SyncLoop: A Multimodal Dual-Loop Framework for Self-Improving Mathematical Reasoning

Xiuwei Chen[1], Wentao Hu[2], Hanhui Li[1*], Yongxin Wang[3], Jun Zhou[1], Zisheng Chen[1], Meng Cao[3], Yihan Zeng[4], Kui Zhang[4], Yujie Yuan[4], Jianhua Han[5], Hang Xu[5], and Xiaodan Liang[1*]

[1] Shenzhen Campus of Sun Yat-sen University, Shenzhen 518107, P. R. China
[2] The Hong Kong Polytechnic University, Hong Kong, P. R. China
[3] Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, United Arab Emirates
[4] Huawei Noah's Ark Lab, Shenzhen, P. R. China
[5] Yinwang Intelligent Technology Co. Ltd., Shanghai, P. R. China

Abstract. Recent advances in multimodal large language models (MLLMs) have shown impressive reasoning capabilities. However, further enhancing existing MLLMs necessitates high-quality vision-language datasets with carefully curated task complexities, which are both costly and challenging to scale. Although recent self-improving models that iteratively refine themselves offer a feasible solution, they still suffer from two core challenges: (i) most existing methods augment visual or textual data separately, resulting in discrepancies in data complexity (e.g., over-simplified diagrams paired with redundant textual descriptions); and (ii) the evolution of data and models is also separated, leading to scenarios where models are exposed to tasks with mismatched difficulty levels. To address these issues, we propose **SyncLoop**, an automatic, closed-loop self-improving framework that jointly evolves both training data and model capabilities. Specifically, given a base dataset and a base model, **SyncLoop** enhances them by a cross-modal data evolution loop and a data-model evolution loop. The former loop expands the base dataset by generating complex multimodal problems that combine structured textual sub-problems with iteratively specified geometric diagrams or mathematical functions, while the latter loop adaptively selects the generated problems based on the performance of the base model, to conduct supervised fine-tuning and reinforcement learning alternately. Consequently, our method continuously refines its model and training data, and consistently obtains considerable performance gains across multiple mathematical reasoning benchmarks.

Keywords: Co-Evolution · Difficulty Alignment · Multimodal Reasoning

* Corresponding authors.

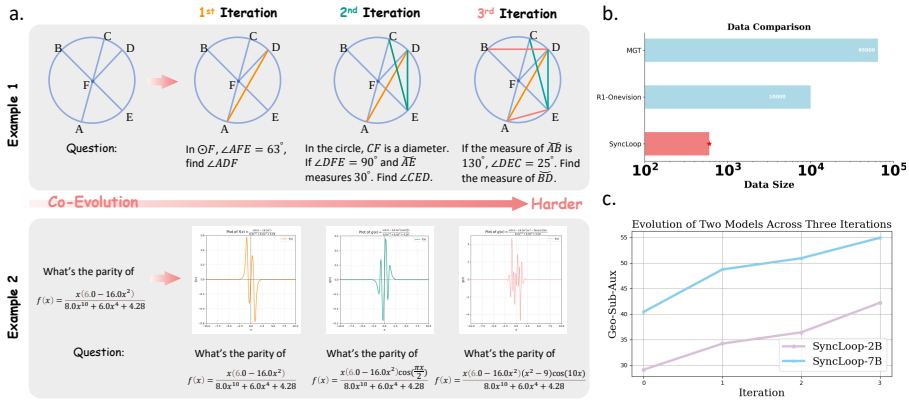

Fig. 1: a: Co-evolution of visual-textual pairs with escalating task difficulty over three iterations. **b:** Data usage comparison among different methods. Note that the x-axis (gate count) is on a log-scale. **c:** Performance of 2B and 7B models across three iterations.

1 Introduction

Recent advancements in large language models (LLMs) have achieved remarkable progress in solving problems, including mathematics [3, 17], coding [2, 12], etc. These capabilities are enabled by advanced strategies, including chain-of-thought prompting [43], tool-augmented reasoning [10, 18, 22], *etc*. In particular, OpenAI o1 [30] and Deepseek-R1 [14] have shown that reinforcement learning plays a critical role in aligning model outputs with desired behaviors by using structured reward signals derived from correctness, consistency, or human preference. This mechanism has proven especially effective in eliciting nuanced self-verification and self-correction behavior in LLMs, thereby reinforcing the reliability and depth of their reasoning chains, particularly in mathematical and logical domains.

Despite these advances, achieving such strong reasoning performance remains heavily reliant on large-scale, high-quality, and complexity-aligned datasets. As task complexity increases, collecting suitable training data becomes significantly more costly and difficult, presenting a major bottleneck to further progress. This challenge has sparked growing interest in *self-improving* paradigms, where models iteratively enhance their capabilities by generating new synthetic data and refining reasoning traces.

Recent studies have shown that reasoning abilities can be substantially improved through carefully curated and progressively challenging data. For example, OpenVLThinker [7] adapts the self-improvement paradigm to the vision-language domain by iteratively alternating between supervised fine-tuning (SFT) and reinforcement learning (GRPO), distilling R1-style reasoning traces from text-based models into multimodal contexts.

However, despite these promising developments, existing approaches face two key limitations: (1) **Mismatched evolution of visual complexity and textual reasoning difficulty**. Prior methods often suffer from decoupled difficulty

Table 1: Comparison of evolutionary strategies.

Model	Visual Evolve	Text Evolve	Capability-Aligned
MindGYM [44], VisPlay [16]	✘	✔	-
R-CoT [6]	✔	✘	-
MAVIS [50]	✔	✘	-
OpenVLThinker [7]	-	-	✘
SyncLoop(Ours)	✔	✔	✔

scaling, where the evolution of visual and textual complexities is asynchronous. Some approaches prioritize visual complexity but fail to match it with deep reasoning tasks, while others enhance textual semantics but are constrained by static visual sources. This disconnect restricts the model's ability to learn integrated cross-modal reasoning strategies. (2) **The discrepancy between model capability and task difficulty**. As models improve over the course of training, their ability to tackle more complex tasks naturally increases. However, current approaches rely on static or manually defined difficulty schedules, which do not adapt to the model's evolving capability. This misalignment can lead to inefficient training, either under-challenging the model or overwhelming it with excessively difficult data.

