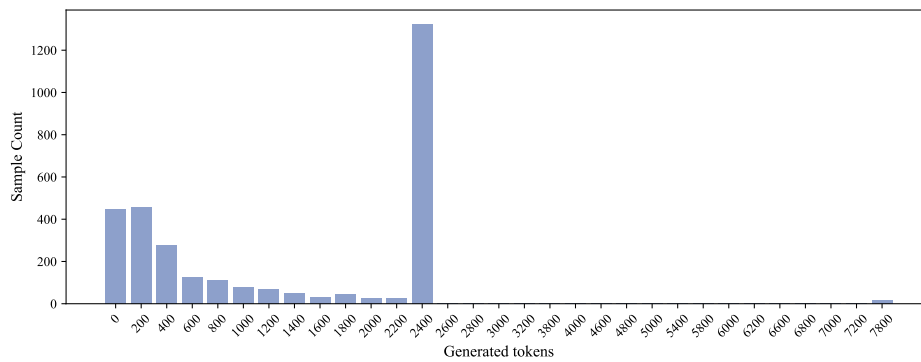


Figure 20: The generated token distribution of Vision-R1-7B on SOLIDGEO.

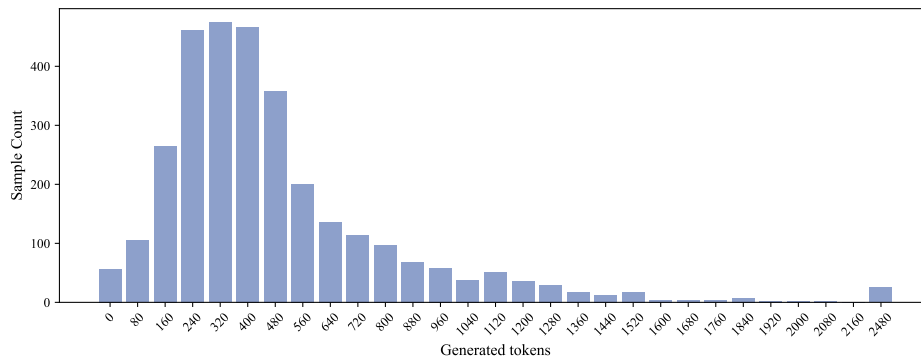


Figure 21: The generated token distribution of R1-Onevision-7B on SOLIDGEO.

Table 6: Prompt Design for Problem Classification and Annotation

System Prompt
You are a 3D geometry problem classification expert. Your task is to analyze solid geometry problems and output structured annotations. You should extract geometric elements, spatial relationships, calculation targets, classify the problem type, and assign a reasoning complexity level. Respond in JSON format, and do not include explanations.
User Prompt for Difficulty Level Classification
Please assign a difficulty level (1: easy, 2: medium, 3: hard) to the following solid geometry problem. Use the following criteria: - Level 1: Basic Execution — Problems in this category require the direct application of a single formula or a well-known geometric rule. The solution path is straightforward with minimal intermediate steps. - Level 2: Multi-Step Reasoning — Problems involve several computational steps or the construction of simple proofs. They require organizing multiple procedures in a logical sequence and understanding intermediate relationships between geometric elements. - Level 3: Creative Spatial Reasoning — Problems in this level necessitate advanced spatial visualization and innovative approaches. Instead of a standard formula, solvers must explore geometric relationships and often create non-obvious solution strategies. Return only the level number.
User Prompt for Category Classification
You are a geometry expert. Given a solid geometry problem, classify it into one or more of the following categories based on its core reasoning focus: 1. Measurement of Basic Solid Geometric Forms. Key Features: Formula-driven computations for regular polyhedrons and revolution solids (e.g., Platonic solids, cylinders, cones, spheres). <i>Example: "Calculate the volume of a regular octahedron with 3cm edge length".</i> 2. Solid Shape Identification. Key Features: Visual recognition of geometric solids or their components (faces/edges/vertices) without quantitative analysis. <i>Example: "Identify the name of the 3D shape shown in the diagram".</i> 3. Spatial Metric Relations. Key Features: Calculations involving spatial relationships (point-to-plane distances, line-to-surface angles) using geometric theorems. <i>Example: "Determine the shortest distance from a point to a given plane in 3D space".</i> 4. Multi-view Projection Analysis. Key Features: Conversion between 2D orthographic projections (front/top/side views) and 3D reconstructions. <i>Example: "Reconstruct a solid model from its front and top views".</i> 5. Planar Unfolding and 3D Configuration. Key Features: Analysis of surface unfoldings (nets) and spatial folding patterns, including path optimization. <i>Example: "Complete the missing face in a dodecahedron's net".</i> 6. Composite Solid Structural Analysis. Key Features: Solving problems involving intersecting/cut/combined solids (e.g., Boolean operations on shapes). <i>Example: "Find the intersection volume of a sphere and a cube".</i> 7. 3D Coordinate and Vector Applications. Key Features: Algebraic solutions using 3D coordinate systems, vectors, or parametric equations. <i>Example: "Compute the angle between two spatial vectors".</i> 8. Engineering Solid Geometric Modeling. Key Features: Real-world geometric optimization and structural validation (e.g., architectural/mechanical design). <i>Example: "Optimize the curved surface design of a water storage tank".</i> Output the predicted categories as a list. Use multiple labels if necessary.

F Annotator and Human Level

To ensure high-quality and reliable annotations, we assembled a team comprising two domain experts and several trained annotators. The two experts possess extensive experience in the field of solid geometry and played a key role in designing the annotation schema, defining the eight fine-grained subject categories, and establishing the difficulty level criteria.

All annotators are graduate students with strong mathematical backgrounds and spatial reasoning ability. They received targeted training and annotated the data strictly following expert-defined protocols. After the model-assisted initial annotation stage, each sample was independently reviewed by three annotators to verify the assigned subject category and difficulty level. This rigorous process ensures the consistency and credibility of all annotations used in our benchmark. Specifically, the expert-provided guidelines instructed annotators to classify each problem based on its core reasoning focus into one or more of the eight fine-grained solid geometry categories, such as spatial metrics, projection transformations, or coordinate modeling, and to assign a difficulty level according to a three-tier rubric: Level 1 for direct application of geometric formulas, Level 2 for multi-step deduction involving intermediate reasoning, and Level 3 for complex spatial reasoning or geometric abstraction. Annotators were asked to ignore irrelevant linguistic complexity, focus on reasoning-relevant elements, and flag ambiguous cases for expert arbitration to ensure labeling quality and inter-annotator consistency.

To establish a human performance baseline, we recruited 60 high school students and asked them to complete the questions independently, with each student assigned their own section of the questions without external assistance. Their responses were evaluated using the same criteria as the model. The results provide meaningful references for understanding the relative strengths and weaknesses of current multimodal models in solid geometry reasoning tasks.

