Visual Mathematical Reasoning using Vision-Language Models

Mounika Mukkamalla, Neeharika Gupta, Taejas Gupta, Vishal Singh, Vishesh Agrawal

Abstract

Mathematical reasoning has been a recent area of focus for Large Language Models (LLMs), leveraging their comprehension abilities for problem solving. When the problems are presented visually in the form of visual puzzles or problems, the modality is no longer supported by LLMs alone. In this work, we perform various analytical experiments on different types of problems present in the MathVista dataset and evaluate the performance of Vision-Language Models (VLMs). Additionally, we highlight certain shortcomings of traditional evaluation metrics and propose a new scoring system for mathematical reasoning statements. Finally, we present an analysis of the results and suggest directions for future research.

1 Introduction

Mathematical reasoning has been a common presence in testing human intelligence, be it IQ tests or mental ability evaluations. Since a young age, humans are taught mathematical concepts and reasoning to develop intellect and thinking. Drawing inspiration from humans, to actually develop 'intelligent' machine learning models, it is important to ensure that these models are capable of some level of mathematical reasoning. Current research in mathematical reasoning for LLMs primarily handles problems presented in a textual format as input. However, there exist mathematical problems that require visual inputs along with the textual query. Such problems include mathematical puzzles, geometric questions, and visual question-and-answer problems. In these cases, LLMs alone do not suffice. This is where VLMs come into the picture with their ability to process both visual and textual input.

2 Related Works

Nowadays, LLMs have become the go-to solution for any given problem due to their sophisticated model architectures and the extensive data that they have been trained on. There has been a lot of work on text and vision modalities. Some of the recent approaches range from unimodal models like LLMs for text (GPT-3 [Brown et al., 2020], T5 [Raffel et al., 2020]) and vision models (Vision Transformer [Dosovitskiy et al., 2020], ResNet [He et al., 2016]) to LMMs catering to multiple modalities (GPT-4 [Achiam et al., 2023], LLaMA [Touvron et al., 2023]).

Various types of mathematical problems (Ahn et al., 2024) have been dealt with by training models tailored to specific kinds of problems. MAmmoTH (Yue et al., 2023) – is trained via hybrid instruction tuning, LLemma (Azerbayev et al., 2023) is capable of formal theorem proving without any further finetuning, Minerva (Lewkowycz et al., 2022) generates solutions without relying on external tools, and WizardMath (Luo et al., 2023) empowers reasoning via reinforced evol-instruct.

Recent models are also focusing on the reasoning of the answer instead of just the answer. (Lu et al., 2023) evaluates mathematical reasoning in visual contexts, (Kazemi et al., 2023) provides reasoning behind geometric problems, (Wang et al., 2023) seamlessly integrates code to enhance the model's reasoning capabilities and (Imani et al., 2023) boosts model confidence by providing more samples via zero-shot chain-of-thought prompting.

3 Problem Description

3.1 Objective

Solving visual mathematical problems is a challenging task. In this work, we analyze the responses generated by different VLMs and compare them across three main tasks. We additionally propose a human evaluation scoring system to evaluate mathematical reasonings.

To tackle the problem of visual mathematical

Task	Prompt
Problem Type	`There are 4 types of possible tasks that the following mathematical problem could be classified. These are Figure Question Answering (FQA), Textbook Question answering (TQA), Math Word Problems (MWP), Geometry Problem solving (GPS)` `Provide the task of the problem in the following return prompt and replace the ### by the right task acronym : "The task : ###." +` **`<Question>`**
Answer	`Provide the solution in the following return prompt and replace the ### by the right value. "The answer : ###". Do not give the reasoning, just answer is sufficient +` **`<Question>`**
Reasoning	**`<Question>`**

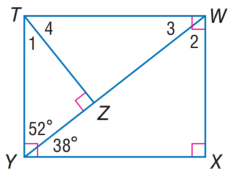
Table 1: Different prompts used to get the results on the three main tasks, with **`<Question>`** replaced by the text in the query field of the sample.

reasoning, we leverage VLMs to assess three main tasks: (1) identification of the problem type, (2) correctness of the answer, and (3) validity of the reasoning provided by the VLMs.

