Advancing Multimodal Reasoning: From Optimized Cold Start to Staged Reinforcement Learning

Shuang Chen[1,*], **Yue Guo**[2,*], **Zhaochen Su**[3], **Yafu Li**[4], **Yulun Wu**[1], **Jiacheng Chen**[4], **Jiayu Chen**[2], **Weijie Wang**[1], **Xiaoye Qu**[4,†], **Yu Cheng**[5,†]

[1] Zhejiang University [2] Fudan University [3] Soochow University
[4] Shanghai AI Laboratory [5] The Chinese University of Hong Kong

Code Page: https://github.com/CSfufu/Revisual-R1

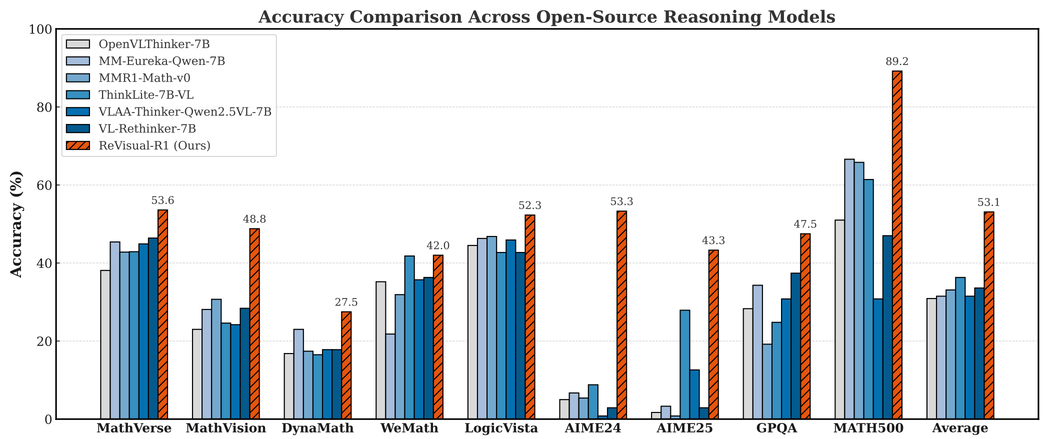


Figure 1: Overall performance across five multimodal reasoning benchmarks (MathVerse, MathVision, DynaMath, WeMath, and LogicVista) and four textual reasoning benchmarks (AIME24, AIME25, GPQA, and MATH500). Our **ReVisual-R1** achieves better performance than existing works.

Abstract

Inspired by the remarkable reasoning capabilities of Deepseek-R1 in complex textual tasks, many works attempt to incentivize similar capabilities in Multimodal Large Language Models (MLLMs) by directly applying reinforcement learning (RL). However, they still struggle to activate complex reasoning. In this paper, rather than examining multimodal RL in isolation, we delve into current training pipelines and identify three crucial phenomena: 1) Effective cold start initialization is critical for enhancing MLLM reasoning. Intriguingly, we find that initializing with carefully selected text data alone can lead to performance surpassing many recent multimodal reasoning models, even before multimodal RL. 2) Standard GRPO applied to multimodal RL suffers from gradient stagnation, which degrades training stability and performance. 3) Subsequent text-only RL training, following the multimodal RL phase, further enhances multimodal reasoning. This staged training approach effectively balances perceptual grounding and cognitive reasoning development. By incorporating the above insights and addressing multimodal RL issues, we introduce **ReVisual-R1**, achieving a new state-of-the-art among open-source 7B MLLMs on challenging benchmarks including MathVerse, MathVision, WeMath, LogicVista, DynaMath, and challenging AIME2024 and AIME2025.

*Equal contributions.
†Corresponding authors.

Preprint. Under review.

1 Introduction

Recently, the field of large language models (LLMs) has witnessed significant advancements in complex cognitive reasoning [1, 2], notably exemplified by reasoning models like DeepSeek-R1 [3]. These models successfully leveraged Reinforcement Learning (RL) to facilitate the self-emergence of intricate reasoning abilities in text-only models. Inspired by this success, a natural extension has been to apply similar RL paradigms to Multimodal Large Language Models (MLLMs) with the goal of incentivizing multimodal reasoning capabilities [4, 5, 6, 7, 8, 9, 10, 11].

Despite these promising efforts, current methods often struggle to fully unlock complex reasoning within MLLMs. This difficulty suggests that directly transplanting RL techniques from text-only domains may not adequately address the unique dynamics and requirements inherent in multimodal learning and reasoning. Motivated by this gap, our work comprehensively studies the training pipelines of the multimodal reasoning models. Through this investigation, we identify three crucial phenomena that significantly influence the efficacy of multimodal training:

First, we observe that sufficient cold start initialization is indispensable for effectively cultivating the reasoning ability of MLLMs. Conventional cold-start phases for MLLMs, often relying on simplistic visual and textual pre-training corpora, appear to insufficiently prepare models for the demands of complex problem-solving, a challenge evident in the reasoning limitations of various existing models [12, 4, 13, 14, 15]. This initial deficit critically hinders the subsequent RL stages from eliciting sophisticated, self-critical reasoning patterns. To unlock deeper deliberative reasoning in MLLMs, an enriched cold-start initialization is therefore not merely beneficial, but indispensable. Specifically, initializing with carefully selected text data that instills foundational reflective capabilities and the capacity for extended Chain-of-Thought (CoT) reasoning proves to be a powerful strategy. Intriguingly, such targeted textual initialization allows our model to surpass the multimodal reasoning performance of many recent multimodal reasoning models.

Second, we identify that the standard Group Relative Policy Optimization (GRPO) algorithm [16], commonly applied for multimodal RL, suffers from a gradient stagnation problem. This issue significantly degrades both the training stability and the ultimate performance of the multimodal RL phase. To address this fundamental limitation and improve the efficacy of multimodal RL, we propose Prioritized Advantage Distillation (PAD). PAD is designed to mitigate gradient stagnation by strategically filtering out zero-advantage samples and re-weighting informative trajectories, thereby focusing the learning process on more impactful data and improving training stability.

Third, we discover that conducting further post-training using text RL after the multimodal RL training phase further enhances multimodal reasoning ability. This observation highlights the complex interplay between visual grounding and linguistic fluency in achieving superior multimodal reasoning. Based on these three key findings, we devise a three-stage curriculum that effectively balances visual and textual competencies for multimodal reasoning. This curriculum comprises: (1) a textual cold-start to explicitly instill complex reasoning templates; (2) a multimodal RL stage to connect linguistic reasoning with visual perception; and (3) a text-only RL refinement stage aims to restore linguistic fluency and refine reasoning expression without eroding the newly gained visual grounding skills, effectively consolidating the multimodal reasoning capabilities.

To this end, we introduce ReVisual-R1, the first 7B-parameter open-source MLLM designed with this staged curriculum. As shown in Figure 1, ReVisual-R1 presents strong performance in challenging multimodal reasoning tasks, while simultaneously preserving strong general-purpose textual skills. Extensive experiments on a suite of challenging benchmarks, including MathVerse [17], MathVision [18], MathVista [19], DynaMath [20], WeMath [21], and LogicVista [22], as well as the AIME24/25 [23], GPQA [24], MATH-500 [25] benchmark, confirm that ReVisual-R1 significantly outperforms much larger public models.

To summarize, our contributions are as follows:

- We systematically investigate MLLM cold-start initialization, revealing the insufficiency of existing multimodal pre-training datasets and demonstrating that a text-centric, high-difficulty cold-start phase is crucial for unlocking complex multimodal reasoning capabilities.
- We identify and address the critical issue of gradient stagnation in GRPO for multimodal RL by proposing Prioritized Advantage Distillation (PAD), a novel technique that ensures more stable training and sample-efficient learning for MLLMs.

- We present ReVisual-R1, an open-source 7B MLLM developed through a principled three-stage curriculum. This approach uniquely cultivates deep, self-reflective reasoning and robust visual grounding, enabling ReVisual-R1 to achieve state-of-the-art performance on complex multimodal reasoning tasks, rivaling even larger or proprietary models.

2 Preliminaries

In this section, we first formulate the task setting and key concepts in the multimodal reasoning problem. Then, we describe the base training algorithm framework used in our method.

2.1 Multimodal Reasoning Formulation

In multimodal reasoning tasks, the input can be represented as $x = (v, q)$, where v denotes the visual content, and q denotes the textual query. Our work aims to guide a MLLM to generate a multi-step, self-reflective reasoning process t, which ultimately assists the model in producing a solution y that correctly answers the query based on the multimodal input.