We confirm that our human baseline study involving high school students was conducted under a formal and responsible ethical protocol. The study adhered to the following procedures:

- **Informed Consent:** Consent was obtained from the legal guardians of all student participants, alongside assent from the students themselves.
- **Voluntary Participation:** Participation was entirely voluntary, and students retained the right to withdraw at any time without penalty.
- **Privacy Protection:** All collected data was fully anonymized; no personal information was recorded or retained.
- **Compensation:** Participating students received a set of school-approved mathematics learning materials and educational tools. These items were selected to support students’ academic growth and were appropriate for their grade level.

G Evaluation Details

For open-source models, all experiments—except for LLaMA-4-Maverick-17B-128E, which is accessed via API—are conducted locally on A800 GPUs. This setup ensures consistent hardware conditions for evaluating most models.

G.1 Prompt for Response Generation

To ensure the model provides accurate responses, we design distinct CoT and 2-shot prompts tailored for multiple-choice, single-step, and multi-step free-form questions. The original prompt directly instructs the model to generate the final answer without intermediate reasoning. Detailed information can be found in Table 7, Table 8, Table 9.

G.2 Prompt for Answer Evaluation

Our evaluation is conducted using the Deepseek API. For the evaluation of multiple-choice, single-step, and multi-step free-form questions, different prompts are designed to ensure accuracy in answer extraction and assessment. We first use the Deepseek API to extract the model’s answers and then it

compares the extracted answers with the ground truth to determine the correctness of the answers. The specific prompts are shown in 10 below.

G.3 Model Details

All experiments are conducted using models configured with a temperature of 0.2 for text generation. For System-1 models, we set the max_tokens limit to 4096, while for System-2 models—designed for more complex reasoning tasks—the limit is extended to 16,384. Comprehensive details regarding the models utilized in the evaluation are presented in Table 11.

Original
You are an assistant for solving math problems. Your input consists of a math question and images, give your answer directly, without any intermediate steps.
CoT (Chain of Thought)
You are an assistant for solving math problems. Your input consists of a math question and images. Your task is to output the solution steps and the answer. The output format should be a step-by-step approach. Each question is multiple choice with one correct answer. Your final answer must be one of A, B, C, or D, and it should be placed within {}. For example: {A}, {B}, {C}, or {D}.
CoT with 2-shot
Example 1: Question: If a triangle has two sides of length 3 and 4, what is the length of the hypotenuse? A.10 B.8 C.5 D.4 Answer: Step 1 (Mathematical theorem used: Pythagorean theorem): The Pythagorean theorem states that in a right triangle, the square of the hypotenuse is equal to the sum of the squares of the other two sides. The formula is: $c^2 = a^2 + b^2$, where a and b are the legs, and c is the hypotenuse. Step 2 (Substitute the known values): Given $a = 3$ and $b = 4$. Substituting these values into the formula: $c^2 = 3^2 + 4^2 = 9 + 16 = 25$ Step 3 (Calculate the hypotenuse): Taking the square root gives: $c = \sqrt{25} = 5$ Answer: {C} Example 2: Question: In the right triangle ABC, AB is perpendicular to BC. It is known that AC=5 and AB=4. Find the area of the right triangle. A.20 B.10 C.5 D.6 Answer: Step 1 (Mathematical theorem used: Pythagorean theorem): We first use the Pythagorean theorem to find the length of BC. The formula is: $AC^2 = AB^2 + BC^2$, where AC is the hypotenuse, and AB and BC are the legs. Step 2 (Substitute the known values): Given $AC = 5$ and $AB = 4$. Substituting these values: $5^2 = 4^2 + BC^2 \implies 25 = 16 + BC^2$ Step 3 (Solve for BC): $BC^2 = 25 - 16 = 9 \implies BC = \sqrt{9} = 3$ Step 4 (Calculate the area): The area of the right triangle is given by $\frac{1}{2} \times AB \times BC$. Substituting the known values: Area = $\frac{1}{2} \times 4 \times 3 = 6$ Answer: {D} Your final answer must be one of A, B, C, or D, and it should be placed within {}

Table 7: The prompts used for choice questions in the evaluation for response generation.

Original Prompt
You are an assistant for solving math problems. Your input consists of a math question and images. Give your answer directly, without any intermediate steps.
CoT (Chain of Thought)
You are an assistant for solving math problems. Your input consists of a math question and images. Your task is to output the solution steps and the answer. The output format should be a step-by-step approach.
CoT with 2-shot
Example 1: Question: If a triangle has two sides of length 3 and 4, what is the length of the hypotenuse? Answer: Step 1: (Mathematical theorem used: Pythagorean theorem): The Pythagorean theorem states that in a right triangle, the square of the hypotenuse is equal to the sum of the squares of the other two sides. The formula is: $c^2 = a^2 + b^2$, where a and b are the legs, and c is the hypotenuse. Step 2: (Substitute the known values): Given $a = 3$ and $b = 4$. Substituting these values into the formula: $c^2 = 3^2 + 4^2 = 9 + 16 = 25$. Step 3: (Calculate the hypotenuse): Taking the square root gives: $c = \sqrt{25} = 5$. Answer: 5 Example 2: Question: In the right triangle ABC, AB is perpendicular to BC. It is known that $AC = 5$ and $AB = 4$. Find the area of the right triangle. Answer: Step 1: (Mathematical theorem used: Pythagorean theorem): We first use the Pythagorean theorem to find the length of BC. The formula is: $AC^2 = AB^2 + BC^2$, where AC is the hypotenuse, and AB and BC are the legs. Step 2: (Substitute the known values): Given $AC = 5$ and $AB = 4$. Substituting these values: $5^2 = 4^2 + BC^2 \implies 25 = 16 + BC^2$. Step 3: (Solve for BC): $BC^2 = 25 - 16 = 9 \implies BC = \sqrt{9} = 3$. Step 4: (Calculate the area): The area of the right triangle is given by $\frac{1}{2} \times AB \times BC$. Substituting the known values: Area = $\frac{1}{2} \times 4 \times 3 = 6$. Answer: 6 Please reason step by step. Each step is placed on a new line, using the following format: Step X (Mathematical theorem/basis used): Detailed solution steps. Answer: { }

Table 8: Prompts used for single-step free-form questions in the evaluation for response generation.