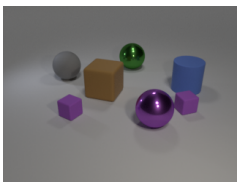
3.2 Dataset

We work with the MathVista dataset (Lu et al., 2023), which consists of five main categories of mathematical problems – math word problems (MWP), figure question answering (FQA), geometry problem solving (GPS), textbook question answering (TQA), and visual question answering (VQA). We focus our experiments on the first four categories as they are more mathematically oriented.

For the scope of this project, we evaluate our approaches on the testmini set – a subset of 1000 labelled samples. We work with a smaller set of 100 math problems for the human annotations and evaluation of reasoning due to the extensive nature of both the tasks.



(a)



(b)

Figure 1: Sample problems from the MathVista testmini set. (a) "Find $m \\angle 3$"; (b) "Subtract all big red metallic spheres. Subtract all big brown matte things. How many objects are left?"

4 Methods

4.1 Human annotations

While the MathVista dataset includes the correct answers for each question, it does not provide the reasoning for the answers. To evaluate the mathematical reasoning ability of VLMs on this dataset, we manually annotated our subset of 100 English math-targeted VQA problems. To incorporate a variation in individual response style, all five members of our group wrote down the reasoning for the 100 problems, giving 500 gold standard reasoning annotations. All of us worked individually with no communication about the style or format of the annotations for bias and fairness reasons. The dataset and 100 annotations can be found here.

4.2 Prompting strategy

To get our results on the three main tasks for each sample in the dataset, we used different kinds of prompts, as shown in Table 1. There was no explicit prompt required to get the reasoning as the VLMs we worked with provide the reasoning by default.

4.3 Post-processing strategy

We wrote a Python script that uses a heuristic-based approach to perform post-processing to have a consistent representation of mathematical expressions across the human annotations and the reasonings generated by the VLMs (such as ∠ and `m\\angle` being replaced by `angle`). This ensured that the similarity scores computed during evaluation were not penalized due to different representations of the same mathematical expression.

4.4 Traditional evaluation metrics and limitations

Our proposed idea was to use conventional n-gram/seq2seq evaluation metrics like BLEU (Papineni et al., 2002), BLEURT (Sellam et al., 2020), and BERTScore (Zhang et al., 2019) to compare the reasoning provided by a VLM with the gold standard annotations to determine whether the VLM's response was logically correct. Table 2 highlights a significant limitation of these metrics. The first column is the reasoning provided by GPT-4 (Achiam et al., 2023). The second column consists of two reference statements. The first statement is both logically and mathematically incorrect, but written in a similar style to GPT-4's response. The second statement is correct but written in a different style.

GPT-4 Response	Reference Statement	Logical and Mathematical Correctness	BLEU	BLEURT	BERTScore		
					Precision	Recall	F1 Score
The model in the image has 4 dots divided into 2 equal groups. There are 2 dots in each group.	The model in the image has 2 dots divided into 4 equal groups. There are 2 dots in each group.	Incorrect	0.75	0.87	0.99	0.99	0.99
The model in the image has 4 dots divided into 2 equal groups. There are 2 dots in each group.	There are 4 blue dots overall, and we divide into 2 equal groups. Groups are shown as circles in the image. Hence each group (each circle) contains 4/2 = 2 dots. Math and image supports this reasoning. So answer is 2.	Correct	0.08	0.56	0.93	0.87	0.90

Table 2: Evaluation scores obtained for two different reference statements for the GPT-4 generated response. The correct reference statement gets worse scores across all the traditional evaluation metrics.

It is observed that all three metrics are significantly lower for the correct statement. This suggests that these conventional metrics are unreliable for logical reasoning and mathematical correctness. Further, the metrics are not interpretable for logical reasoning. The changes in values across different examples provide no significant interpretation of logical similarity between two statements, indicating a need for a new scoring system.