To address these challenges, we propose a fully automated, adaptive multimodal learning framework (**SyncLoop**) that jointly evolves both the model and its training data in a closed-loop fashion, with a particular focus on diagram-based mathematical reasoning tasks. Table 1 summarizes the differences between our framework and state-of-the-art methods: Unlike existing methods that rely on static data, our method dynamically adjusts task complexity based on real-time assessments of model performance, ensuring a tighter coupling between model capability and data difficulty throughout the learning process. Specifically, to tackle **the challenge (1)**, we incorporate the process into a cross-modal data evolution loop, where complex visual elements (*e.g.*, geometric diagrams or function plots) are jointly synthesized with semantically aligned textual problems. This is achieved by leveraging an external reasoning engine to generate and render visual augmentations, followed by the automatic construction of multi-step reasoning questions grounded in the generated visuals. The resulting multimodal samples are filtered and validated to ensure internal consistency and cross-modal coherence, yielding a dataset in which visual and textual complexities are jointly calibrated. For **the challenge (2)**, we adopt a data-model evolution loop. This loop utilizes data generated from the cross-modal data evolution loop and applies it to iteratively fine-tune the model using SFT and GRPO. SFT maintains output structure and coherence, while GRPO improves generalization through rule-based optimization. We introduce a simple error-based filtering method that evaluates sample difficulty via prediction variance over 32 generations. By selecting samples with an general error rate (*e.g.*, 0.3), which are prone to error yet still within the model's grasp and pose a meaningful challenge, the framework ensures

that task difficulty remains aligned with model capability, achieving continuous improvement over iterations (*cf.*, Figure 1 (c)).

Finally, we investigate the impact of different data strategies and iterative training regimes on model performance. Our findings offer insights into the design of effective self-improving frameworks for improving complex diagram-based mathematical reasoning and guiding the progressive evolution of vision-language models.

Our contributions are summarized as follows:

• We propose a closed-loop self-improving framework, named **SyncLoop**, that jointly evolves training data and model capabilities.

• The proposed framework utilizes two co-evolution loops to improve the compatibility of cross-modal complexity and that between task difficulty and model capability.

• Extensive experiments (*e.g.*, Geo-Sub, MathVista, MathVerse) demonstrate the effectiveness of different data strategies and iterative training regimes, revealing their impact on self-improving frameworks.

2 Related Work

Reinforcement Learning. Recent research has demonstrated that reinforcement learning (RL) can significantly enhance the reasoning capabilities of large language models (LLMs) such as OpenAI-o1 [30]. Some approaches have introduced RL-based mechanisms to facilitate test-time scaling, achieving notable success in tasks such as mathematical reasoning and code generation. Building on this progress, DeepSeek-R1 [14] proposed a rule-based reward strategy and adopted the Group Relative Policy Optimization (GRPO) [36] algorithm, demonstrating strong performance with only a few update steps. Motivated by the success of LLMs, recent studies [5, 19, 28, 31, 33, 37, 46, 48, 51] have begun to explore the reasoning capabilities of MLLMs. For example, R1-OneVision [46] integrates supervised fine-tuning with RL to bridge the gap between visual perception and deep logical reasoning. Vision-R1 [19] generates cold-start initialization data and employs GRPO with hard format reward functions to enhance the emergent reasoning capabilities of MLLMs. VisualThinker-R1-Zero [51] applies the R1 style to a base MLLM without supervised fine-tuning, surpassing traditional fine-tuning methods while exhibiting "visual aha moment" behaviors. R1-VL [48] proposes the Step-wise Group Relative Policy Optimization (StepGRPO), an online reinforcement learning framework for improving MLLMs' reasoning ability through dense, effective step-wise rewards. MM-EUREKA [28] demonstrates that rule-based reinforcement learning can be effectively extended to multimodal reasoning, enabling emergent reasoning behaviors without supervised fine-tuning and offering superior data efficiency. Curr-ReFT [5] adopts a multi-stage curriculum learning strategy with progressively increasing difficulty to support the self-evolution of reasoning capabilities, LMM-R1 [33] adopts a staged approach that begins with textual reasoning and advances toward complex multimodal reasoning tasks. MM-EUREKA [28] introduces a novel data filtering strategy that simultaneously

removes both unsolvable and trivial cases, along with rejection samples, retaining only high-confidence instances. The effectiveness of these methods in multimodal understanding tasks can be attributed to the presence of high-quality CoT datasets and the use of R1-style RL. Nonetheless, a significant drawback persists: the disconnect between the complexity of the training data and the difficulty of the tasks, particularly the inconsistency between the complexity of visual inputs and the challenges of textual reasoning. In this paper, we introduce a self-evolving training approach that alternates between RL and SFT. Our method dynamically modifies the multimodal training data to align with the model's reasoning abilities, ensuring that the complexities of both visual and textual elements are consistent throughout the training process. In contrast to MM-EUREKA, our method prioritizes samples that are both meaningfully challenging and conditionally accessible, striking a balance between task difficulty and model learnability.

Self-Improvement. Self-improvement [1, 11, 16, 35] is a paradigm in which models generate and train on synthetic data generated from the same or other models. While self-improvement has been widely studied in the NLP domain, several works [4, 13, 23, 27, 38, 47] have explored the approach of first generating high-quality data and subsequently fine-tuning models on this data to achieve continuous performance improvement. For example, STaR [47] introduces a bootstrapping mechanism that enhances LLM reasoning capabilities by iteratively generating and filtering "chain-of-thought" rationales, then fine-tuning the model on correct rationales to progressively improve performance. ReST [13] integrates self-generated data with offline RL, alternating between a "Grow" phase that expands the dataset by generating multiple outputs per input, and an "Improve" phase that ranks and filters these outputs using a reward model based on human preferences. Other works [27] follow a similar approach of generating synthetic data, filtering out low-quality samples, and fine-tuning models on the filtered high-quality data. Recent efforts [6, 9, 15, 26, 34, 40, 50] have introduced synthetic data generation pipelines that leverage a visual data engine to produce visual images along with corresponding reasoning tasks. However, these methods primarily focus on expanding the training distribution. Our approach shifts the emphasis from passive data enrichment to self-improvement with a co-evolution mechanism. A closely related work is OpenVLThinker [7], which introduces an iterative training paradigm in which models are exposed to progressively more complex tasks. However, their approach relies on manually defined task difficulty, which does not adapt to the evolving capabilities of the model. [6]

3 Method

In this section, we present the details of the proposed **SyncLoop** framework for self-improving reasoning. The key idea behind **SyncLoop** is the joint evolution of multimodal data and models, thereby mitigating discrepancies not only between visual and textual modalities but also between task difficulty and model

[6] Additional Related Work is provided in the **APPENDIX A**.