Formally, we aim to learn a policy $\pi_\theta(y|x)$, parameterized by θ, which maps the input question space $\mathcal{X}$ to the solution space $\mathcal{Y}$. Our objective is to optimize the model parameters such that the expected reward $r(y, x)$ over the output distribution is maximized:

$$\theta^* = \arg\max_\theta \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi_\theta(y|x)}[r(y, x)] \tag{1}$$

where $\mathcal{D}$ represents the distribution of multimodal reasoning tasks. Similar to Deepseek R1 [3], in our work, we mainly use rule-based reward, $r(x, y) = 1$ if y is correct, otherwise $r(x, y) = 0$.

2.2 Group Relative Policy Optimization

Group Relative Policy Optimization (GRPO) extends traditional policy optimization methods by organizing training samples into groups and optimizing policies relative to reference models within each group, offering several advantages for training language models on complex reasoning tasks.

Formally, given a batch of samples $\mathcal{B}$, GRPO divides them into K groups $\{\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_K\}$ based on certain criteria. For each group $\mathcal{G}_i$, we maintain both a policy model π_θ and a reference model $\pi_{\theta_{\text{ref}}}$. The GRPO objective for each group is formulated as:

$$\mathbb{E}_{x\sim\mathcal{G}_i}\mathbb{E}_{y\sim\pi_\theta(y|x)}\left[\min\left(\frac{\pi_\theta(y|x)}{\pi_{\theta_{\text{ref}}}(y|x)}\hat{A}(x, y), \text{clip}\left(\frac{\pi_\theta(y|x)}{\pi_{\theta_{\text{ref}}}(y|x)}, 1-\epsilon, 1+\epsilon\right)\hat{A}(x, y)\right)\right] \tag{2}$$

where ϵ is a hyperparameter controlling the size of the trust region, and $\hat{A}(x, y)$ is the group-specific advantage function. For each input x with G generated responses $\{y_1, \ldots, y_G\}$ within a group, the advantage for response y_i is defined as:

$$\hat{A}(x, y_i) = \frac{r(x, y_i) - \text{mean}(\{r(x, y_1), \ldots, r(x, y_G)\})}{\text{std}(\{r(x, y_1), \ldots, r(x, y_G)\}) + \epsilon} \tag{3}$$

where ϵ is a small constant for numerical stability. This relative advantage is then used within a clipped surrogate objective function.

Here, $r(x, y_i)$ represents the reward for response y_i to input x. This advantage function measures how much better a specific response is compared to the average performance within its group, normalized by the group's reward variance. By using group-specific advantages, GRPO encourages the model to improve responses within each group while maintaining diversity across groups.

3 GRAMMAR: Generalized Multimodal Reasoning Dataset

In this section, we first show an intriguing finding involving the multimodal reasoners in Section 3.1, paving the way for the strategy of our data curation pipeline in Section 3.2.

Table 1: Textual and multimodal reasoning datasets source of our GRAMMAR.

Multimodal				Text-only			
Source	Samples	Source	Samples	Source	Samples	Source	Samples
FigureQA [26]	100K	Super-CLEVR [27]	30K	Big-Math-RL [28]	251K	GAIR_LIMO [29]	0.8K
MAVIS [30]	218K	TabMWP [31]	38K	Big-Math-RL-U	35K	s1K-1.1 [32]	1K
GeoQA [33]	5K	UniGeo [34]	16K	OpenThoughts [35]	114K	OpenMathR [36]	3,200K
Geometry3K [37]	2.1K	MultiMath [38]	300K	DeepMath [39]	103K	OrcaMath [40]	200K
IconQA [41]	107K			OpenR1-220k [42]	220K	NuminaMath-CoT [43]	859K

3.1 Preliminary Study of Cold Start

Currently, the leading text and multimodal reasoning models in the community [44, 3] primarily rely on cold-start training with extensive self-critique data or large-scale multimodal reinforcement learning based on verifiable rewards [3]. While cold-start training is essential for improving the reasoning capabilities of models [3], its effect on the reasoning abilities of multimodal models remains underexplored. To investigate this, we collect two open-source cold-start multimodal datasets, Vision-R1 [4] and R1-One-Vision [12], along with two cold-start textual datasets, DeepMath [39] and OpenR1-Math [42]. Then we randomly sample 40,000 instances from these datasets to fine-tune Qwen2.5-VL-7B-Instruct [45]. The fine-tuned models are subsequently evaluated on multimodal reasoning benchmarks (MathVerse and MathVision) as well as text reasoning benchmarks (AIME24 and Math500). The experimental outcomes and average performance enhancements from the multimodal and textual cold-start datasets are illustrated in Figure 2.

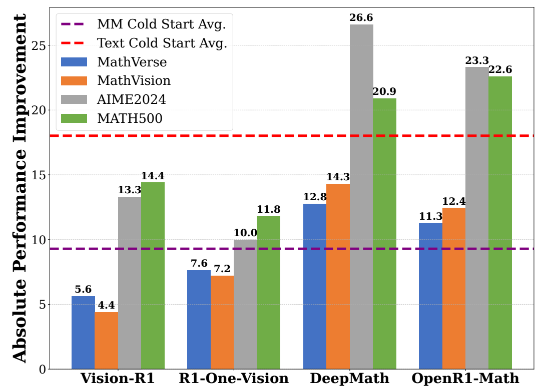


Figure 2: Absolute performance improvement on Qwen2.5-VL-7B-Instruct across textual and multimodal reasoning tasks. The red and purple dashed lines represent the average absolute gains of VisionR1/R1-One-Vision and DeepMath/OpenR1-Math over the baseline, respectively, across four reasoning tasks.

The results in Figure 2 reveal that models trained with text-only cold start data exhibit substantial improvements in both textual and multimodal reasoning tasks. In contrast, models trained solely on multimodal datasets, such as Vision-R1 and R1-One-Vision, show limited gains in both multimodal and textual reasoning. This suggests that the complexity and patterns presented by textual cold start data may better stimulate the models' reasoning capabilities.

To further investigate this observation, we perform an analysis using a subset of 100 examples sampled from the Vision-R1 [4] and DeepMath [39] datasets. Specifically, we analyze the response lengths and pass rates of the doubao-1.5-thinking-pro-vision model [46] on these samples. Responses to textual prompts from DeepMath averaged 8,207.76 tokens, which is substantially longer than the 821.48 tokens generated in response to multimodal prompts from Vision-R1. Moreover, the pass rate for Vision-R1 is 96.00%, whereas DeepMath achieve a pass rate of only 75.0%. These findings further indicate that current multimodal cold start datasets may lack sufficient complexity to inspire advanced reasoning capabilities of reasoning models. It indicates that existing multimodal cold start datasets in the community may not be sufficiently challenging to enhance the complex reasoning capabilities and generalization of policy models, *i.e.,* they may not provide effective initial strategies during cold start training. Therefore, it is desirable to develop a data curation pipeline for multimodal policy model to improve the generalization capabilities.

3.2 Data Curation

Informed by Section 3.1 regarding the variability in open-source reasoning data [47], [48], [49], [50], [51], we develop GRAMMAR, a new dataset designed to enhance the generalization of reasoning capabilities in multimodal models. GRAMMAR comprises 47k diverse textual thought samples with explicit reasoning paths, augmented by 31k complex textual examples and 21k multimodal questions with ground truth annotations suitable for rule-based reinforcement learning.

The construction of GRAMMAR involved a multi-stage curation pipeline. We begin by amassing a wide array of open-source reasoning data, spanning various difficulty levels (details in Table 1). This initial collection underwent rule-based filtering to ensure answer verifiability, excluding items like proof problems and those with difficult-to-verify ground truths. Subsequently, Qwen2.5-VL-7B-Instruct was employed for initial pruning of overly simple or complex questions. Qwen2.5-VL-32B-Instruct is then used to assess the remaining samples to classify them into ten difficulty levels. To maximize data diversity and minimize redundancy, we encoded questions using NV-Embedding-V2 [52], applied HDBSCAN [53] for clustering, assigned topics to clusters via Qwen2.5-7B-Instruct, and performed balanced sampling across both topics and difficulty strata.

4 Staged Reinforcement Optimization (SRO)

Our data investigations (Section 3.1) and the curation of the GRAMMAR dataset highlight the necessity of high-quality, reasoning-focused data for developing advanced MLLM capabilities. Building directly on these data-centric foundations, we introduce Staged Reinforcement Optimization (SRO), a framework designed to systematically cultivate robust reasoning and diverse competencies in MLLMs. SRO achieves this through a sequence of distinct learning phases, each tailored to address specific training challenges and leverage appropriate components of the GRAMMAR dataset. This section details the SRO architecture and its constituent techniques.

4.1 Stage 1: Multimodal RL

After the cold start training, the SRO framework commences with a dedicated Multimodal Reinforcement Learning (MRL) phase. This initial stage is pivotal for enabling the MLLM to ground textual concepts in visual information and execute cross-modal reasoning, primarily using the multimodal samples from our GRAMMAR dataset. We employ GRPO as the core RL algorithm for this phase. To ensure stable and effective learning, particularly when dealing with complex tasks and potentially sparse rewards common in multimodal settings, we integrate two key enhancements: Prioritized Advantage Distillation (PAD) to improve gradient quality by addressing specific GRPO limitations, and an efficient-length reward function for more controlled and stable response generation.