4.5 Proposed scoring

To address the limitations of the traditional evaluation metrics, we propose a new scoring method for our use case. We identified three main aspects pertaining to evaluating mathematical reasoning statements – Logical correctness (LC), Mathematical correctness (MC), and Readability (R). LC and R are classified into three categories - Low (0), Medium (0.5), and High (1). MC is categorized as Correct (1) or Incorrect (0). In LC, a High category is assigned to reasonings that follow consistent and correct logic while a Low category is assigned to logically incorrect reasonings. Our final proposed score is the weighted sum of these three aspects.

$$\textbf{Proposed Score} = 0.5 \times LC + 0.3 \times MC + 0.2 \times R$$

The idea behind the weights is to give more importance to logical correctness and mathematical correctness over readability and semantic matches, leading to more interpretable results.

4.5.1 Human evaluation

We present our proposed scores for four VLMs' reasonings on our 100 human annotated samples. The reasonings were evaluated manually by our group based the three aspects described above. The same set of problems were assigned to members for evaluating each VLM's reasoning to maintain consistency across the scoring.

5 Experiments

5.1 Category-wise

The first set of experiments involve understanding the VLMs' ability to identify the type of problem based on the four categories presented to them. The idea behind this experiment is to understand the VLMs' innate ability to actually recognize the problem type when presented with the categories. This would help us understand if the model is failing in the first step itself and is therefore bound to reason incorrectly. Using zero-shot prompting (section 4.2), the VLMs' understanding of the presented problem and its category is tested on math-targeted VQA problems. Accuracy is used to assess the VLMs' performance on this task by doing a direct match between the task category provided in the MathVista dataset and that returned by the VLMs.

Model	Accuracy Score
GPT4	0.543
LLaVA	0.103

Table 3: Accuracy scores on problem type recognition

5.2 Answer-wise

The second set of experiments involve evaluating the VLMs' ability to arrive at the correct answer when presented with a visual math problem. For this experiment, we use all 1000 samples in the dataset. Due to resource the limitations, the first two experiments have been only run on ChatGPT-4 (Achiam et al., 2023) and LLaVA-7b (Liu et al., 2024). For ChatGPT-4, we make API calls directly to the VLM. For LLaVA-7b we use HuggingFace to run it on a T4 GPU. Accuracy is used to assess the VLMs' performance on this task by doing a direct match between the answer provided in the MathVista dataset and that returned by the VLMs.

Model	Accuracy	TQA	FQA	MWP	GPS
GPT4	0.397	0.377	0.250	0.521	0.514
LLaVA	0.070	0.155	0.107	0.091	0.014

Table 4: Accuracy scores on answer

5.3 Reasoning-wise

The third set of experiments involve evaluating the VLMs' ability to correctly solve the problem presented to them, while presenting correct logical arguments to support their solution. This helps us assess the VLMs' actual logical reasoning capability. To achieve this, we have used four VLMs on 100 math problems and then employed human evaluation along with our proposed scoring on the same (section 4.5) to arrive at the final scores.

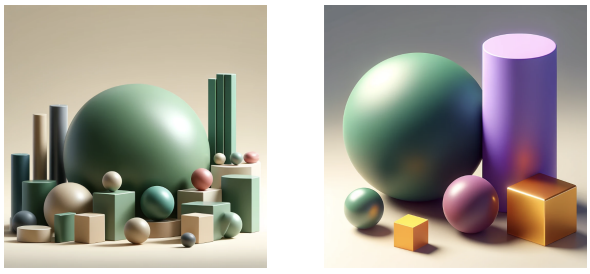
Model	Proposed Score
GPT4	0.697
Claude	0.647
Gemini	0.502
LLaVA	0.322

Table 5: Human evaluation scores on reasoning

5.4 Generating images using answer and reasoning

We performed an additional experiment to determine whether the VLMs were generating quality reasonings with a proper understanding of the problems. We hypothesized that a VLM should be able to recreate the image of a question when given the correct answer along with the reasoning that it had previously generated in a separate session.

Upon experimenting with zero-shot prompting, we observed that the VLM generates the image without any idea of the scope of the image, and the number of shapes and their colors in the generated image are very off. With one-shot prompting, we provide the VLM with a reference image, and it is able to generate a more accurate image, as shown in Figure 2.