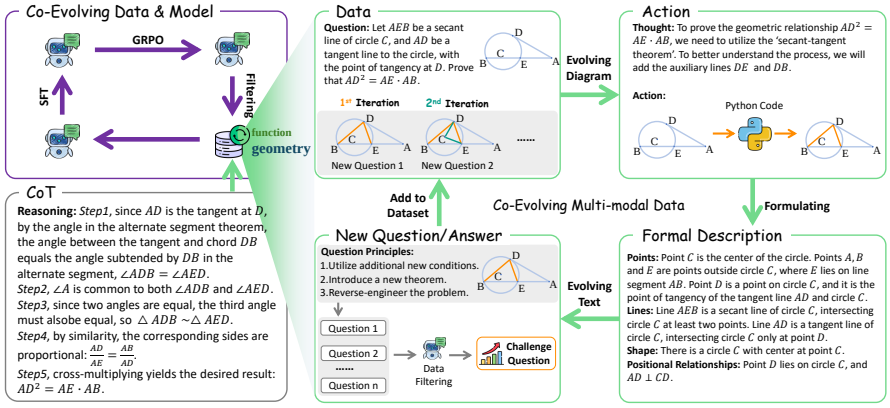

Fig. 2: The proposed SyncLoop framework. Starting from a base dataset and a base model, a co-evolving multimodal data loop iteratively synthesizes complex paired visual and textual samples. These samples are subsequently filtered and utilized in the co-evolving data & model loop to iteratively improve the reasoning performance of the model.

capabilities. We begin by formulating our task in Sec. 3.1, and then introduce the two core components of **SyncLoop**, namely the multimodal data co-evolution loop (Sec. 3.2), and the data and model co-evolution loop (Sec. 3.3), as shown in Figure 2.

3.1 Task Formulation

In this paper, we are given a pre-trained multimodal large language model (MLLM) denoted as π_θ and a dataset $\mathcal{D}$ consisting of image-question-answer triplets, namely, $\mathcal{D} = \{D_n = (I^n, Q^n, G^n) | n = 1, ..., |\mathcal{D}|\}$, where I^n, Q^n, and G^n denote an image, questions related to I^n, and the corresponding answers of Q^n, respectively. Our goal is to improve the complex reasoning capability of π_θ by optimizing its parameters θ on $\mathcal{D}$ within T iterations. Following recent studies [6, 18, 50], we assume that the necessary external tools and models are available, such as an oracle model that can generate and execute Python code to edit images [18].

Specifically, let $t = 1, ..., T$ denote the iteration step. At the beginning of the t-th iteration, we first conduct the multimodal data co-evolution loop to augment $\mathcal{D}_t$. This is achieved by prompting an MLLM [20] as the oracle model for solving each question in $\mathcal{D}_t$, generating step-by-step reasoning trajectories including necessary image augmentations like drawing auxiliary lines. For an arbitrary triplet $D_t^n \in \mathcal{D}_t$, the MLLM produces an action sequence Ac_t^n, corresponding Python code segment C_t^n, and a natural language reasoning trace R_t^n. The generated code C_t^n is then executed using an external image editing tool [21] to yield a more complex image I_{t+1}^n that includes auxiliary augmentations. To ensure

Algorithm 1 SyncLoop Algorithm

Input: Seed Dataset $\mathcal{D}$, Base Model π_θ, Principles P, Number of Iterations T
Output: Improved Model $\pi_{\theta,\mathrm{RL}}^{T+1}$, Evolved Datasets D_{T+1}

Initialize $\pi_\theta^1 = \pi_\theta$
for $t = 1$ to T **do**
  ▷ **Co-Evolving Multimodal Data** ▷ see §3.2
  for $n = 1$ to $|\mathcal{D}_t|$ **do**
    $Ac_t^n, C_t^n, R_t^n \leftarrow$ generate action and reasoning conditioned on D_t
    $I_{t+1} \leftarrow$ apply auxiliary augmentations by executing (Ac_t^n, C_t)
    $Q_{t+1}, A_{t+1} \leftarrow$ generate challenging questions and answers conditioned on I_{t+1}, P, Q_t
    $D_{t+1} \leftarrow$ add filtered tuple $(I_{t+1}, Q_{t+1}, A_{t+1})$
  end for
  ▷ **Co-Evolving Data and Model** ▷ see §3.3
  $D_t^{\mathrm{train}} \leftarrow$ select data from D_{t+1} using error rate evaluated with $\pi_{\theta,\mathrm{RL}}^t$
  $\pi_{\theta,\mathrm{SFT}}^{t+1} \leftarrow$ update π_θ^t with SFT on D_t^{train} ▷ using Equation 1
  $\pi_{\theta,\mathrm{RL}}^{t+1} \leftarrow$ update $\pi_{\theta,\mathrm{SFT}}^{t+1}$ with GRPO on D_t^{train} ▷ using Equation 2
end for

that the questions align with I_{t+1}^n with increased complexity, more challenging questions Q_{t+1}^n and answers G_{t+1}^n regarding I_{t+1}^n are generated and curated, consequently a new multimodal triplet $D_{t+1}^n = (I_{t+1}^n, Q_{t+1}^n, G_{t+1}^n)$ is obtained. We assess the difficulty of D_{t+1}^n and if it is sufficiently challenging, we incorporate it into $\mathcal{D}_t$. Afterwards, we utilize the updated dataset $\mathcal{D}_{t+1}$ to train the base model π_θ through a combination of SFT and GRPO, where SFT establishes the initial reasoning structure and GRPO improves its generalization capability.

3.2 Co-Evolving Multimodal Data

The co-evolving multimodal data loop is introduced to mitigate the complexity discrepancy between visual and textual data, which can be further divided into an action and reasoning generation process and a challenging question generation process.

Action and Reasoning Generation. We introduce a two-stage framework designed to enable the precise construction of auxiliary lines in geometric diagrams. We first extract the coordinates of key points using Optical Character Recognition (OCR) and other vision-based techniques. This coordinate-level representation ensures robustness and consistency in subsequent diagram transformations, particularly under increased geometric complexity. Leveraging this representation and image I_t (we omit the superscript n for conciseness), we prompt[7] GPT-4o to determine whether auxiliary image augmentations are needed to facilitate problem solving. This generates a decision, denoted as *Thought*, indicating whether

[7] All detailed prompts are provided in the **APPENDIX F.3.**

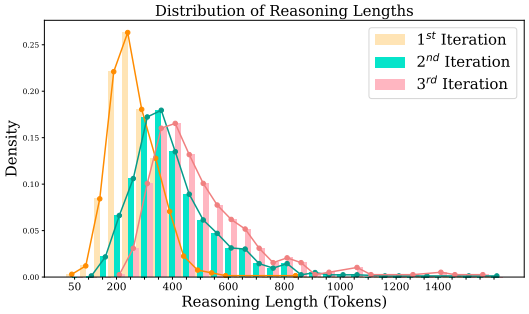

Fig. 3: Evolution of reasoning length (tokens) over three iterations.

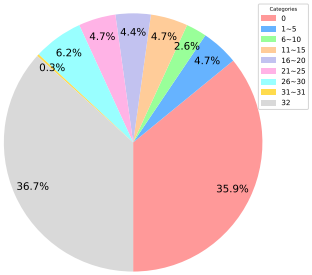

Fig. 4: Assessment of third iteration training data difficulty using second iteration model $\pi^2_{\theta,\mathrm{RL}}$.

auxiliary augmentations are necessary and specifying their types (*e.g.*, parallel lines, perpendicular lines, connecting lines and function graph). Subsequently, it produces the updated image I_{t+1} with newly added auxiliary augmentations, as illustrated in Figure 2. In addition to I_{t+1}, the pair (I_t, Q_t) is also fed to GPT-4o to simultaneously generate the full reasoning trace R_t, which is used in the subsequent generation of challenging questions. For mathematical functions, we prompt GPT-4o to decide whether plotting the graph would aid in solving the given problem. The rest of the process follows the same procedure as before.