Original Prompt
You are an assistant for solving math problems. Your input consists of a math question and images. Each problem is a multi-step problem. Give your answer directly, without any intermediate steps.
CoT (Chain of Thought)
You are a math problem-solving assistant. Your input is a math problem and a picture of the problem. Each problem is a multi-step problem. Your task is to output the solution ideas and answers for each step. The output format is step-by-step.
CoT with 2-shot Examples
Example 1: Question: If a triangle has two sides of length 3 and 4, (1) what is the length of the hypotenuse? (2) what is the area of this triangle? Answer: (1) Step 1: (Mathematical theorem used: Pythagorean theorem): The Pythagorean theorem states that in a right triangle, the square of the hypotenuse is equal to the sum of the squares of the other two sides. The formula is: $c^2 = a^2 + b^2$, where a and b are the legs, and c is the hypotenuse. Step 2: (Substitute the known values): Given $a = 3$ and $b = 4$. Substituting these values into the formula: $c^2 = 3^2 + 4^2 = 9 + 16 = 25$. Step 3: (Calculate the hypotenuse): Taking the square root gives: $c = \sqrt{25} = 5$. So the length of the hypotenuse is 5. (2) Step 1: The area of a right triangle is half the product of its two sides. Step 2: So the area of this triangle is $3 \times 4/2 = 6$. So the area of this triangle is 6. Example 2: Question: In the right triangle ABC, AB is perpendicular to BC. It is known that $AC = 5$ and $AB = 4$. (1) Find the area of the right triangle. (2) What is the height of the hypotenuse of this right triangle? Answer: (1) Step 1: (Mathematical theorem used: Pythagorean theorem): We first use the Pythagorean theorem to find the length of BC. The formula is: $AC^2 = AB^2 + BC^2$, where AC is the hypotenuse, and AB and BC are the legs. Step 2: (Substitute the known values): Given $AC = 5$ and $AB = 4$. Substituting these values: $5^2 = 4^2 + BC^2 \implies 25 = 16 + BC^2$. Step 3: (Solve for BC): $BC^2 = 25 - 16 = 9 \implies BC = \sqrt{9} = 3$. Step 4: (Calculate the area): The area of the right triangle is given by $\frac{1}{2} \times AB \times BC$. Substituting the known values: Area = $\frac{1}{2} \times 4 \times 3 = 6$. So the area of the right triangle is 6. (2) Step 1: According to the equal area method, the area of a right triangle is equal to half the product of the two right-angled sides, which is also equal to half the product of the hypotenuse and the corresponding height. Step 2: According to the above principle and the conclusion of the first step, we can get $AB \times BC/2 = AC \times h/2$. Step 3: Substituting the values, we get $h = 3 \times 4/5 = 2.4$. So the height of the hypotenuse of this right triangle is 2.4. Please reason step by step. Each step is placed on a new line, using the following format: Step X (Mathematical theorem/basis used): Detailed solution steps. Answer: { }

Table 9: Prompts used for multi-step free-form questions in the evaluation for response generation.

Multiple-Choice Prompt
You are an assistant for evaluating math problems. Your task is to extract the model’s answer to the given multiple-choice question and compare it with the ground truth. Steps: 1. Extract the model’s answer. The answer must be one of A, B, C, or D. 2. Compare the extracted answer with the ground truth. 3. Indicate whether the model’s answer is correct or incorrect. Output format: - Extracted Answer: {A}, {B}, {C}, or {D}. - Correctness: [true/false].
Single-Step Free-Form Prompt
You are an assistant for evaluating math problems. Your task is to extract the model’s answer to the given single-step free-form question and compare it with the ground truth. Steps: 1. Extract the model’s final answer. 2. Compare the extracted answer with the ground truth. 3. Indicate whether the model’s answer is correct or incorrect. Output format: - Extracted Answer: [Final Answer]. - Correctness: [true/false].
Multi-Step Free-Form Prompt
You are an assistant for evaluating math problems. Your task is to extract the model’s answers to each sub-question of a multi-step free-form problem and compare them with the ground truth. Steps: 1. Extract the final answers for each sub-question. 2. Compare the extracted answers with the corresponding ground truth. 3. Indicate whether each answer is correct or incorrect. Output format: - Sub-Question 1: Extracted Answer: [Answer]. Correctness: [true/false]. - Sub-Question 2: Extracted Answer: [Answer]. Correctness: [true/false].

Table 10: Prompts used for evaluating different types of math problems with the Deepseek API.

Model	Source	URL
Deepseek-V3	Deepseek-V3-0324	https://api-docs.deepseek.com/
Math-LLaVA-13B	local checkpoint	https://huggingface.co/Zhiqiang007/Math-LLaVA
LLaVA-v1.5-7B	local checkpoint	https://huggingface.co/liuhaotian/llava-v1.5-7b
InternLM-XComposer2.5-VL-7B	local checkpoint	https://huggingface.co/internlm/internlm-xcomposer2d5-7b
Deepseek-VL2-7B	local checkpoint	https://huggingface.co/deepseek-ai/deepseek-vl2
LLaVA-NeXT-Interleave-7B	local checkpoint	https://huggingface.co/lmms-lab/llava-next-interleave-qwen-7b
LLaVA-OneVision-Chat-7B	local checkpoint	https://huggingface.co/lmms-lab/llava-onevision-qwen2-7b-ov-chat
Qwen2.5VL-Instruct-7B	local checkpoint	https://huggingface.co/Qwen/Qwen2.5-VL-7B-Instruct
LLaVA-OneVision-Chat-72B	local checkpoint	https://huggingface.co/lmms-lab/llava-onevision-qwen2-72b-ov-chat
InternVL3-8B	local checkpoint	https://huggingface.co/OpenGVLab/InternVL3-8B
Mistral-small-3.1-24b-instruct	local checkpoint	https://huggingface.co/mistralai/Mistral-Small-3.1-24B-Instruct-2503
InternVL3-78B	local checkpoint	https://huggingface.co/OpenGVLab/InternVL3-78B
Llama-4-Maverick-17B-128E	local checkpoint	https://huggingface.co/meta-llama/Llama-4-Maverick-17B-128E-Instruct
LlamaV-o1-11B	local checkpoint	https://huggingface.co/omkarthawakar/LlamaV-o1
LLaVA-CoT-11B	local checkpoint	https://huggingface.co/Xkev/Llama-3.2V-11B-cot
VLM-R1-3B	local checkpoint	https://huggingface.co/omlab/VLM-R1-Qwen2.5VL-3B-Math-0305
R1-Onevision-7B	local checkpoint	https://huggingface.co/Fancy-MLLM/R1-Onevision-7B
Vision-R1-7B	local checkpoint	https://huggingface.co/0silly/Vision-R1-7B
Skywork-R1V2-38B	local checkpoint	https://huggingface.co/Skywork/Skywork-R1V2-38B
QvQ-72B-Preview	local checkpoint	https://huggingface.co/Qwen/QvQ-72B-Preview
GPT-4V	gpt-4-vision-latest	https://platform.openai.com/
Claude-3.5-Sonnet	claude-3.5-sonnet-2024-05-24	https://www.anthropic.com/news/claude-3-5-sonnet
Gemini-1.5-Pro	gemini-1.5-Pro-latest	https://ai.google.dev/
Gemini-2.5-Pro	gemini-2.5-pro-preview-05-06	https://ai.google.dev/
GPT-4o	gpt-4o-2024-05-14	https://platform.openai.com/
Claude-3.7-Sonnet	claude-3.7-sonnet-latest	https://www.anthropic.com/news/claude-3-7-sonnet
OpenAI-o1	OpenAI-o1-latest	https://platform.openai.com/

Table 11: The source of the models used in the evaluation.