4.1.1 Prioritized Advantage Distillation (PAD)

In this paper, we discover a significant challenge when applying GRPO in complex multimodal settings is "Gradient Stagnation". This phenomenon refers to a reduction in learning efficacy due to a predominance of near-zero advantage estimates, which is particularly acute when dealing with sparse binary rewards. Essentially, if entire groups of generated responses yield uniform rewards (e.g., all correct or all incorrect), the resulting advantage signals become null, leading to zero policy gradients and thereby halting learning for those samples. This issue, also noted in concurrent works [13, 54], can severely impede training progress. To specifically counteract gradient stagnation and enhance the efficiency of GRPO, we introduce Prioritized Advantage Distillation (PAD). PAD refines the training process by strategically focusing updates on the most informative samples within each batch, namely those exhibiting significant, non-zero advantage signals. This approach optimizes computational resource allocation and promotes more consistent learning. The PAD mechanism, detailed below, operates on each batch after initial advantage estimation:

- **Per-Sequence Advantage Calculation:** Compute the absolute advantage $|\tilde{A}_i|$ for each sequence i in the original batch $\mathcal{B}$, representing its learning signal magnitude.
- **Effective Sample Filtering:** Form an "effective set" $\mathcal{E}$ by selecting sequences i whose absolute advantage $|\tilde{A}_i|$ falls within a specified informative range $[T_{low}, T_{high}]$. Critically, $T_{low} > 0$ filters out stagnant (near-zero advantage) samples, ensuring that candidates for sub-sampling provide potentially useful learning signals.
- **Prioritized Sub-sampling from Effective Set:** From this effective set $\mathcal{E}$, $k' = \min(\rho|\mathcal{B}|, |\mathcal{E}|)$ sequences are drawn to form a distilled mini-batch. Selection is prioritized based on sequences' absolute advantages ($\hat{A}_i$ for $i \in \mathcal{E}$), with the probability for selecting sequence i determined by a temperature-controlled Softmax distribution:

$$\Pr(i \text{ is selected} \mid i \in \mathcal{E}) = \frac{\exp(\hat{A}_i/\tau)}{\sum_{j\in\mathcal{E}} \exp(\hat{A}_j/\tau)} \tag{4}$$

The temperature τ governs sampling concentration and is typically decayed during training (e.g., linearly from 1.0 to 0.3) to shift from exploration towards exploitation. This enriches the mini-batch with the most informative samples from $\mathcal{E}$.

PAD thus directly counteracts gradient stagnation via a dual mechanism: first, by filtering out stagnant samples, and second, by prioritizing updates using informative, non-zero advantages from the remaining set. This selective optimization of the learning process ensures efficient computational resource allocation towards high-value samples. Consequently, PAD leads to enhanced training stability, improved learning efficiency, and more effective acquisition of complex reasoning skills, particularly in challenging scenarios with sparse or binary rewards.

4.1.2 Efficient-Length Reward Function

While complex reasoning tasks often necessitate extended outputs, excessively long sequences can be suboptimal [55, 56]. Therefore, in this paper, besides the primary reward signal for task accuracy, we introduce an efficient-length reward to regulate the verbosity of generated responses. Specifically, let L_y be the token length of the generated sequence and L_{budget} be a pre-defined target length budget. A raw reward score, R_{raw}, is first computed as a linear function of the deviation:

$$R_{\text{raw}} = \alpha(L_{\text{budget}} - L_y) + \delta \tag{5}$$

where α is a positive scaling factor controlling the penalty magnitude for length deviation, and δ is a baseline reward (e.g., 0.5) assigned when $L_y = L_{\text{budget}}$. The final Efficient-Length Reward, R_{len}, is obtained by clipping R_{raw} to the range $[0, 1]$:

$$R_{\text{len}}(L_y, L_{\text{budget}}, \alpha, \delta) = \max(0.0, \min(1.0, R_{\text{raw}})) \tag{6}$$

In our experiments, we typically use a small α (e.g., 0.005) to ensure a gentle slope for the reward adjustment. This formulation provides a continuous and bounded reward signal. Specifically, a reward of δ is assigned if the generated length L_y matches the budget L_{budget}. The reward proportionally increases (up to a maximum of 1.0) as L_y becomes shorter than L_{budget}, thereby encouraging conciseness. This Efficient-Length Reward guides the model towards producing responses that are not only accurate but also parsimonious, without prematurely curtailing potentially valuable, longer reasoning paths, thus fostering more robust and efficient learning of complex multimodal reasoning.

4.2 Stage 2: Textual RL

While MRL is indispensable for grounding reasoning across visual and textual inputs, intensive MRL training can inadvertently lead to a decline in purely textual capabilities, which we define as "textual capability decay". To further elevate the model's capacity for sophisticated abstract reasoning, we integrate a subsequent Textual Reinforcement Learning (TRL) phase. This stage aims to achieve both robust linguistic fluency and advanced reasoning. Linguistic fluency is restored and enhanced by fine-tuning on high-quality, text-only corpora focused on instruction-following and conversational abilities. Simultaneously, to foster advanced reasoning, the TRL phase exposes the model to complex, text-centric problem-solving tasks. This compels the model to refine and generalize intricate reasoning patterns, articulate multi-step thought processes with greater clarity, and master linguistic nuances essential for higher-order cognition.

For policy optimization during this TRL phase, we employ GRPO, augmented with our proposed PAD mechanism for efficient sample utilization. The reward function is multifaceted, designed to promote linguistic excellence, which encompasses both fluency and conciseness. Conciseness, in particular, is also encouraged by the efficient-length reward (Equation 6).

5 Experiments

5.1 Experiments Setup

Datasets The training of ReVisual-R1 follows our proposed three-stage methodology, utilizing carefully curated datasets for each phase. The cold-start phase employed approximately 40k pure text entries focused on establishing foundational language understanding. Subsequently, the Multimodal Reinforcement Learning (MRL) phase used approximately 26k diverse multimodal entries from our GRAMMAR dataset (see Section 3.2 for data curation details) to develop cross-modal reasoning.

Finally, the text-based RL (TRL) phase consisted of approximately 30k text entries designed to refine nuanced understanding and generation capabilities.

Benchmarks We evaluate ReVisual-R1 on a comprehensive suite of benchmarks, selected to test diverse reasoning skills. For visual-mathematical reasoning, we employed MathVerse [17], MathVision [18], WeMath [21], and DynaMath [20]. Broader multimodal reasoning was assessed using MathVista [19] and LogicVista [22]. Performance on challenging text-based mathematical reasoning was measured on AIME24/25 [23] and MATH-500 [25], while general question answering was tested with GPQA [24]. We report pass@1 accuracy for performance on the evaluated benchmarks except AIME. For the AIME24/25 benchmark, performance is measured using pass@32 accuracy.

Baselines The performance of ReVisual-R1 is benchmarked against several categories of models, with detailed results presented in Table 2. These baselines include: (1) leading closed-source models such as doubao-1.5-vision-pro-32k [57], OpenAI-GPT-4o [58], Claude-3.7-Sonnet [59], and Gemini-2.0-Flash [60]). (2) diverse open-source general-purpose MLLMs like the InternVL3-8B [61], LLaVA-OneVision [62], and the Qwen2.5-VL-7B [45]; and (3) specialized open-source reasoning MLLMs, including VLAA-Thinker-7B [63], OpenVLThinker-7B [14], MMR1-Math-v0 [64], MM-Eureka [65], and VL-Rethinker-7B [13].

Implementation Our ReVisual-R1 model is based on the Qwen-2.5-VL-7B-Instruct model. Its training comprised three distinct stages. The process begins with a cold-start phase utilizing LLaMA Factory [66] and pure text data to establish foundational language understanding. Following this, Multimodal Reinforcement Learning (MRL) is implemented using Easy R1 [67]. In this stage, the GRPO Kullback-Leibler (KL) divergence constraint is omitted to encourage broader policy exploration. The final stage involves Text-based Reinforcement Learning (TRL), also conducted via Easy R1. During TRL, the vision tower is frozen to concentrate learning on textual reasoning, and a small KL penalty is incorporated alongside entropy annealing to enhance training stability. All experiments are conducted on a setup of 8 NVIDIA A100-80G GPUs. Detailed prompt settings and training hyperparameters are provided in the Appendix.

5.2 Main Results

As shown in Table 2, our model achieves state-of-the-art (SOTA) results on math-related benchmarks among open-source reasoning multimodal models and even beats some commercial large MLLM. These results prove the effectiveness of our method beneficial to the open-source community.