Challenging Question Generation. This step aims to match the level of problem difficulty with the degree of image complexity. For geometric diagrams, we utilize the GPT-4o to generate a formatted description F_t conditioned on the generated images I_{t+1}. To promote diversity in the generated sub-problems, we define a set of guiding principles P:

1) *Math Constraints*: We extract mathematical constraints (*e.g.*, perpendicularity, equality, and sub-images) from the formal description F_t.

2) *New Theorems and Concepts*: We incorporate relevant mathematical theorems and conceptual principles (*e.g.*, the properties of a right triangle imply the Pythagorean theorem, symmetry, parity) that are closely related to the formal description F_t.

3) *Backward Reasoning*: We also include the concept (*e.g.*, the Tangent-Secant theorem) of sub-steps in the inference trace (*i.e.*, R_t).

Using these principles, we prompt GPT-4o with the tuples (F_t, R_t, P) (*e.g.*, F_t is applied exclusively to geometric diagrams) to generate a diverse set of sub-problems $(q_1, q_2, \ldots, q_m$, where m typically ranges from 4 to 10, depending on the complexity of the image). To mitigate the inherent limitations of formal language in capturing visual content, we incorporate the role-playing strategy from R1-Onevision [45]. By analyzing the differences and commonalities among sub-problems $(q_1, q_2, \ldots, q_m)$, we align them with corresponding sub-image elements to compose challenging geometric reasoning questions. Specifically, by combining the sub-problems $(q_1, q_2, \ldots, q_m)$ with their associated sub-image descriptions F_t, we prompt GPT-4o to compose challenging questions. For example, given

two sub-problems q_1 and q_2 that address different parts of an image, we combine them into a more challenging question Q_{t+1} by leveraging their thematic and contextual connections.

To obtain corresponding answers, we feed each generated question Q_{t+1} together with its formal description F_t into GPT-4o three times. For each input, the model produces a step-by-step reasoning trace R_{t+1} and the final answer A_{t+1}. We then retain only those samples with three answers $A_{t+1,1}, A_{t+1,2}, A_{t+1,3}$ are equal. We further prompt model to filter out any questions that are inconsistent with the original image constraints, ensuring alignment between the question and the visual content. As illustrated in Figure 3, the increasing length of the reasoning trajectories serves as a reliable indicator of problem difficulty, aligning with our data evolution design.

3.3 Co-Evolving Data and Model

SFT to Follow Reasoning Structure This stage aims to unify the model's reasoning format with the structure needed during the later reinforcement learning (GRPO) phase, ensuring compatibility through a standardized template (*i.e.*, <think></think><answer></answer>). The model is trained on the previously constructed triplets (I_{t+1}, Q_t, R_t), learning to generate the reasoning trace and final answer in a structured format. The corresponding training objective is formulated as follows:

$$\mathcal{L}_{\text{SFT}} = -\mathbb{E}_{(I,Q,R)\sim D}[\log(\pi_\theta(R|I,Q)]. \tag{1}$$

Group Relative Policy Optimization Based on the above obtained $\pi_{\theta,\text{SFT}}$ model, we employ two reward rules in conjunction with the GRPO algorithm to further refine the policy model. These rewards are designed as follows:

1) Accuracy Reward: This reward evaluates the correctness of the final answer A by processing the final answer via regular expressions and verifying it against the ground truth G.

2) Format Reward: To ensure consistent and well-structured reasoning trajectories, we define a reward based on the model's adherence to the predefined format <think>$\cdots$ </think>.

Additionally, we enforce sequential consistency within the reasoning trace by requiring that each step be explicitly arranged in the correct order. Responses that violate this ordering are penalized during training. The training objective with GRPO is defined as,

$$\mathcal{L}_{\text{RL}} = \mathbb{E}\Big[\min\left(r_t(\theta)A_t, \text{clip}(r_t(\theta), 1-\varepsilon, 1+\varepsilon)A_t\right) - \beta D_{\text{KL}}[\pi_\theta \| \pi_{\text{old}}]\Big] \tag{2}$$

Iterative Refinements and Filtering

In the t-th iteration, we perform SFT followed by GRPO training using all post-filtered data, resulting in an updated model $\pi^{t+1}_{\theta,RL}$ and a new dataset $\mathcal{D}_{t+1}$ for the next round. To align the difficulty of $\mathcal{D}$ with the current capability of π_θ, we forward each sample through the model 32 times and compute the error rate

as the proportion of times the model generated a wrong answer. As illustrated in Figure 4, we evaluate the difficulty of the data for each model obtained from GRPO training. We then retain only the samples with an error rate greater than 0.3 to form the training set for the next iteration, ensuring that the data complexity remains aligned with model capability. More assessments are provided in **APPENDIX F.2**.

Table 2: Main experimental results. For MathVista benchmark, we have specifically compared all models on three sub-tasks that are highly related to mathematical reasoning: geometry reasoning (GEO), algebraic reasoning (ARI) and geometry problem solving (GPS). The result (†) is collected from original papers, R1-VL[48] and R1-Onevison [45]. The remaining results are reproduced under the same experimental setting.

Methods	Data Amount	Geo-Sub	Geo-Sub-Aux	MathVista			
				GEO	ARI	GPS	ALL
Closed-Source Model							
GPT-4o [20]		-	-	-	-	-	63.8†
Reasoning Model			-				
LLamaV-o1-11B[39]	>100k	-	-	-	-	-	54.4†
Insight-V-8B[8]	200K	-	-	-	-	-	49.8†
MGT-PerceReason [32]	65k	-	-	-	-	-	63.2†
R1-Onevision-7B [45]	10k	-	-	-	-	-	64.1†
R1-VL-7B [48]	10k (SFT 120k)	-	-	-	-	-	63.6†
Qwen2-VL-2B [42]	-	28.0	29.1	-	-	-	43.0†
SyncLoop-1^{st}	0.6k	32.0	34.2	38.0	40.0	38.0	49.1
SyncLoop-2^{nd}	0.6k	35.6	36.4	38.0	38.0	38.0	49.3
SyncLoop-3^{rd}	0.4k	38.2	42.2	40.0	40.0	38.0	50.2
Qwen2-VL-7B [42]	-	40.4	40.4	50.0	50.0	49.0	60.0
SyncLoop-1^{st}	0.6k	45.5	46.9	55.0	57.0	54.0	62.1
SyncLoop-2^{nd}	0.6k	46.9	48.0	56.0	58.0	56.0	62.4
SyncLoop-3^{rd}	0.4k	50.9	52.4	59.0	59.0	60.0	63.2