Measurement of Solid Geometric Forms

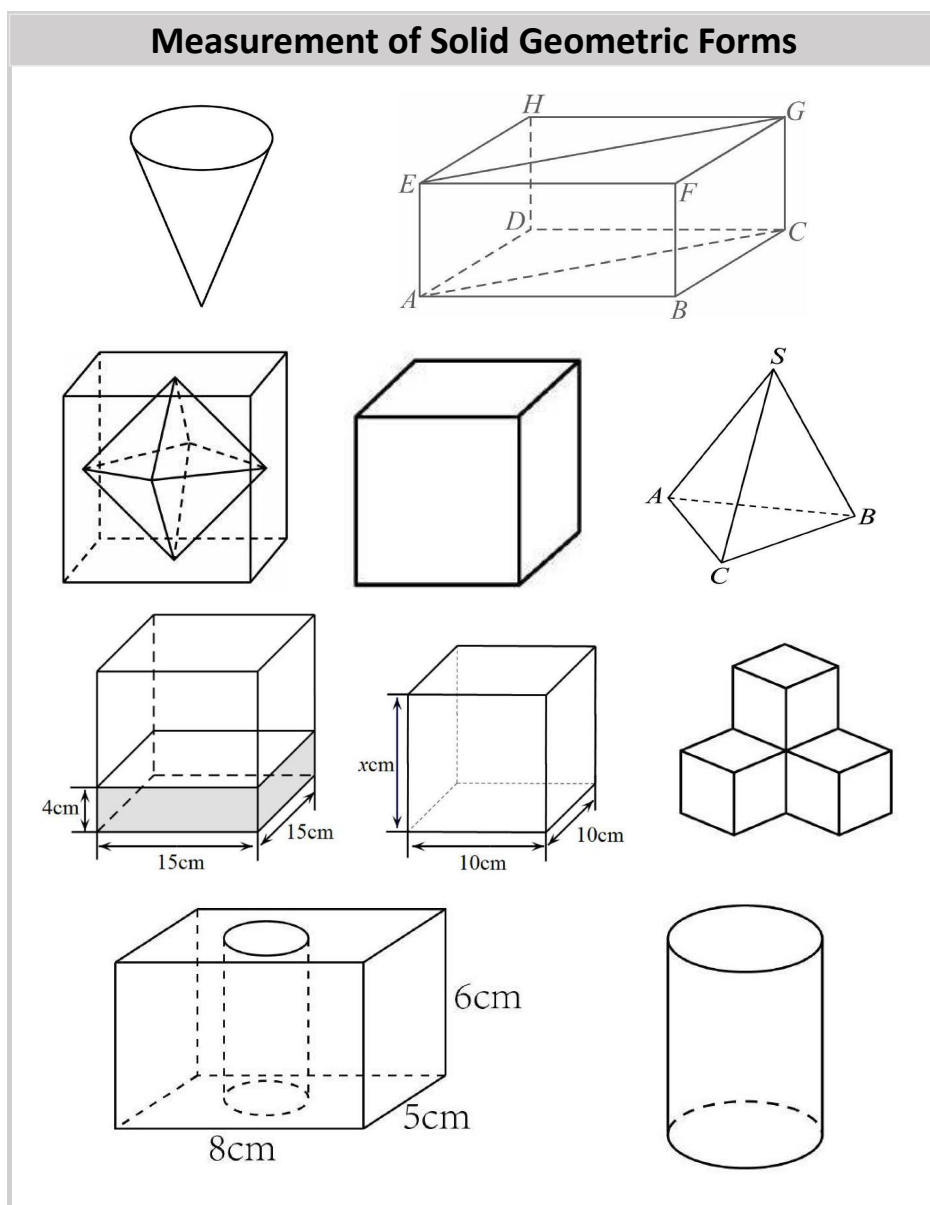


Figure 22: Some images from Measurement of Solid Geometric Forms.