Specifically, ReVisual-R1 achieves an impressive average score of 53.1%, a significant improvement of +16.8 percentage points over the previous open-source SOTA average. Specifically, ReVisual-R1 secures the top position among open-source contenders in nine out of ten individual benchmarks: MathVerse (+5.4% Δ), MathVision (+13.9% Δ), DynaMath (+9.8% Δ), WeMath (+0.2% Δ), LogicVista (+9.6% Δ), AIME24 (+44.6% Δ), AIME25 (+15.4% Δ), GPQA (+10.1% Δ), and MATH500 (+23.4% Δ). The most substantial gains are observed in the challenging AIME24, MATH500, and AIME25 benchmarks, underscoring ReVisual-R1's advanced mathematical and inferential reasoning prowess. On MathVista, ReVisual-R1 achieves the second-best open-source score, narrowly trailing the leading open-source model by only -0.6%.

When compared to closed-source commercial models, ReVisual-R1 also exhibits highly competitive performance. For instance, its average score (53.1%) surpasses that of OpenAI-GPT-4o (41.6%). On specific demanding benchmarks such as MATH500, ReVisual-R1 (89.2%) outperforms both doubao-1.5-vision-pro-32k (85.2%) and OpenAI-GPT-4o (74.6%). Similarly, on AIME24 and AIME25, ReVisual-R1 demonstrates substantial leads over these commercial offerings. While some closed-source models like doubao-1.5-vision-pro-32k show a higher overall average (55.8%), ReVisual-R1's ability to outperform them on several key reasoning tasks highlights its specialized strengths.

Collectively, these results validate the efficacy of our proposed training method, including the structured three-stage curriculum and enhancements like Prioritized Advantage Distillation. The strong performance of ReVisual-R1 not only pushes the boundaries for open-source 7B MLLMs in complex reasoning but also provides a valuable contribution to the broader research community.

Table 2: Performance comparison of various MLLMs on diverse out-of-domain benchmarks. The best scores are **bold**; the second best are underlined (among open-source models). **AIME24** and **AIME25** results are averaged over eight independent inference runs to reduce score variance.

Model	Multimodal Reasoning Benchmarks						Textual Reasoning Benchmarks				
	MathVerse	MathVision	MathVista	DynaMath	WeMath	LogicVista	AIME24	AIME25	GPQA	MATH500	Avg.
Close-Source											
doubao-1.5-vision-pro-32k	64.7	51.5	78.6	44.9	64.2	65.7	26.7	20.0	56.1	85.2	55.8
OpenAI-GPT-4o	40.6	31.1	59.9	34.5	42.9	64.4	9.3	8.3	49.9	74.6	41.6
Claude-3.7-Sonnet	52.0	41.3	66.8	39.7	58.2	49.3	20.0	13.3	61.1	80.4	48.2
Gemini-2.0-Flash	43.6	47.8	70.4	42.1	47.4	52.3	33.3	36.7	35.4	69.0	47.8
Open-Source General Models											
InternVL-3-8B	33.9	28.6	70.5	23.0	37.5	43.6	3.3	6.7	34.8	75.2	35.7
LLaVA-OV-7B	17.6	17.6	62.6	9.0	17.7	32.0	0.0	0.0	0.1	45.2	20.2
Qwen-2.5-VL-7B	38.7	26.6	69.1	12.6	24.5	35.6	10.0	6.7	32.8	67.2	32.4
Open-Source Reasoning Models											
OpenVLThinker-7B	38.1	23.0	65.3	16.8	35.2	44.5	5.0	1.7	28.3	51.0	30.9
MM-Eureka-Qwen-7B	45.4	28.1	72.6	23.0	21.8	46.3	6.7	3.3	34.3	66.6	31.5
MMR1-Math-v0	42.8	30.7	69.8	17.4	31.9	46.8	5.4	0.8	19.2	65.8	33.1
ThinkLite-7B-VL	42.9	24.6	71.6	16.5	41.8	42.7	8.8	27.9	24.8	61.4	36.3
VLAA-Thinker-7B	44.9	24.2	71.7	17.8	35.7	45.9	0.8	12.6	30.8	30.8	31.5
VL-Rethinker-7B	46.4	28.4	**73.7**	17.8	36.3	42.7	2.9	2.9	37.4	47.0	33.6
Our model											
ReVisual-R1	**53.6**	**48.8**	73.1	**27.5**	**42.0**	**52.3**	**53.3**	**43.3**	**47.5**	**89.2**	**53.1**
Δ (Ours–Open SoTA)	+7.2	+18.1	-0.6	+4.5	+0.2	+5.5	+44.5	+15.4	+10.1	+23.4	+16.8

Table 3: Ablation study of different training stage combinations applied to the ReVisual-R1 model, building upon a Cold Start. Best results per column are **bold** and second-best are underlined.

Training Stages Applied	MathVerse	MathVision	MathVista	DynaMath	WeMath	LogicVista	Avg
Cold Start (CS) only	51.9	47.9	70.5	26.5	35.8	50.1	47.1
CS + MRL	50.9	47.6	71.9	25.7	38.8	51.2	47.7
CS + TRL	47.3	47.3	71.0	25.2	33.7	44.7	44.9
CS + MRL + TRL	**53.6**	**48.8**	**73.1**	**27.5**	**42.0**	**52.3**	**49.6**
CS + TRL + MRL	47.5	48.0	70.3	24.2	35.0	48.2	45.5

5.3 Ablation Study

5.3.1 Ablation on SRO

To validate our Staged Reinforcement Optimization (SRO) framework, we conducted ablation studies on different combinations of Multimodal RL (MRL) and Text-based RL (TRL) phases, all building upon our optimized text-centric cold-start (CS). This investigation, with results detailed in Table 3, aimed to empirically determine the most effective sequence for cultivating a balance between robust visual grounding and advanced textual proficiency.

The empirical evidence strongly supports our proposed CS + MRL + TRL (ReVisual-R1-MTR) sequence, which consistently yielded the highest average performance (49.6 Avg). This outcome affirms our core hypothesis: an initial MRL phase dedicated to establishing strong visual grounding, followed by a TRL phase to refine textual fluency and abstract reasoning, is crucial for developing superior multimodal capabilities without degrading the foundational cross-modal understanding.

In a more detailed analysis, the CS + MRL only model (47.7 Avg), while performing well on visually intensive tasks such as MathVista (71.9), did not reach the overall performance of the full MTR sequence. This suggests that MRL, while vital, can lead to a "textual capability decay", which the subsequent TRL stage effectively mitigates. The alternative SRO ordering, CS + TRL + MRL (45.5 Avg), also proved less effective than our MTR approach. This finding indicates that establishing strong visual grounding before intensive textual refinement allows for more synergistic learning, where the TRL phase can enhance reasoning that is already connected across modalities.

Table 4: Ablation results demonstrating the impact of Prioritized Advantage Distillation (PAD) and its core components. Performance metrics are reported for various mathematical reasoning benchmarks, averaged across datasets. Best results per column are **bold** and second-best are <u>underlined</u>.

Model Configuration	Strategy	MathVerse	MathVision	MathVista	DynaMath	WeMath	LogicVista	Avg
ReVisual-R1-Stage2	PAD	**50.9**	**47.6**	**71.9**	<u>25.7</u>	**38.8**	**51.2**	**47.7**
w/o PAD Components:								
- Full PAD (Baseline)	GRPO-Baseline	47.6	45.8	68.8	25.2	34.8	48.6	45.1
- No Prioritized Sub-sampling	GRPO-Filter	47.7	<u>46.7</u>	<u>71.2</u>	25.5	35.1	<u>49.7</u>	46.0
- No Effective Sample Filtering	Random-Sampling	<u>47.9</u>	46.4	70.7	**26.1**	<u>37.1</u>	49.3	<u>46.2</u>

In conclusion, these ablation results provide compelling justification for the MRL-then-TRL ordering within our SRO framework. This strategic sequencing first grounds the model multimodally and then sharpens its linguistic and abstract reasoning faculties, culminating in a more comprehensively capable and high-performing MLLM.

5.3.2 Ablation study on PAD

We conduct ablation studies to evaluate Prioritized Advantage Distillation (PAD), examining its overall efficacy, the contribution of its components, and its sensitivity to key hyperparameters.

To assess PAD's impact, its full implementation was compared against GRPO-Baseline, GRPO-Filter-only, and Random-Sampling strategies. Table 4 demonstrates that full PAD achieved superior performance on mathematical reasoning benchmarks, highlighting the importance of its core components: effective sample filtering and prioritized sub-sampling. Training dynamics (Figure 3) further corroborate PAD's effectiveness, with its sampling strategy yielding higher reward accuracy and faster convergence, thereby enhancing learning efficiency.