4 Experiment

4.1 Benchmark

Dataset. We use the Geometry3k [25] dataset as the base dataset for multimodal data evolution. Geometry3k is a mathematical benchmark dataset that comprises 3,002 geometry problems, divided into 2,101 training examples, 300 for validation, and 601 for testing. Each problem is accompanied by a corresponding geometric diagram, a natural language description, and formal language annotations.
Implementation Details. In our experiments, we adopt two state-of-the-art open-source MLLMs, *i.e.*, Qwen2-VL-2B [42] and Qwen2-VL-7B [42]. For the policy warm-up phase, we employ the LLaMA-Factory. framework with a batch

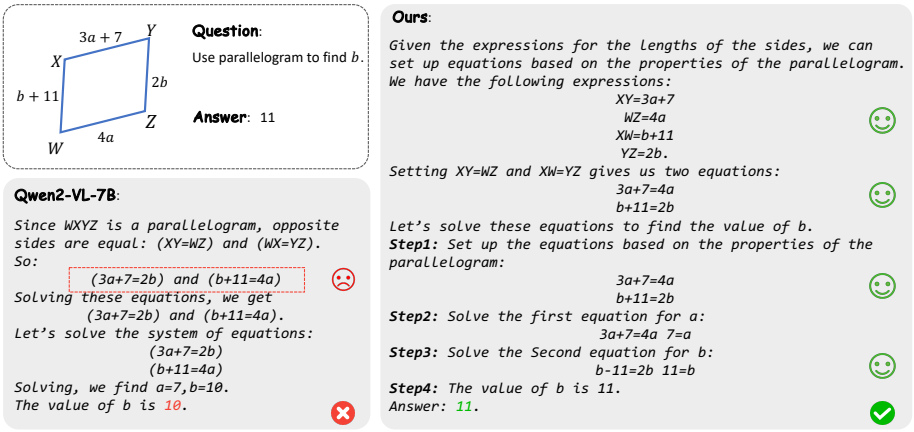

Fig. 5: Qualitative comparison of geometric reasoning.

size of 128 and a learning rate of $1e-5$. For the GRPO phase, we use the VLM-R1. framework. In the first iteration, we perform 32 rollouts per question, reducing to 8 in subsequent iterations. The temperature is set to the default value of 0.9, and the KL divergence coefficient β in Equation 2 is set to 0. All experiments are conducted on 32 NVIDIA V100-32GB GPUs.

Evaluation Settings. We evaluate **SyncLoop** on several multimodal reasoning benchmarks, including Geo-Sub from Geometry3k-test [25], MathVista [24], and MathVerse [49]. To provide a more comprehensive evaluation of model performance, we construct a specialized test subset by sampling images from Geometry3K-test that require the use of auxiliary lines to solve. This results in a focused benchmark containing 274 images. When evaluated on the original images, we denote the setting as Geo-Sub. Using the corresponding images with added auxiliary lines, we refer to the setting as Geo-Sub-Aux. Further details can be found in the **APPENDIX B**.

4.2 Main Results

In the main paper, we only present a portion of the experimental results; additional experiments are provided in the appendix (*e.g.*, **APPENDIX C, D, F**).

As shown in Table 2, **SyncLoop** demonstrates a significant improvement over Qwen2-VL-2B and Qwen2-VL-7B across three benchmarks through three iterations, with geometric reasoning accuracy increasing notably from 29.1 to 42.2 and 40.4 to 52.4, respectively. The progressive improvement observed across three iterations highlights the effectiveness of our self-improving strategy in enhancing reasoning capabilities. Furthermore, compared to prior reasoning models, our approach achieves superior performance while leveraging less than 1% of the training data, with performance approaching that of GPT-4o.

Figure 5 illustrates the model's behavior on a geometry problem involving a parallelogram. The response generated by Qwen2-VL-7B is relatively short but

Table 3: Comparison of alternating vs. consecutive GRPO training schedules with progressive model updates.

Iteration	Model	SFT	GRPO	Geo-Sub-Aux
Second	π_θ^1	✔	✔	40.7
	π_θ^1	-	✔	**48.0**
Third	π_θ^2	✔	✔	42.9
	π_θ^2	-	✔	**52.4**

Table 4: Comparison of alternating vs. consecutive GRPO training schedules from the initial model.

Iteration	Model	SFT	GRPO	Geo-Sub-Aux
Second	π_θ^0	✔	✔	**48.4**
	π_θ^0	-	✔	44.0
Third	π_θ^0	✔	✔	**46.7**
	π_θ^0	-	✔	44.4

Table 5: Comparison of error-rates based on the second iteration.

Models	Data Size ($D_t/D_{1\sim t}$)	Geo-Sub-Aux
Qwen2-VL-7B	-	38.2/40.4
Fullset	0.83K/1.48K	48.4/46.6
π_θ^1- 0.3	0.48K/0.64K	49.1/**48.0**
π_θ^1- 0.6	0.44K/0.54K	**50.9**/47.3
π_θ^1- 0.9	0.38K/0.44K	48.4/47.3
π_θ^1- 1.0	0.34K/0.35K	46.2/46.5

lacks a thorough reasoning process. It contains inaccuracies in both formula application and computational steps, leading to an incorrect final answer. In contrast, our **SyncLoop** produces a well-structured and logically coherent solution. It begins with a thorough analysis of the given conditions, followed by a step-by-step deductive process that ensures correctness at each stage, ultimately yielding the accurate final result. The findings demonstrate that our model exhibits robust and accurate reasoning capabilities in handling mathematical problems, indicating its proficiency in comprehensively understanding and correctly analyzing mathematical functions.

The influence of different iteration strategies. Table 3 presents the results of different iteration strategies. Fine-tuning π_θ^1 with additional SFT in the second iteration leads to performance degradation. In contrast, further refining the model through reinforcement learning yields improved results. As shown in Table 4, when the second iteration model is trained via SFT and RL starting from the initial model π_θ, the same performance degradation is not observed. However, the resulting model underperforms compared to when the training is warm-started from π_θ^1. Similar trends are observed in the third iteration.

The Influence of error-rate. As described in Section 3.3, we conduct an extensive parameter analysis over varying values of error-rate, which governs the complexity of data within each iteration. Table 5 shows the performance of models trained solely on data from the current iteration D_t, compared to those that also incorporate historical data $D_{1,\ldots,t-1}$. Utilizing the complete dataset does not necessarily optimize the model's reasoning capabilities. Specifically, as the complexity of the training data increases, relying exclusively on error-prone samples results in a decline performance. Based on this analysis, error-rate≥ 0.3 is

Table 6: The results on the mathematical function task.

Methods	Data Amount	FunctionQA	Function-Plot
Qwen2-VL-7B	-	58.0	58.0
SyncLoop-1^{st}	0.6k	60.0	59.0
SyncLoop-2^{nd}	0.6k	66.0	69.0
SyncLoop-1^{st} + *Function*	0.7k	61.0	60.0
SyncLoop-2^{nd} + *Function*	0.7k	72.0	71.0

Table 7: Comparison of error-rates based on the second iteration.