Solid Shape Identification

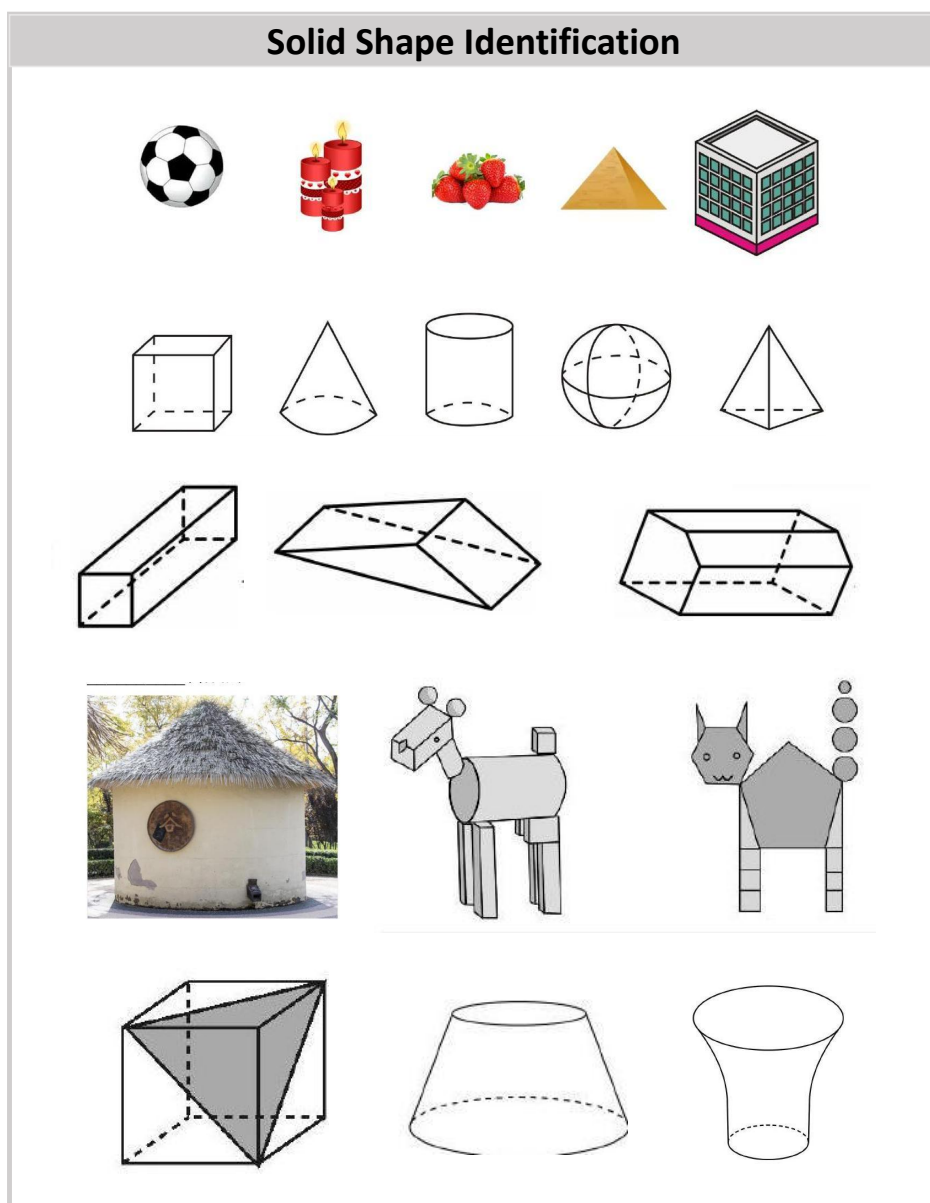


Figure 23: Some images from Solid Shape Identification.

Spatial Metric Relations

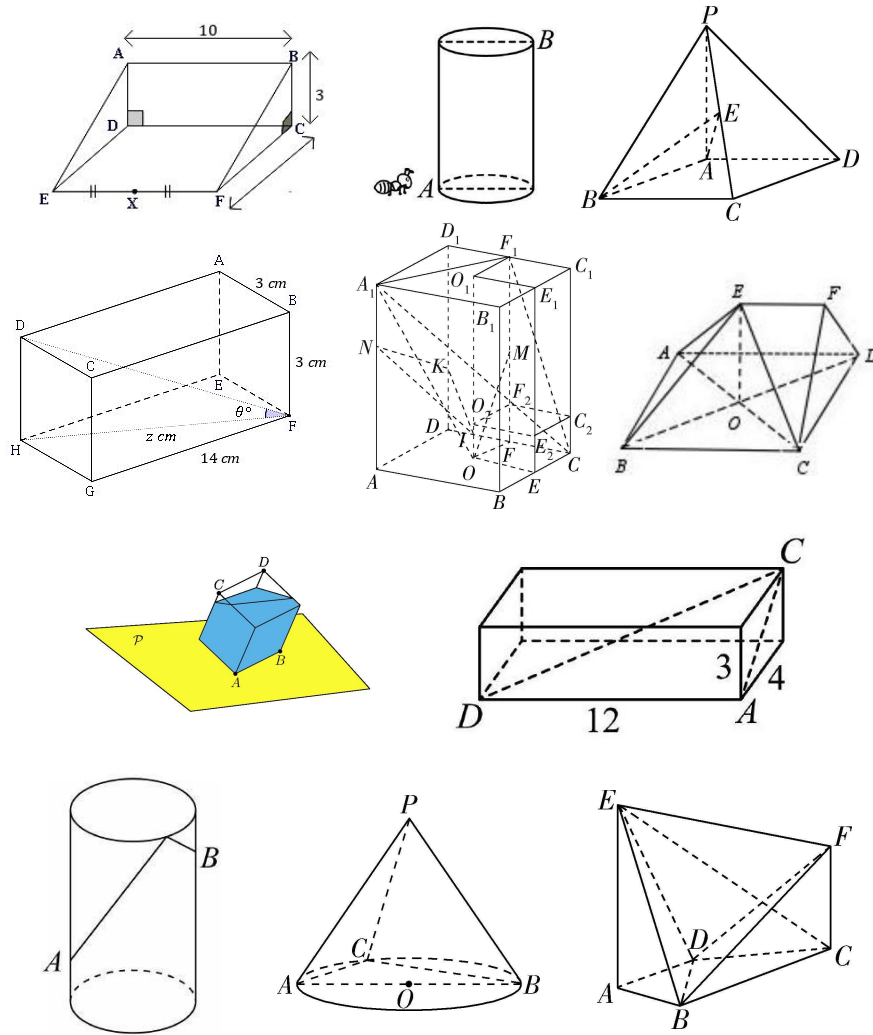
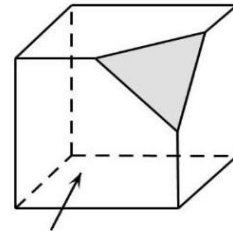
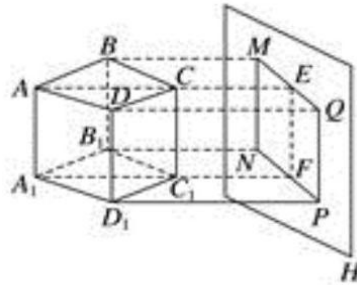


Figure 24: Some images from Spatial Metric Relations.

Multi-view Projection



Main viewing direction

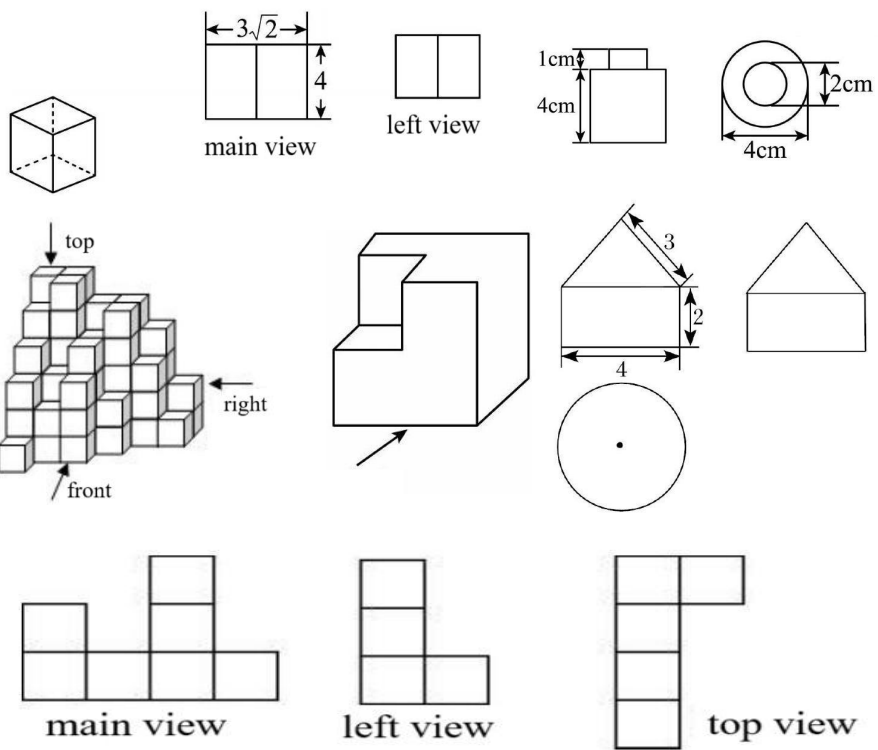


Figure 25: Some images from Measurement of Multi-view Projection.