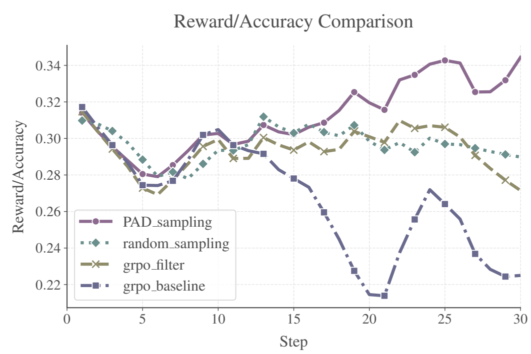


Figure 3: Training reward of different strategies: PAD-sampling, GRPO-Baseline, GRPO-Filter, and Random-Sampling. PAD consistently reaches higher accuracy and converges faster.

5.3.3 Ablation on Efficient-Length Reward

In this paper, in multimodal RL, we devise an Efficient-Length Reward. As depicted in Figure 4, the Efficient-Length Reward significantly impacts training. The regularized model maintained stable and higher reward accuracy (Fig. 4a) and consistently low entropy (Fig. 4b). In contrast, the baseline model suffered an accuracy decline and a dramatic entropy increase. Furthermore, the Efficient-Length Reward helped maintain a stable mean response length (Fig. 4c) and a low clip ratio (Fig. 4d), unlike the baseline, which exhibited uncontrolled growth in response length and a consequently higher clip ratio.

In summary, the Efficient-Length Reward is crucial for stabilizing training, preventing accuracy degradation, maintaining low model entropy, and controlling verbosity.

6 Related Work

6.1 Multimodal Large Language Model

Sophisticated reasoning in Multimodal Large Language Models (MLLMs), inspired by Large Language Model (LLM) advancements, is a key research area. Initial progress involved integrating visual encoders with LLMs (e.g., CLIP [68], MiniGPT4 [69]), with notable advancements driven by visual instruction tuning in series like LLaVA [70] [71]. While leading closed-source models (e.g., GPT-o3 [72], Kimi-VL [73]) excel at long Chain-of-Thought (CoT) reasoning, open-source contributions have focused on CoT adaptations [74, 75, 76] and Supervised Fine-Tuning (SFT) with reasoning

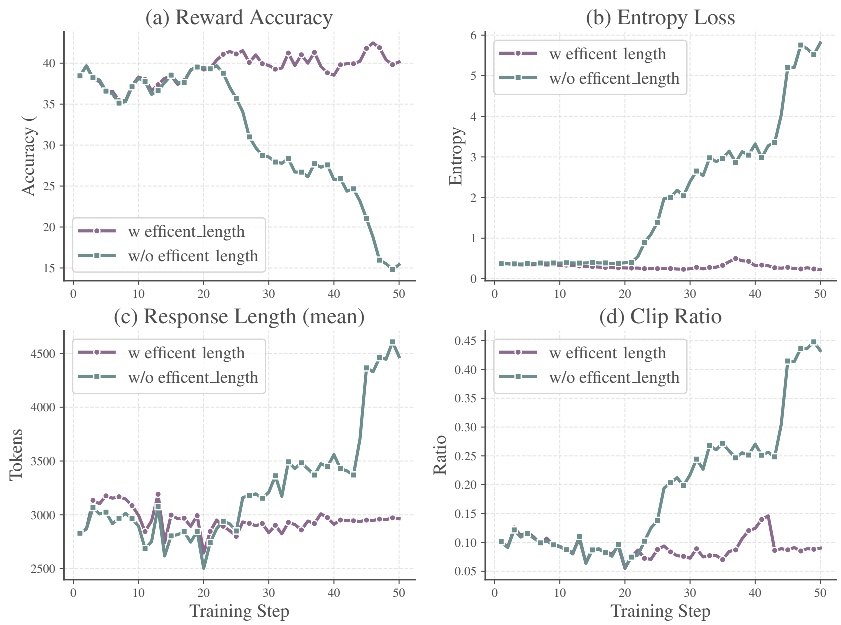


Figure 4: Training dynamics comparing models with (purple lines, "w efficient-length") and without (green lines, "w/o efficient-length") the Efficient-Length Reward.

traces [77] [78]. The open-sourcing of DeepSeek-R1 [3] has further spurred Reinforcement Learning (RL) applications for visual reasoning [4] [6] and specialized domains like mathematical reasoning. Nevertheless, many MLLMs reasoning models [13] [14] [12] [15] are limited by generating relatively short responses, which often curtail genuine reflection, thorough visual exploration, and consequently, deep multimodal reasoning. Our work, in contrast, introduces a novel framework to enable MLLMs to generate significantly longer, reflective responses with explicit visual references, thereby facilitating long CoT reasoning to unlock more comprehensive multimodal reasoning capabilities.

6.2 Reinforcement Learning in Reasoning

Reinforcement learning (RL) significantly advances LLM reasoning, evolving from methods like Reinforcement Learning from Human Feedback (RLHF) [79, 80]. Current LLM research further explores direct RL fine-tuning, specialized cold-start datasets for long-form reasoning, and advanced algorithms like Group Relative Policy Optimization (GRPO) [16] and its refinements (e.g., DAPO [54], DR.GRPO [81], GPG [82]) to elicit deeper reasoning. However, RL application to multimodal reasoning in MLLMs is nascent. Initial MLLMs efforts focus on subdomains like math reasoning [5, 9] or generative reward models [83], often utilizing data from commercial models. Nonetheless, successes such as DeepSeek-R1's [3] rule-based RL are spurring similar MLLMs investigations, indicating growing interest in RL for unlocking sophisticated multimodal reasoning.

7 Conclusion

This paper introduces ReVisual-R1, a 7B open-source MLLM designed to address prevalent challenges in cultivating sophisticated multimodal reasoning. By systematically integrating a strategic, high-difficulty text-only cold-start phase for foundational reasoning, a Multimodal RL stage employing GRPO stabilized by our novel Prioritized Advantage Distillation (PAD) mechanism and guided by rule-based rewards including an Efficient-Length Reward, and a final TextRL refinement phase, our structured three-stage curriculum demonstrates that thoughtful data strategy and targeted algorithmic optimizations are pivotal. ReVisual-R1 achieves state-of-the-art performance among open-source 7B models on a suite of challenging visuo-mathematical and reasoning benchmarks. This work underscores that careful curriculum design and algorithmic enhancements, rather than sheer model scale, can unlock robust, self-reflective multimodal reasoning.

References

[1] Weihao Zeng, Yuzhen Huang, Wei Liu, Keqing He, Qian Liu, Zejun Ma, and Junxian He. 7b model and 8k examples: Emerging reasoning with reinforcement learning is both effective and efficient.

[2] Jianhao Yan, Yafu Li, Zican Hu, Zhi Wang, Ganqu Cui, Xiaoye Qu, Yu Cheng, and Yue Zhang. Learning to reason under off-policy guidance. *arXiv preprint arXiv:2504.14945*, 2025.

[3] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[4] Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models. *arXiv preprint arXiv:2503.06749*, 2025.

[5] Fanqing Meng, Lingxiao Du, Zongkai Liu, Zhixiang Zhou, Quanfeng Lu, Daocheng Fu, Botian Shi, Wenhai Wang, Junjun He, Kaipeng Zhang, et al. Mm-eureka: Exploring visual aha moment with rule-based large-scale reinforcement learning. *arXiv preprint arXiv:2503.07365*, 2025.

[6] Yuhao Dong, Zuyan Liu, Hai-Long Sun, Jingkang Yang, Winston Hu, Yongming Rao, and Ziwei Liu. Insight-v: Exploring long-chain visual reasoning with multimodal large language models. *arXiv preprint arXiv:2411.14432*, 2024.

[7] Jiashuo Sun, Jihai Zhang, Yucheng Zhou, Zhaochen Su, Xiaoye Qu, and Yu Cheng. Surf: Teaching large vision-language models to selectively utilize retrieved information. *arXiv preprint arXiv:2409.14083*, 2024.

[8] Peng Xia, Kangyu Zhu, Haoran Li, Tianze Wang, Weijia Shi, Sheng Wang, Linjun Zhang, James Zou, and Huaxiu Yao. Mmed-rag: Versatile multimodal rag system for medical vision language models. In *The Thirteen International Conference on Learning Representations*.

[9] Yingzhe Peng, Gongrui Zhang, Miaosen Zhang, Zhiyuan You, Jie Liu, Qipeng Zhu, Kai Yang, Xingzhong Xu, Xin Geng, and Xu Yang. Lmm-r1: Empowering 3b lmms with strong reasoning abilities through two-stage rule-based rl. *arXiv preprint arXiv:2503.07536*, 2025.

[10] Zhaochen Su, Linjie Li, Mingyang Song, Yunzhuo Hao, Zhengyuan Yang, Jun Zhang, Guanjie Chen, Jiawei Gu, Juntao Li, Xiaoye Qu, et al. Openthinkimg: Learning to think with images via visual tool reinforcement learning. *arXiv preprint arXiv:2505.08617*, 2025.