Methods	Data Amount	MMMU	MME-sum
Qwen2-VL-7B	-	52.0	2320.8
SyncLoop-1^{st}	0.6k	53.3	2335.3
SyncLoop-2^{nd}	0.6k	54.2	2336.2
SyncLoop-3^{rd}	0.4k	55.2	2337.1

Table 8: Training comparison between original and complex image datasets.

Model	Original Data	Complex Data
SyncLoop-1^{st}	47.27	46.9
SyncLoop-2^{nd}	45.82	48.0
SyncLoop-3^{rd}	51.27	52.4

adopted to as the basis for our final setting. Additional generalization experiments for this selection are presented in **APPENDIX D.2**.

Generalization across tasks. We also evaluate our method on mathematical function reasoning tasks to demonstrate its broader applicability, reporting performance on FunctionQA and Function-Plot from MathVista after two evolutionary rounds. As shown in the table 6, our method also achieves strong performance on other tasks.

Effectiveness demonstrated on generalization metrics. To further demonstrate the generalization capability of our method and examine potential overfitting or degradation, we evaluated our model on more comprehensive benchmarks (*e.g.*, MMMU and MME) over three iterative rounds. The results, presented in the Table 7, show that although our model is trained exclusively on geometric problems, it progressively acquires more general capabilities throughout the iterative process.

Compared to other methods [7]. Due to the authors have recently released their code and **partial** data (*e.g.* for the fine-tuning and reinforcement learning stages). We have since evaluated their model under their experimental setup (*e.g.*, besides changing the model from Qwen2.5-VL-7B to Qwen2-VL-7B.), and the results are presented in Table 9. (*e.g.*, Note that their data volume is 5k, which is significantly larger than our 0.6k.)

The Influence of training data. Table 8 presents a comparison of the 7B model trained on complexified data (complex images with complex text) versus the original data (original images with complex text). The results show that jointly complexifying both images and text leads to better performance, highlighting the importance of image complexity in training.

Ablation on Single Principle. In Table 10, we provide the second-round experimental results using a single criterion. Both the tabulated results and

Table 9: Results of other self-improvement methods.

Methods	Data Amount	Geo-Sub-Aux	MathVista
OpenVLThinker-Medium	5k (SFT 5k)	37.8	60.7
OpenVLThinker-Hard	5k (SFT 5k)	37.5	60.3
SyncLoop-1^{st}	0.6k	46.9	62.1
SyncLoop-2^{nd}	0.6k	48.0	62.4
SyncLoop-3^{rd}	0.4k	52.4	63.2

Table 10: Ablation study on different principles.

Principle	Geo-Sub-Aux
Constraints(0.2k)	47.2
New Theorems(0.1k)	46.9
Backward(0.2k)	47.3
SyncLoop-2^{nd}	48.0

Table 11: Results of training without reasoning data.

Methods	Data Amount	Geo-Sub-Aux
Qwen2-VL-7B	-	40.4
SyncLoop-1^{st}	0.6k	44.7
SyncLoop-2^{nd}	0.6k	46.6

our preliminary experimental findings indicate that data diversity is critical for performance improvement.

Abalation without reasoning process. Additionally, we have included results of training without reasoning data, as shown in the Table 11.

Mathematical function output visualization. Figure 6 presents the results of **SyncLoop** evaluation on mathematical function reasoning tasks. The findings demonstrate that our model exhibits robust and accurate reasoning capabilities in handling mathematical problems, indicating its proficiency in comprehensively understanding and correctly analyzing mathematical functions.

4.3 Deeper Analysis

Why would adding auxiliary lines to the Fig make the problem more difficult? In the proposer phase, auxiliary lines are utilized to *construct structurally more complex scenes* (*e.g.*, new) rather than to *facilitate* the solution of existing ones (*e.g.*, old). We sample 50 images from the Geo-Sub test set and add one to three lines or points per image that are unrelated to solving the problems. The accuracy of SyncLoop-1^{st}-7B drops from **45.9** to **42.1**, since the additional visual elements create extra *distraction*.

Regarding the source of data annotations. To provide a *solid mathematical foundation*, we use GPT-4o as the model for data generation. This approach is consistent with frameworks such as AlphaEvolve [29] and Voyager [41], which use large models to establish a reliable logical baseline.

Why not pair the generated hard questions with the original images? Pairing these harder question with the original images can lead to errors, such as referencing nonexistent points or lines.

Assessment of task difficulty per iteration. Figure 3 *qualitatively* illustrates the reasoning length after three iterations. Appendix Fig 3 provides a *quantitative*

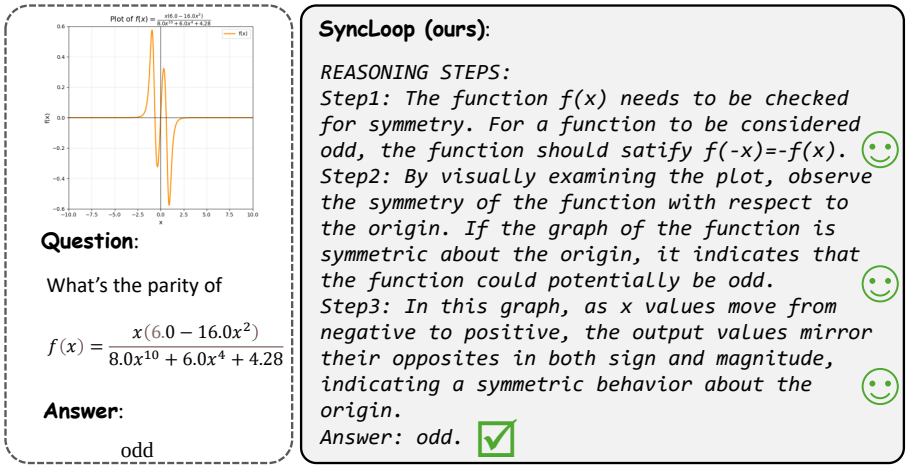

Fig. 6: Visualization of the Mathematical Function.

view of the results for the same model tested on data from the three iterations (*e.g.*, horizontal trends show rising failed cases and declining perfect scores).

5 Conclusion

In this paper, we propose a closed-loop self-improving framework (**SyncLoop**), a multi-dimensional evolution framework that operates through two interleaved loops: **a cross-modal data evolution loop** and **a data-model co-evolution loop**. Recent studies often suffer from the decoupling of textual and visual evolution, as well as a mismatch between model capability and task difficulty. To address these limitations, we introduce a synchronized co-evolution mechanism in which auxiliary guidance is employed to progressively increase the complexity of image data, while the resulting complex visuals are used to generate increasingly challenging diagram-based mathematical reasoning tasks. This ensures the joint evolution of both modalities. Furthermore, by leveraging the error-based filtering method, the model selects samples that align with the current model's blind spots or underdeveloped reasoning capabilities, thereby maintaining consistency between data complexity and model capability through iterative training. Extensive experiments demonstrate the effectiveness of our multi-iteration evolution training strategy. The different data curation strategies and iterative mechanisms not only improve performance but also offer promising directions for future research in self-improving methods.