Planar Unfolding and Configuration

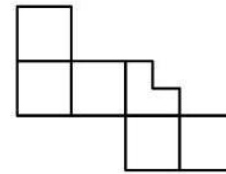
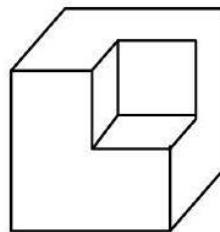
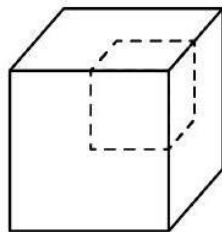
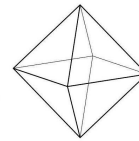
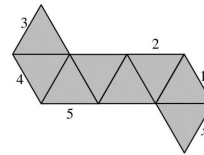
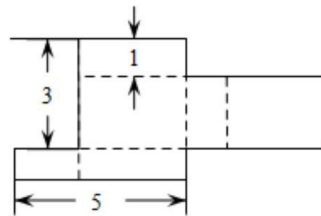
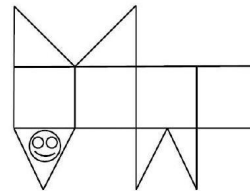
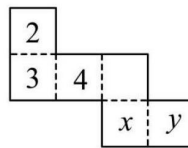
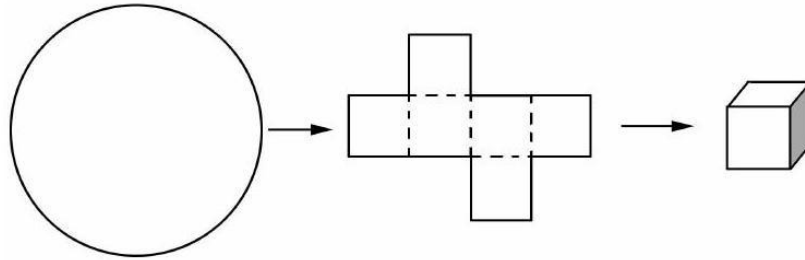


Figure 26: Some images from Planar Unfolding and Configuration.

Composite Solid Structure

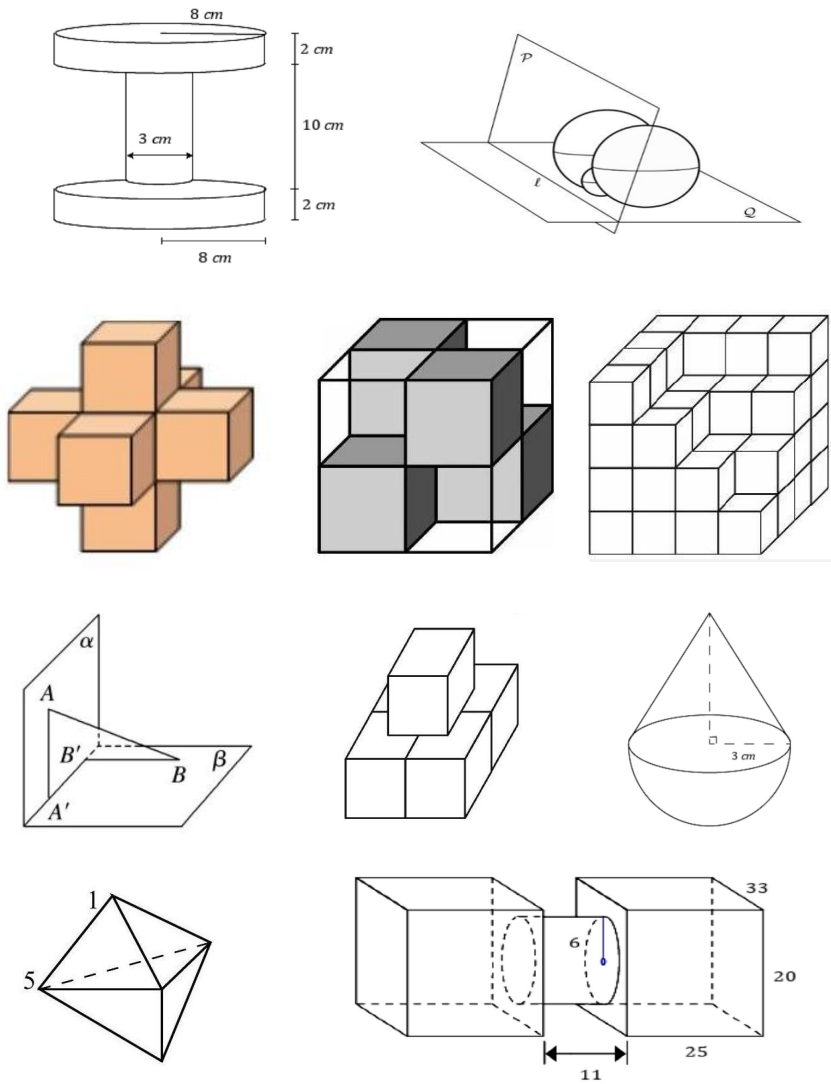


Figure 27: Some images from Composite Solid Structure.