(a) (b)

Figure 2: Zero-shot and one-shot prompting for generating image from textual query and reasoning for a question similar to that in Figure 1b

6 Analysis, Discussion and Conclusion

6.1 Analysis

The results of our different experiments indicate that GPT-4 consistently outperforms other VLMs. LLaVA performs significantly worse, possibly due to the smaller configuration of the model used by us. The proposed scoring highlights the superior performance of GPT-4 and Claude for mathematical reasoning.

6.2 Discussion

Mathematical reasoning is considered to be a very logical and deductive task. Currently, not enough VLMs have been trained for mathematical problem solving. We prompted various VLMs for different categories of mathematical tasks using different prompts. For the MathVista dataset, VLMs' performance is lower than human performance. While looking at category-wise scores, VLMs fare terribly when object-identification and understanding is required in the question. These VLMs are not able to reason and answer correctly simultaneously. These models also need to understand the mathematical context instead of the generic meaning to proceed with the reasoning process. Text degeneration is another problem faced by VLMs, particularly LLaVA, during reasoning – the reasoning generated by it is either repetitive or has little to no context. It lacks coherence with respect to the mathematical expressions used.

6.3 Conclusion

In this work, we evaluated the performance of different VLMs on math VQA tasks. We not only check the accuracy of the answer, but also the quality of reasoning and problem category identification. To compare various VLMs' reasoning, we manually annotate the reasoning of a subset of problems. We perform extensive experiments with different VLMs to check their efficacy over mathematical questions. While evaluating the reasonings, we identified the shortcomings of traditional evaluation metrics and used human evaluation as a better alternative. To conclude, we propose a scoring system which provides a comprehensive and interpretible way to evaluate mathematical reasonings.

6.4 Future directions

Our investigation helped us identify that there are some tasks in math VQA that are harder to reason. To enhance VLMs' understanding of VQA, external tools like solvers and object understanding tools can be deployed. To improve the reasoning generated by the model, we can explore zero or few-shot learning along with chain-of-thought reasoning.

7 Individual Contributions

All five members contributed equally toward data annotation and human evaluation. Mounika was responsible for the script to query ChatGPT-4 for the three experiments performed. Taejas was responsible for the post-processing task. Vishal was responsible for generating the conventional metric scores, and the accuracy scores across different experiments. Vishal, Vishesh and Neeharika ran the three different experiments on LLaVa-7b. Vishesh and Neeharika were responsible for obtaining the scores from the human evaluations for each VLM model. Vishesh ran experiments described in section 5.4. All team members obtained reasonings by manually prompting Claude and Gemini. For the final report, Vishal, Vishesh and Mounika wrote section 3.1, and 3.2, Taejas wrote section 4.1, 4.2, and 4.3 and generated Tables 1 and 2, Vishal wrote section 3.3, 4.4, 4.5, 5.1, 5.2, 5.3, and 6.1, Neeharika and Vishesh designed the structure, generated results tables 3, 4 and 5, and wrote sections 5.4, 6.2, 6.3, 6.4. The entire group was involved in brainstorming ideas and dividing work in a fair and equal manner.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

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

Zhangir Azerbayev, Hailey Schoelkopf, Keiran Paster, Marco Dos Santos, Stephen McAleer, Albert Q Jiang, Jia Deng, Stella Biderman, and Sean Welleck. 2023. Llemma: An open language model for mathematics. *arXiv preprint arXiv:2310.10631*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Shima Imani, Liang Du, and Harsh Shrivastava. 2023. MathPrompter: Mathematical reasoning using large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*, pages 37–42, Toronto, Canada. Association for Computational Linguistics.

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

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

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2023. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*.

Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, and Dongmei Zhang. 2023. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. *arXiv preprint arXiv:2308.09583*.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. Bleurt: Learning robust metrics for text generation. *arXiv preprint arXiv:2004.04696*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro,

Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Ke Wang, Houxing Ren, Aojun Zhou, Zimu Lu, Sichun Luo, Weikang Shi, Renrui Zhang, Linqi Song, Mingjie Zhan, and Hongsheng Li. 2023. Mathcoder: Seamless code integration in llms for enhanced mathematical reasoning. *arXiv preprint arXiv:2310.03731*.

Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. 2023. Mammoth: Building math generalist models through hybrid instruction tuning. *arXiv preprint arXiv:2309.05653*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.