[11] Chuming Shen, Wei Wei, Xiaoye Qu, and Yu Cheng. Satori-r1: Incentivizing multimodal reasoning with spatial grounding and verifiable rewards. *arXiv preprint arXiv:2505.19094*, 2025.

[12] 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.

[13] Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhu Chen. Vl-rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. *arXiv preprint arXiv:2504.08837*, 2025.

[14] Yihe Deng, Hritik Bansal, Fan Yin, Nanyun Peng, Wei Wang, and Kai-Wei Chang. Open-vlthinker: An early exploration to complex vision-language reasoning via iterative self-improvement, 2025.

[15] Liang Chen, Lei Li, Haozhe Zhao, Yifan Song, and Vinci. R1-v: Reinforcing super generalization ability in vision-language models with less than $3. `https://github.com/Deep-Agent/R1-V`, 2025. Accessed: 2025-02-02.

[16] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[17] 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.

[18] Ke Wang, Junting 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.

[19] 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.

[20] 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, 2025.

[21] Runqi Qiao, Qiuna Tan, Guanting Dong, Minhui Wu, Chong Sun, Xiaoshuai Song, Zhuoma GongQue, Shanglin Lei, Zhe Wei, Miaoxuan Zhang, Runfeng Qiao, Yifan Zhang, Xiao Zong, Yida Xu, Muxi Diao, Zhimin Bao, Chen Li, and Honggang Zhang. We-math: Does your large multimodal model achieve human-like mathematical reasoning?, 2024.

[22] Yijia Xiao, Edward Sun, Tianyu Liu, and Wei Wang. Logicvista: Multimodal llm logical reasoning benchmark in visual contexts, 2024.

[23] Jia Li, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Huang, Kashif Rasul, Longhui Yu, Albert Q. Jiang, Ziju Shen, et al. Numinamath: The largest public dataset in ai4maths with 860k pairs of competition math problems and solutions. `https://huggingface.co/datasets/Numinamath`, 2024. Hugging Face repository, 13:9.

[24] David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024.

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

[26] Samira Ebrahimi Kahou, Vincent Michalski, Adam Atkinson, Ákos Kádár, Adam Trischler, and Yoshua Bengio. Figureqa: An annotated figure dataset for visual reasoning. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Workshop Track Proceedings*. OpenReview.net, 2018.

[27] Zhuowan Li, Xingrui Wang, Elias Stengel-Eskin, Adam Kortylewski, Wufei Ma, Benjamin Van Durme, and Alan L. Yuille. Super-clevr: A virtual benchmark to diagnose domain robustness in visual reasoning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 14963–14973. IEEE, 2023.

[28] Alon Albalak, Duy Phung, Nathan Lile, Rafael Rafailov, Kanishk Gandhi, Louis Castricato, Anikait Singh, Chase Blagden, Violet Xiang, Dakota Mahan, and Nick Haber. Big-math: A large-scale, high-quality math dataset for reinforcement learning in language models. *CoRR*, abs/2502.17387, 2025.

[29] Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. Limo: Less is more for reasoning, 2025.

[30] Renrui Zhang, Xinyu Wei, Dongzhi Jiang, Ziyu Guo, Yichi Zhang, Chengzhuo Tong, Jiaming Liu, Aojun Zhou, Shanghang Zhang, Peng Gao, and Hongsheng Li. MAVIS: mathematical visual instruction tuning with an automatic data engine. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025.

[31] Pan Lu, Liang Qiu, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, Tanmay Rajpurohit, Peter Clark, and Ashwin Kalyan. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[32] Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling, 2025.

[33] Jiaqi Chen, Jianheng Tang, Jinghui Qin, Xiaodan Liang, Lingbo Liu, Eric P. Xing, and Liang Lin. Geoqa: A geometric question answering benchmark towards multimodal numerical reasoning. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 513–523. Association for Computational Linguistics, 2021.

[34] 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 Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 3313–3323. Association for Computational Linguistics, 2022.

[35] OpenThoughts Team. Open Thoughts. https://open-thoughts.ai, January 2025.

[36] Ivan Moshkov, Darragh Hanley, Ivan Sorokin, Shubham Toshniwal, Christof Henkel, Benedikt Schifferer, Wei Du, and Igor Gitman. Aimo-2 winning solution: Building state-of-the-art mathematical reasoning models with openmathreasoning dataset. *arXiv preprint arXiv:2504.16891*, 2025.

[37] Pan Lu, Ran Gong, Shibiao Jiang, Liang Qiu, Siyuan Huang, Xiaodan Liang, and Song-Chun Zhu. Inter-gps: Interpretable geometry problem solving with formal language and symbolic reasoning. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 6774–6786. Association for Computational Linguistics, 2021.

[38] Shuai Peng, Di Fu, Liangcai Gao, Xiuqin Zhong, Hongguang Fu, and Zhi Tang. Multimath: Bridging visual and mathematical reasoning for large language models. *CoRR*, abs/2409.00147, 2024.

[39] Zhiwei He, Tian Liang, Jiahao Xu, Qiuzhi Liu, Xingyu Chen, Yue Wang, Linfeng Song, Dian Yu, Zhenwen Liang, Wenxuan Wang, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning, 2025.

[40] Arindam Mitra, Hamed Khanpour, Corby Rosset, and Ahmed Awadallah. Orca-math: Unlocking the potential of slms in grade school math. *CoRR*, abs/2402.14830, 2024.

[41] Pan Lu, Liang Qiu, Jiaqi Chen, Tanglin Xia, Yizhou Zhao, Wei Zhang, Zhou Yu, Xiaodan Liang, and Song-Chun Zhu. Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning. In Joaquin Vanschoren and Sai-Kit Yeung, editors, *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, 2021.

[42] Hugging Face. Open r1: A fully open reproduction of deepseek-r1, January 2025.

[43] Jia LI, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Costa Huang, Kashif Rasul, Longhui Yu, Albert Jiang, Ziju Shen, Zihan Qin, Bin Dong, Li Zhou, Yann Fleureau, Guillaume Lample, and Stanislas Polu. Numinamath. `[https://huggingface.co/AI-MO/NuminaMath-CoT](https://github.com/project-numina/aimo-progress-prize/blob/main/report/numina_dataset.pdf)`, 2024.

[44] Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang, Jianyu Jiang, Jiawei Wang, et al. Seed1. 5-vl technical report. *arXiv preprint arXiv:2505.07062*, 2025.

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

[46] ByteDance Seed, Jiaze Chen, Tiantian Fan, Xin Liu, Lingjun Liu, Zhiqi Lin, Mingxuan Wang, Chengyi Wang, Xiangpeng Wei, Wenyuan Xu, et al. Seed1. 5-thinking: Advancing superb reasoning models with reinforcement learning. *arXiv e-prints*, pages arXiv–2504, 2025.

[47] Pan Lu, Liang Qiu, Jiaqi Chen, Tony Xia, Yizhou Zhao, Wei Zhang, Zhou Yu, Xiaodan Liang, and Song-Chun Zhu. Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning, 2022.

[48] Jiaqi Chen, Jianheng Tang, Jinghui Qin, Xiaodan Liang, Lingbo Liu, Eric P. Xing, and Liang Lin. Geoqa: A geometric question answering benchmark towards multimodal numerical reasoning, 2022.

[49] Renrui Zhang, Xinyu Wei, Dongzhi Jiang, Yichi Zhang, Ziyu Guo, Chengzhuo Tong, Jiaming Liu, Aojun Zhou, Bin Wei, Shanghang Zhang, et al. Mavis: Mathematical visual instruction tuning. *arXiv e-prints*, pages arXiv–2407, 2024.

[50] Adam Dahlgren Lindström and Savitha Sam Abraham. Clevr-math: A dataset for compositional language, visual and mathematical reasoning, 2022.

[51] Shuaichen Chang, David Palzer, Jialin Li, Eric Fosler-Lussier, and Ningchuan Xiao. Mapqa: A dataset for question answering on choropleth maps, 2022.

[52] Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Nv-embed: Improved techniques for training llms as generalist embedding models. *arXiv preprint arXiv:2405.17428*, 2024.

[53] Ricardo JGB Campello, Davoud Moulavi, and Jörg Sander. Density-based clustering based on hierarchical density estimates. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 160–172. Springer, 2013.

[54] Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

[55] Xiaoye Qu, Yafu Li, Zhaochen Su, Weigao Sun, Jianhao Yan, Dongrui Liu, Ganqu Cui, Daizong Liu, Shuxian Liang, Junxian He, et al. A survey of efficient reasoning for large reasoning models: Language, multimodality, and beyond. *arXiv preprint arXiv:2503.21614*, 2025.

[56] Tingchen Fu, Jiawei Gu, Yafu Li, Xiaoye Qu, and Yu Cheng. Scaling reasoning, losing control: Evaluating instruction following in large reasoning models. *arXiv preprint arXiv:2505.14810*, 2025.