Acknowledgments. This work is supported by National Key Research and Development Program of China (2024YFE0203100), Scientific Research Innovation Capability Support Project for Young Faculty (No.ZYGXQNJSKYCXNLZCXM-I28), Shenzhen Science and Technology Program(Grant No. QNXMA20250701093-544048) and General Embodied AI Center of Sun Yat-sen University.

Bibliography

[1] Bhattarai, M., Barron, R., Eren, M., Vu, M., Grantcharov, V., Boureima, I., Stanev, V., Matuszek, C., Valtchinov, V., Rasmussen, K., et al.: Heal: Hierarchical embedding alignment loss for improved retrieval and representation learning. arXiv preprint arXiv:2412.04661 (2024)

[2] Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H.P.D.O., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., et al.: Evaluating large language models trained on code. arXiv preprint arXiv:2107.03374 (2021)

[3] Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., et al.: Training verifiers to solve math word problems. arXiv preprint arXiv:2110.14168 (2021)

[4] Costello, C., Guo, S., Goldie, A., Mirhoseini, A.: Think, prune, train, improve: Scaling reasoning without scaling models. arXiv preprint arXiv:2504.18116 (2025)

[5] Deng, H., Zou, D., Ma, R., Luo, H., Cao, Y., Kang, Y.: Boosting the generalization and reasoning of vision language models with curriculum reinforcement learning. arXiv preprint arXiv:2503.07065 (2025)

[6] Deng, L., Liu, Y., Li, B., Luo, D., Wu, L., Zhang, C., Lyu, P., Zhang, Z., Zhang, G., Ding, E., et al.: R-cot: Reverse chain-of-thought problem generation for geometric reasoning in large multimodal models. arXiv preprint arXiv:2410.17885 (2024)

[7] Deng, Y., Bansal, H., Yin, F., Peng, N., Wang, W., Chang, K.W.: Openvlthinker: An early exploration to complex vision-language reasoning via iterative self-improvement. arXiv preprint arXiv:2503.17352 (2025)

[8] Dong, Y., Liu, Z., Sun, H.L., Yang, J., Hu, W., Rao, Y., Liu, Z.: Insight-v: Exploring long-chain visual reasoning with multimodal large language models. arXiv preprint arXiv:2411.14432 (2024)

[9] Fang, Y., Zhu, L., Lu, Y., Wang, Y., Molchanov, P., Kautz, J., Cho, J.H., Pavone, M., Han, S., Yin, H.: Vila ^{2}: Vila augmented vila. arXiv preprint arXiv:2407.17453 (2024)

[10] Feng, J., Huang, S., Qu, X., Zhang, G., Qin, Y., Zhong, B., Jiang, C., Chi, J., Zhong, W.: Retool: Reinforcement learning for strategic tool use in llms. arXiv preprint arXiv:2504.11536 (2025)

[11] Fernando, C., Banarse, D., Michalewski, H., Osindero, S., Rocktäschel, T.: Promptbreeder: Self-referential self-improvement via prompt evolution. arXiv preprint arXiv:2309.16797 (2023)

[12] Gu, A., Rozière, B., Leather, H., Solar-Lezama, A., Synnaeve, G., Wang, S.I.: Cruxeval: A benchmark for code reasoning, understanding and execution. arXiv preprint arXiv:2401.03065 (2024)

[13] Gulcehre, C., Paine, T.L., Srinivasan, S., Konyushkova, K., Weerts, L., Sharma, A., Siddhant, A., Ahern, A., Wang, M., Gu, C., et al.: Reinforced self-training (rest) for language modeling. arXiv preprint arXiv:2308.08998 (2023)

[14] Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al.: Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. arXiv preprint arXiv:2501.12948 (2025)
[15] Guo, J., Zheng, T., Li, Y., Bai, Y., Li, B., Wang, Y., Zhu, K., Neubig, G., Chen, W., Yue, X.: Mammoth-vl: Eliciting multimodal reasoning with instruction tuning at scale. In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 13869–13920 (2025)
[16] He, Y., Huang, C., Li, Z., Huang, J., Yang, Y.: Visplay: Self-evolving vision-language models from images. arXiv preprint arXiv:2511.15661 (2025)
[17] Hendrycks, D., Burns, C., Kadavath, S., Arora, A., Basart, S., Tang, E., Song, D., Steinhardt, J.: Measuring mathematical problem solving with the math dataset. arXiv preprint arXiv:2103.03874 (2021)
[18] Hu, Y., Shi, W., Fu, X., Roth, D., Ostendorf, M., Zettlemoyer, L.S., Smith, N.A., Krishna, R.: Visual sketchpad: Sketching as a visual chain of thought for multimodal language models. ArXiv **abs/2406.09403** (2024)
[19] Huang, W., Jia, B., Zhai, Z., Cao, S., Ye, Z., Zhao, F., Xu, Z., Hu, Y., Lin, S.: Vision-r1: Incentivizing reasoning capability in multimodal large language models. arXiv preprint arXiv:2503.06749 (2025)
[20] Hurst, A., Lerer, A., Goucher, A.P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A., Welihinda, A., Hayes, A., Radford, A., et al.: Gpt-4o system card. arXiv preprint arXiv:2410.21276 (2024)
[21] Jupyter, P.: Jupyter notebook (2026)
[22] Li, C., Wu, W., Zhang, H., Xia, Y., Mao, S., Dong, L., Vulić, I., Wei, F.: Imagine while reasoning in space: Multimodal visualization-of-thought. arXiv preprint arXiv:2501.07542 (2025)
[23] Liang, Y., Zhang, G., Qu, X., Zheng, T., Guo, J., Du, X., Yang, Z., Liu, J., Lin, C., Ma, L., et al.: I-sheep: Self-alignment of llm from scratch through an iterative self-enhancement paradigm. arXiv preprint arXiv:2408.08072 (2024)
[24] Lu, P., Bansal, H., Xia, T., Liu, J., Li, C., Hajishirzi, H., Cheng, H., Chang, K.W., Galley, M., Gao, J.: Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. arXiv preprint arXiv:2310.02255 (2023)
[25] Lu, P., Gong, R., Jiang, S., Qiu, L., Huang, S., Liang, X., Zhu, S.C.: Intergps: Interpretable geometry problem solving with formal language and symbolic reasoning. In: Annual Meeting of the Association for Computational Linguistics (2021)
[26] Luo, R., Zhang, H., Chen, L., Lin, T.E., Liu, X., Wu, Y., Yang, M., Li, Y., Wang, M., Zeng, P., et al.: Mmevol: Empowering multimodal large language models with evol-instruct. In: Findings of the Association for Computational Linguistics: ACL 2025. pp. 19655–19682 (2025)
[27] Lupidi, A., Gemmell, C., Cancedda, N., Dwivedi-Yu, J., Weston, J., Foerster, J., Raileanu, R., Lomeli, M.: Source2synth: Synthetic data generation and curation grounded in real data sources. arXiv preprint arXiv:2409.08239 (2024)