3D Coordinate and Vector Reasoning

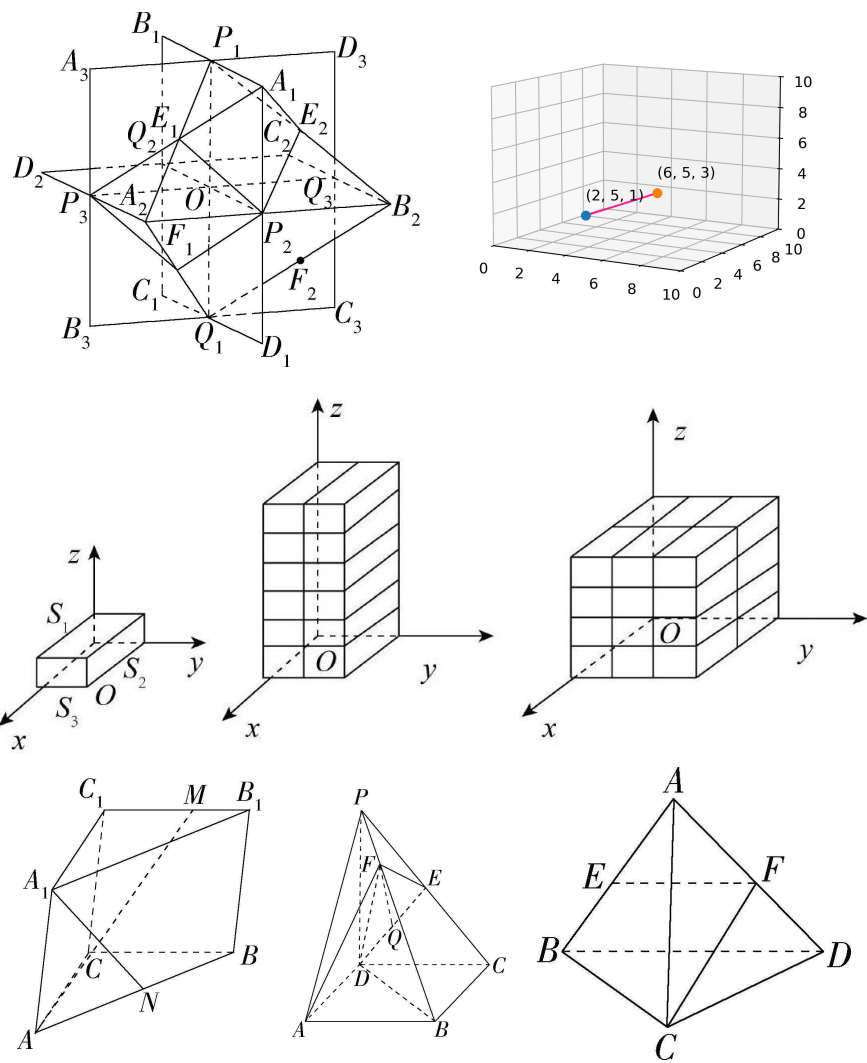


Figure 28: Some images from 3D Coordinate and Vector Reasoning.

Solid Geometric Modeling

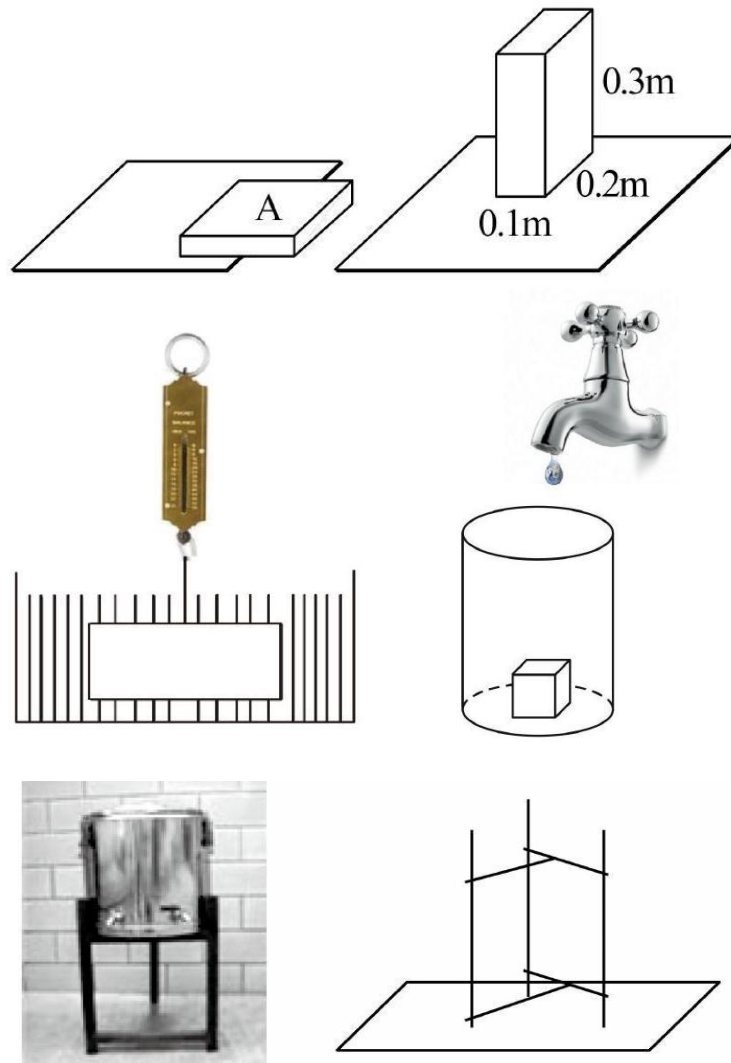


Figure 29: Some images from Measurement of Solid Geometric Modeling.

H Case Study

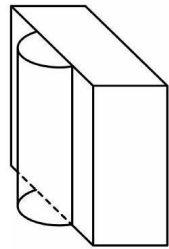
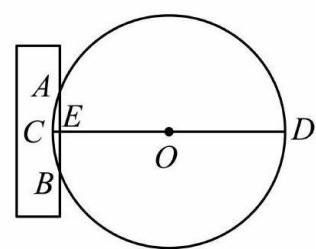
To delve into the failure cases of models, we detailed five typical error types in Table 12. Furthermore, to facilitate a better understanding of each error type, we provide examples of each error made by Gemini-2.5-pro from Figure 30 to Figure 34.

Table 12: Detailed Descriptions of Error Types.

Error Type	Explanation
Reason Error	Errors that occur in the logical reasoning process while using knowledge concepts to solve the problem step by step.
Visual Perception Error	Errors in visual perception, where the model incorrectly identifies shapes or numbers in an image.
Knowledge Error	For a specific knowledge concept, the model is unclear or confused about it, or it misuses another knowledge concept to solve the problem.
Hallucination	The thought process introduces factors that are not consistent with the facts, which are not mentioned in the context of the image or question.
Calculation Error	Errors arising from incorrect arithmetic operations, such as multiplication, addition, or formula application, despite having the correct reasoning path.

Calculation Error

Question: There is a cylindrical wooden block buried in the wall (as shown in Figure (1)). Its diameter is unknown, so it is sawed open (along the cross section) with a saw. (As shown in Figure (2)) When the measured depth CE is 1 inch, the sawed width AB is 1 foot, and the diameter CD of the wooden block is ____ inches. (1 foot = 10 inches)

Ground Truth: 12.16

Gemini-2.5-pro Answer

Step 1: Recognize that triangle $\triangle CDE$ is a right triangle with $CE=1$ inch (depth), $AB=12$ inches (sawed width), so $AE=6$ inches, i.e., the length of AE is half of AB .

Step 2: Use the Pythagorean theorem to compute the radius r of the circle: $r^2 = CE^2 + AE^2 = 1^2 + 6^2 = 1 + 36 = 37$. So, $r = \sqrt{37}$

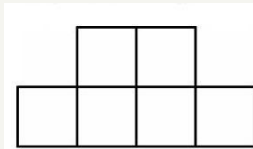
Step 3: The diameter is $2r = 2 \times \sqrt{37} \approx 12.12$ inches.

Answer: 12.12 inches

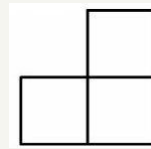
Figure 30: An example of calculation error.