[57] Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang, Jianyu Jiang, Jiawei Wang, et al. Seed1. 5-vl technical report. *arXiv preprint arXiv:2505.07062*, 2025.

[58] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[59] Anthropic. Claude 3.7 sonnet. `https://www.anthropic.com`, 2024.

[60] 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.

[61] Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.

[62] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *CoRR*, abs/2408.03326, 2024.

[63] Hardy Chen, Haoqin Tu, Fali Wang, Hui Liu, Xianfeng Tang, Xinya Du, Yuyin Zhou, and Cihang Xie. Sft or rl? an early investigation into training r1-like reasoning large vision-language models. *arXiv preprint arXiv:2504.11468*, 2025.

[64] Sicong Leng, Jing Wang, Jiaxi Li, Hao Zhang, Zhiqiang Hu, Boqiang Zhang, Hang Zhang, Yuming Jiang, Xin Li, Deli Zhao, Fan Wang, Yu Rong, Aixin Sun, and Shijian Lu. Mmr1: Advancing the frontiers of multimodal reasoning. `https://github.com/LengSicong/MMR1`, 2025.

[65] Fanqing Meng, Lingxiao Du, Zongkai Liu, Zhixiang Zhou, Quanfeng Lu, Daocheng Fu, Botian Shi, Wenhai Wang, Junjun He, Kaipeng Zhang, Ping Luo, Yu Qiao, Qiaosheng Zhang, and Wenqi Shao. Mm-eureka: Exploring visual aha moment with rule-based large-scale reinforcement learning. *CoRR*, abs/2503.07365, 2025.

[66] Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand, 2024. Association for Computational Linguistics.

[67] Yaowei Zheng, Junting Lu, Shenzhi Wang, Zhangchi Feng, Dongdong Kuang, and Yuwen Xiong. Easyr1: An efficient, scalable, multi-modality rl training framework. `https://github.com/hiyouga/EasyR1`, 2025.

[68] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, pages 8748–8763. PMLR, 2021.

[69] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. MiniGPT-4: Enhancing vision-language understanding with advanced large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[70] 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.

[71] 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, 2024.

[72] OpenAI. Introduction to chatgpt-o3, 2025.

[73] Kimi Team. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025.

[74] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023.

[75] Jarvis Guo, Tuney Zheng, Yuelin Bai, Bo Li, Yubo Wang, King Zhu, Yizhi Li, Graham Neubig, Wenhu Chen, and Xiang Yue. Mammoth-vl: Eliciting multimodal reasoning with instruction tuning at scale, 2024.

[76] Zhaochen Su, Jun Zhang, Tong Zhu, Xiaoye Qu, Juntao Li, Min Zhang, and Yu Cheng. Timo: Towards better temporal reasoning for language models. *arXiv preprint arXiv:2406.14192*, 2024.

[77] Chancharik Mitra, Brandon Huang, Trevor Darrell, and Roei Herzig. Compositional chain-of-thought prompting for large multimodal models, 2024.

[78] Minghe Gao, Shuang Chen, Liang Pang, Yuan Yao, Jisheng Dang, Wenqiao Zhang, Juncheng Li, Siliang Tang, Yueting Zhuang, and Tat-Seng Chua. Fact :teaching mllms with faithful, concise and transferable rationales, 2024.

[79] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[80] Yafu Li, Xuyang Hu, Xiaoye Qu, Linjie Li, and Yu Cheng. Test-time preference optimization: On-the-fly alignment via iterative textual feedback. *arXiv preprint arXiv:2501.12895*, 2025.

[81] Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025.

[82] Xiangxiang Chu, Hailang Huang, Xiao Zhang, Fei Wei, and Yong Wang. Gpg: A simple and strong reinforcement learning baseline for model reasoning, 2025.

[83] Minghe Gao, Xuqi Liu, Zhongqi Yue, Yang Wu, Shuang Chen, Juncheng Li, Siliang Tang, Fei Wu, Tat-Seng Chua, and Yueting Zhuang. Benchmarking multimodal cot reward model stepwise by visual program, 2025.

[84] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi 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.

[85] Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*, 2023.

[86] Yi-Fan Zhang, Huanyu Zhang, Haochen Tian, Chaoyou Fu, Shuangqing Zhang, Junfei Wu, Feng Li, Kun Wang, Qingsong Wen, Zhang Zhang, et al. Mme-realworld: Could your multimodal llm challenge high-resolution real-world scenarios that are difficult for humans? *arXiv preprint arXiv:2408.13257*, 2024.

A Appendix

A.1 Training settings

The training process can be divided into three distinct phases: cold start, multimodal reinforcement learning, and text-only reinforcement learning. Key hyperparameters for each training phase are detailed in Table 5.

A.2 Algorithm in Prioritized Advantage Distillation (PAD)

The PAD mechanism, introduced conceptually in the main text, is detailed in Algorithm 1 to clarify its step-by-step operation in refining training batches for more effective learning.

Initially, PAD filters the original batch $\mathcal{B}$ to create an "effective set" $\mathcal{E}$ of sample indices and a corresponding map $\hat{A}_{\mathcal{E}}$ for their advantages (Lines 2-10 in Algorithm 1). For each sequence i in $\mathcal{B}$, its absolute advantage $|\tilde{A}_i|$ is computed. If this value falls within a specified informative range $[T_{low}, T_{high}]$, where $T_{low} > 0$ is crucial for excluding stagnant (near-zero advantage) samples, the index i is added to $\mathcal{E}$, and its absolute advantage $\hat{A}_{i,abs}$ is stored in $\hat{A}_{\mathcal{E}}$.

If this effective set $\mathcal{E}$ is non-empty, prioritized sub-sampling is performed (Lines 12-29). This multi-step process involves: (a) Calculating sampling probabilities P_j for each sequence index $j \in \mathcal{E}$

Table 5: Key Hyperparameters for Training Stages.

Component	Hyperparameter	Component	Hyperparameter
Cold Start	Learning Rate = 2.0×10^{-5} Gradient Accumulation = 8 Number of Epochs = 5 LR Scheduler = Cosine Warmup Ratio = 0.05 Max Sequence Length = 32768 Precision = BF16 DeepSpeed = Zero2	**Actor**	Global Batch Size = 128 Micro Batch Rollout = 4 Max Grad Norm = 1.0 Learning Rate (lr) = 1×10^{-6} Weight Decay = 1×10^{-2} Entropy Coef Init (β_0) = 0.02 Entropy Coef Min (β_{min}) = 0.0 Entropy Decay Rate (λ) = 0.985 (exp) Entropy Warmup Steps (τ_w) = 140 Total Updates = 200000
GRPO	Adv Estimator = grpo KL Penalty Type = low var kl KL Coef = 2×10^{-3} τ = 0.3	**Model Settings**	Max Prompt Length = 8192 Max Response Length = 8192 Rollout Batch Size = 512 Generation Temperature = 1.0 Generation Top P = 0.95

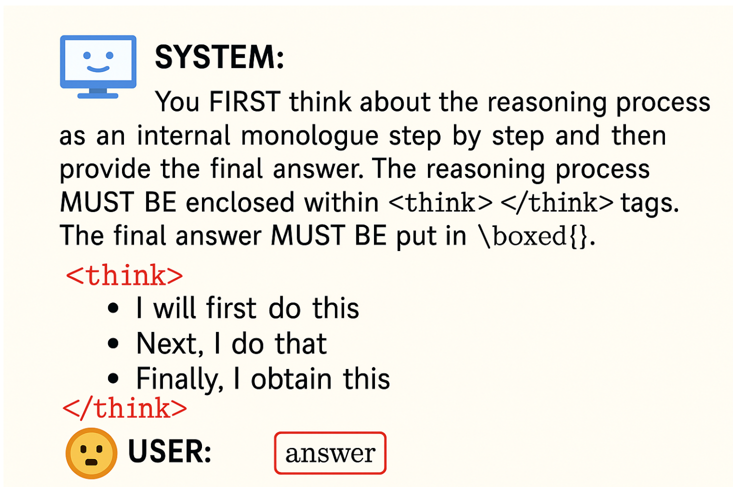


Figure 5: Prompt Template for ReVisual-R1 in both inference and training stages.

via a temperature-controlled Softmax distribution over their stored absolute advantages $\hat{A}_{\mathcal{E}}[j]$ (Lines 14-21). A uniform probability distribution across $\mathcal{E}$ serves as a fallback mechanism should the Softmax normalization term Z be zero. (b) Determining the sub-sample size k' as $\min(\lceil \rho N \rceil, |\mathcal{E}|)$, where ρ is the sub-sampling ratio and N is the original batch size (Line 23). (c) Sampling k' indices from $\mathcal{E}$ according to the calculated probabilities P_{dist} (Line 25). (d) Constructing the final distilled mini-batch $\mathcal{B}_{\text{distilled}}$ by retrieving the original sequences corresponding to these k' sampled indices (Lines 27-29). If $\mathcal{E}$ is void, an empty batch is returned. This entire procedure ensures that training batches are enriched by systematically filtering out uninformative data and prioritizing samples anticipated to yield more substantial learning signals.