[28] Meng, F., Du, L., Liu, Z., Zhou, Z., Lu, Q., Fu, D., Shi, B., Wang, W., He, J., Zhang, K., et al.: Mm-eureka: Exploring visual aha moment with rule-based large-scale reinforcement learning. arXiv preprint arXiv:2503.07365 (2025)
[29] Novikov, A., Vũ, N., Eisenberger, M., Dupont, E., Huang, P.S., Wagner, A.Z., Shirobokov, S., Kozlovskii, B., Ruiz, F.J., Mehrabian, A., et al.: Alphaevolve: A coding agent for scientific and algorithmic discovery. arXiv preprint arXiv:2506.13131 (2025)
[30] OpenAI: Openai o1 system card. arXiv preprint arXiv:2412.16720 (2024)
[31] Peng, Y., Wang, X., Wei, Y., Pei, J., Qiu, W., Jian, A., Hao, Y., Pan, J., Xie, T., Ge, L., et al.: Skywork r1v: Pioneering multimodal reasoning with chain-of-thought. arXiv preprint arXiv:2504.05599 (2025)
[32] Peng, Y., Zhang, G., Zhang, M., You, Z., Liu, J., Zhu, Q., Yang, K., Xu, X., Geng, X., Yang, X.: Lmm-r1: Empowering 3b lmms with strong reasoning abilities through two-stage rule-based rl. ArXiv **abs/2503.07536** (2025)
[33] Peng, Y., Zhang, G., Zhang, M., You, Z., Liu, J., Zhu, Q., Yang, K., Xu, X., Geng, X., Yang, X.: Lmm-r1: Empowering 3b lmms with strong reasoning abilities through two-stage rule-based rl. arXiv preprint arXiv:2503.07536 (2025)
[34] Qiao, R., Tan, Q., Yang, P., Wang, Y., Wang, X., Wan, E., Zhou, S., Dong, G., Zeng, Y., Xu, Y., et al.: We-math 2.0: A versatile mathbook system for incentivizing visual mathematical reasoning. arXiv preprint arXiv:2508.10433 (2025)
[35] Rosser, J., Foerster, J.N.: Agentbreeder: Mitigating the ai safety impact of multi-agent scaffolds. arXiv preprint arXiv:2502.00757 (2025)
[36] Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y., Wu, Y., et al.: Deepseekmath: Pushing the limits of mathematical reasoning in open language models. arXiv preprint arXiv:2402.03300 (2024)
[37] Shi, W., Hu, Z., Bin, Y., Liu, J., Yang, Y., Ng, S.K., Bing, L., Lee, R.K.W.: Math-llava: Bootstrapping mathematical reasoning for multimodal large language models. In: Findings of the Association for Computational Linguistics: EMNLP 2024. pp. 4663–4680 (2024)
[38] Singh, A., Co-Reyes, J.D., Agarwal, R., Anand, A., Patil, P., Garcia, X., Liu, P.J., Harrison, J., Lee, J., Xu, K., et al.: Beyond human data: Scaling self-training for problem-solving with language models. arXiv preprint arXiv:2312.06585 (2023)
[39] Thawakar, O., Dissanayake, D., More, K., Thawkar, R., Heakl, A., Ahsan, N., Li, Y., Zumri, M., Lahoud, J., Anwer, R.M., et al.: Llamav-o1: Rethinking step-by-step visual reasoning in llms. arXiv preprint arXiv:2501.06186 (2025)
[40] Trinh, T.H., Wu, Y., Le, Q.V., He, H., Luong, T.: Solving olympiad geometry without human demonstrations. Nature **625**(7995), 476–482 (2024)
[41] Wang, G., Xie, Y., Jiang, Y., Mandlekar, A., Xiao, C., Zhu, Y., Fan, L., Anandkumar, A.: Voyager: An open-ended embodied agent with large language models. arXiv preprint arXiv:2305.16291 (2023)
[42] Wang, P., Bai, S., Tan, S., Wang, S., Fan, Z., Bai, J., Chen, K., Liu, X., Wang, J., Ge, W., et al.: Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. arXiv preprint arXiv:2409.12191 (2024)

[43] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q.V., Zhou, D., et al.: Chain-of-thought prompting elicits reasoning in large language models. Advances in neural information processing systems **35**, 24824–24837 (2022)
[44] Xu, Z., Chen, D., Ling, Z., Li, Y., Shen, Y.: Mindgym: Enhancing vision-language models via synthetic self-challenging questions. arXiv preprint arXiv:2503.09499 (2025)
[45] Yang, Y., He, X., Pan, H., Jiang, X., Deng, Y., Yang, X., Lu, H., Yin, D., Rao, F., Zhu, M., Zhang, B., Chen, W.: R1-onevision: Advancing generalized multimodal reasoning through cross-modal formalization. ArXiv **abs/2503.10615** (2025)
[46] Yang, Y., He, X., Pan, H., Jiang, X., Deng, Y., Yang, X., Lu, H., Yin, D., Rao, F., Zhu, M., et al.: R1-onevision: Advancing generalized multimodal reasoning through cross-modal formalization. arXiv preprint arXiv:2503.10615 (2025)
[47] Zelikman, E., Wu, Y., Mu, J., Goodman, N.: Star: Bootstrapping reasoning with reasoning. Advances in Neural Information Processing Systems **35**, 15476–15488 (2022)
[48] Zhang, J., Huang, J., Yao, H., Liu, S., Zhang, X., Lu, S., Tao, D.: R1-vl: Learning to reason with multimodal large language models via step-wise group relative policy optimization. arXiv preprint arXiv:2503.12937 (2025)
[49] Zhang, R., Jiang, D., Zhang, Y., Lin, H., Guo, Z., Qiu, P., Zhou, A., Lu, P., Chang, K.W., Qiao, Y., et al.: Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? In: European Conference on Computer Vision. pp. 169–186. Springer (2024)
[50] Zhang, R., Wei, X., Jiang, D., Guo, Z., Li, S., Zhang, Y., Tong, C., Liu, J., Zhou, A., Wei, B., et al.: Mavis: Mathematical visual instruction tuning with an automatic data engine. arXiv preprint arXiv:2407.08739 (2024)
[51] Zhou, H., Li, X., Wang, R., Cheng, M., Zhou, T., Hsieh, C.J.: R1-zero's" aha moment" in visual reasoning on a 2b non-sft model. arXiv preprint arXiv:2503.05132 (2025)