Reasoning Error

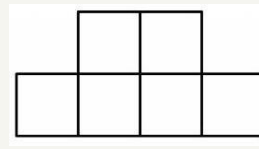
Question: An object is composed of multiple small cubes with edge lengths of 1. The shapes of the geometric body as seen from the front, left, and top are shown in the figures below. Then, the volume of the geometric body is ____.



View from the front



View from the left



View from the top

Ground Truth: 8

Gemini-2.5-pro Answer

To find the volume of the geometric body composed of small cubes with edge lengths of 1, we need to analyze the views from the front, left, and top.

Step 1 (Analyze the front view):

- The front view shows a structure with 3 columns.
- The left and right columns each have 1 cube. The middle column has 2.

Step 2 (Analyze the left view):

- The left view shows a structure with 2 columns. The left column has 1 cube. The right column has 2 cubes.

Step 3 (Analyze the top view):

- The top view shows a structure with 5 cubes arranged in a cross pattern.
- This confirms the layout seen from the front and left views.

Step 4 (Determine the number of cubes):

- From the front view, we have 1 cube in the left column, 2 cubes in the middle column, and 1 cube in the right column.
- From the left view, we have 1 cube in the left column and 2 cubes in the right column. Combining these views, we can deduce the following:
- The left column has 1 cube. The middle column has 2 cubes.
- The right column has 1 cube. The top view confirms that there are 6 cubes in total.

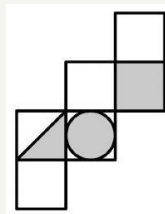
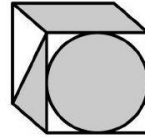
Step 5 (Calculate the volume): Since each cube has an edge length of 1, the volume of each cube is $1^3 = 1$. The total volume is the number of cubes multiplied by the volume of each cube.

Answer: The volume of the geometric body is 5.

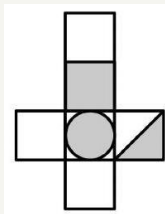
Figure 31: An example of reasoning error.

Visual Perception Error

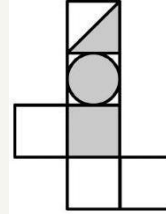
Question: As shown in the figure, if the cube is unfolded, which of the following patterns can be obtained?



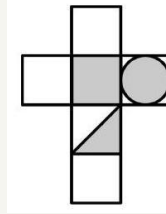
A



B



C



D

Ground Truth: D

Gemini-2.5-pro Answer

To determine which pattern can be obtained by unfolding the given cube, we need to visualize how the cube's faces are connected and how they would lay flat when unfolded.

Step 1: Identify the faces of the cube.

-The cube has six faces: one with a circle, one with a triangle, and four plain faces.

Step 2: Visualize the unfolding process.

-When the cube is unfolded, the faces will be laid out in a connected manner. The face with the circle will be adjacent to the face with the triangle, and the plain faces will be adjacent to these two faces.

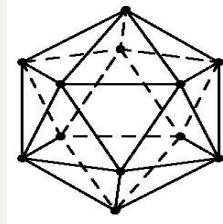
Step 3: Compare the given options with the expected layout.

- Option A: The layout shows the circle and triangle faces adjacent to each other, with the plain faces correctly positioned.
- Option B: The layout shows the circle and triangle faces adjacent to each other, but the plain faces are not correctly positioned.
- Option C: The layout shows the circle and triangle faces adjacent to each other, but the plain faces are not correctly positioned.
- Option D: The layout shows the circle and triangle faces adjacent to each other, but the plain faces are not correctly positioned.
- Answer: {A}

Figure 32: An example of visual perception error.

Knowledge Error

Question: A regular icosahedron is a 20-faced solid where each face is an equilateral triangle and five triangles meet at every vertex. The regular icosahedron shown below has one vertex at the top, one vertex at the bottom, an upper pentagon of five vertices all adjacent to the top vertex and all in the same horizontal plane, and a lower pentagon of five vertices all adjacent to the bottom vertex and all in another horizontal plane. Find the number of paths from the top vertex to the bottom vertex such that each part of a path goes downward or horizontally along an edge of the icosahedron.



Ground Truth: 810

Gemini-2.5-pro Answer: 1563

Step 1: The problem asks for the number of paths from the top vertex to the bottom vertex in a regular icosahedron. The paths must go downward or horizontally.

Step 2: Since the icosahedron has 20 faces and 12 vertices, and each face is an equilateral triangle, we assume that the graph formed by the vertices and edges is regular with degree 5.

Step 3: **Treat this as a 5-step traversal problem over a regular graph from one vertex to another. At each step, the vertex has 5 options (since degree is 5), so total possible paths $\approx 5^5 = 3125$.**

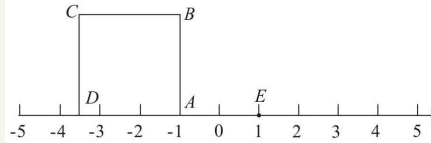
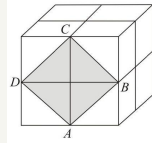
Step 4: Since only downward or horizontal paths are allowed, we restrict to half of these, so approximate total $= \frac{3125}{2} = 1562.5 \approx 1563$.

Figure 33: An example of knowledge error.

Hallucination Error

Question: As shown in Figure 1, this is a Rubik's Cube composed of 8 equally sized smaller cubes, with a total volume of 64.

- (1) Determine the edge length of the Rubik's Cube.
- (2) In the figure, quadrilateral ABCD is a square. Find the area and the side length of this square.
- (3) Place the square ABCD on the number line as shown in Figure 2, such that point A coincides with -1, point E coincides with 1, and point F is symmetric to point D with respect to point E. Then, the number represented by point D on the number line is _.



Ground Truth: (1) 4 (2) $8\sqrt{2}$ (3) $-1-2\sqrt{2}$

Gemini-2.5-pro Answer

(1) The total volume of the Rubik's Cube is 64 and it consists of 8 equal smaller cubes. So, each small cube has a volume of $\frac{64}{8} = 8$, thus its edge length is $\sqrt[3]{8} = 2$. The entire cube is $2 \times 2 \times 2$, so the cube's edge length is $2 \times 2 = 4$.

(2) The square ABCD lies on the face diagonal of the cube. The diagonal AC is $\sqrt{4^2 + 4^2} = \sqrt{32} = 4\sqrt{2}$. Since ABCD is a square with diagonal $4\sqrt{2}$, its side length is $\frac{4\sqrt{2}}{\sqrt{2}} = 4$, and area is $4^2 = 16$.

(3) To determine point D on the number line, we consider rotational symmetry as illustrated in a cube net. Since ABCD is placed on the number line and the cube has 8 small cubes, **we assume it folds along a 3D diagonal.**

In 3D rotation, D corresponds to a mirrored projection over the cube center, and since E is 1, the center is at 0. Thus, point D = $0 - (3 - 0) = -3$.

So the answer is -3.

Figure 34: An example of hallucination error.