A.3 Performance on Multimodal General Benchmarks

In this section, we provide an analysis in Table 6 to reveal that ReVisual-R1 demonstrates strong and competitive performance across these general MLLM benchmarks. Specifically, on the MMMU benchmark [84], Revisual-R1 secures the second-best score (50.55), closely following ThinkLite-7B-VL. It achieves leading performance on MM-Vet [85] with a top score of 49.81. In the MME-RealWorld benchmark [86], Revisual-R1 (62.68) delivers a solid performance, though it is surpassed by mmr1-mathv0 and ThinkLite-7B-VL. These results underscore Revisual-R1's robust and well-rounded reasoning capabilities on these general multimodal tasks, particularly its notable strength on

Algorithm 1 Prioritized Advantage Distillation (PAD)

Require: Original batch $\mathcal{B} = \{seq_1, \ldots, seq_N\}$ with $N = |\mathcal{B}|$; advantage estimates $A_{\text{est}} = \{\tilde{A}_1, \ldots, \tilde{A}_N\}$; thresholds $T_{\text{low}}, T_{\text{high}}$; temperature τ; sub–sampling ratio ρ

Ensure: Distilled mini-batch $\mathcal{B}_{\text{distilled}}$

$\triangleright$ **Steps 1 & 2: Per-Sequence Advantage Calculation and Effective Sample Filtering**

$\mathcal{E} \leftarrow \emptyset$ $\triangleright$ Set of indices for effective samples

$\hat{A}_{\mathcal{E}} \leftarrow \{\}$ $\triangleright$ Map: original index $i \in \mathcal{E} \rightarrow$ its absolute advantage $\hat{A}_i$

for $i \leftarrow 1$ **to** N **do**

  $\hat{A}_{i,abs} \leftarrow |\tilde{A}_i|$ $\triangleright$ Absolute advantage of current sequence i

  if $T_{\text{low}} \leq \hat{A}_{i,abs} \leq T_{\text{high}}$ **then**

    $\mathcal{E} \leftarrow \mathcal{E} \cup \{i\}$ $\triangleright$ Add index to effective set

    $\hat{A}_{\mathcal{E}}[i] \leftarrow \hat{A}_{i,abs}$ $\triangleright$ Store absolute advantage for effective sample i

  end if

end for

$\mathcal{B}_{\text{distilled}} \leftarrow \emptyset$

if $|\mathcal{E}| > 0$ **then**

  $\triangleright$ **Step 3: Prioritized Sub-sampling from the Effective Set**

  $\triangleright$ *a. Calculate sampling probabilities P_j for each $j \in \mathcal{E}$*

  $Z \leftarrow \sum_{j \in \mathcal{E}} \exp(\hat{A}_{\mathcal{E}}[j]/\tau)$ $\triangleright$ Normalization term (Softmax denominator over $\mathcal{E}$)

  $P_{\text{dist}} \leftarrow \{\}$ $\triangleright$ Map: original index $j \in \mathcal{E} \rightarrow$ its sampling probability P_j

  for all $j \in \mathcal{E}$ **do**

    if $Z > 0$ **then**

      $P_{\text{dist}}[j] \leftarrow \exp(\hat{A}_{\mathcal{E}}[j]/\tau)/Z$

    else

      $P_{\text{dist}}[j] \leftarrow 1/|\mathcal{E}|$ $\triangleright$ Uniform fallback if $Z = 0$

    end if

  end for

  $\triangleright$ *b. Determine actual sub-sample size k'*

  $k' \leftarrow \min(\lceil \rho N \rceil,\ |\mathcal{E}|)$

  $\triangleright$ *c. Sample k' indices from $\mathcal{E}$ according to probabilities P_{dist}*

  $S_{\text{sampled\_indices}} \leftarrow \textsc{Sample}(\mathcal{E},\ P_{\text{dist}},\ k')$ $\triangleright$ $S_{\text{sampled\_indices}}$ is a list of k' indices from $\mathcal{E}$

  $\triangleright$ *d. Form the distilled mini-batch*

  for all $idx \in S_{\text{sampled\_indices}}$ **do**

    $\mathcal{B}_{\text{distilled}} \leftarrow \mathcal{B}_{\text{distilled}} \cup \{seq_{idx}\}$

  end for

end if

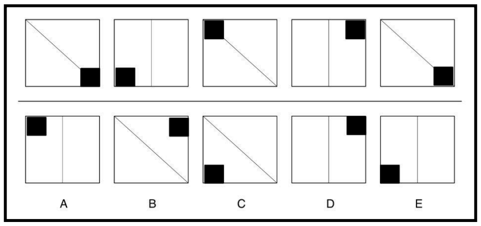
return $\mathcal{B}_{\text{distilled}}$

MM-Vet, within a competitive field where no single open-source 7B model consistently dominates all evaluated benchmarks.

Table 6: Performance of open-source multimodal reasoning 7B models on general MLLM benchmarks

Model	MMMU	MM-Vet	MME-RealWorld
VL-Rethinker	42.22	47.59	61.43
OpenVLThinker	19.37	40.73	13.46
MMR1-Math-v0	47.33	42.75	**69.12**
R1-VL-7B	48.23	45.64	59.31
ThinkLite-7B-VL	**51.22**	37.44	68.49
VLAA-Thinker-7B	50.44	41.46	64.44
Revisual-R1	50.55	**49.81**	66.68

A.4 Reasoning Example



Question:

Which of the boxes comes next in the sequence? Select answers from A-E

Ours Revisual-R1 7B Model:

<think>

Okay, let's try to figure out the sequence here. So there are five options labeled A to E, each with some black squares in different positions. The problem is asking which one comes next in the sequence.

First, I need to look at the given figures and see if there's a pattern. Let me list them out again to make sure I have them right, Top row:
1. A: A square divided by a diagonal, with a black square in the bottom right corner.
2. B: Two vertical stripes, each with a black square on the right side.
3. C: A square with a diagonal from top left to bottom right, and a black square in the top left quadrant.
4. D: Two vertical stripes, with a black square in the right stripe's upper part.
5. E: A square split into two vertical parts, with a black square in the left part.

[Reflection]Wait, maybe I should number them as per their order in the problem. The first row is options 1 to 5, then the second row is A to E. **Wait, the problem shows** the first row as the original five, then the next row labeled A to E.

[Revisit Image]Wait, looking back:The user presented the problem with two rows. The first row has five images (probably labeled 1 to 5), but the user then labels the second row as A to E. **Wait,** the actual question's options are A to E, each in the second row.

The sequence **seems to** involve the placement of the black squares relative to the diagonals or divisions of the square. **Let me analyze each option**:

Original top row (before the line):……

Then the next row is labeled A to E: A: Square split vertically, left third has a black square.……

[Self-Critic]Wait, perhaps the original problem's figures are arranged as follows:

First row (top): ……

Second row (labeled A to E): A: Vertical division, left part has a black square.……

[Logic]Now, trying to see the pattern. Let's look at the original figures (before the line) and then the options A-E. The task is to determine which comes next.Looking at the first row's figures (original positions):……

[Self-Critic]Wait, maybe the pattern alternates between diagonal and vertical splits. Let's check:

Original sequence (top row):1. Diagonal, lower right.……

Then the next row (A-E): A: Vertical split (three parts?), left part black.……

If the pattern is alternating between diagonal and vertical splits, then after figure 5 (diagonal), the next would be a vertical split.

[Hypothesis] Looking at the original vertical splits: figures 2, 4, and then **maybe** A, D, E.

Let's see: Original vertical splits (two parts) are figures 2 (lower right), 4 (upper right). Then after figure 5 (diagonal), next could be another vertical split. Among options A-E, which are vertical splits: A: vertical split into three columns? **Or two columns?** The image for A is a square split vertically into left and right, with a black square in the left part. D is vertical split into two, with black in the right upper. E is vertical split into two, black in left lower.

So after figure 5 (diagonal), the next might be a vertical split. The last vertical split in the original sequence (figure 4) has the black square in the upper right. Then the next vertical split could be in the lower left (E) or upper left (A?), **but need to check the order**.

[Hypothesis & Analysis] Alternatively, looking at the positions:\n\nOriginal vertical splits:\n\nFigure 2: right lower.\n\nFigure 4: right upper.\n\n**So if the pattern for vertical splits is alternating between lower and upper in the right section, then next could be lower right again?** But figure 2 is right lower, figure 4 is right upper. So maybe the next vertical split would be right lower again. Then option E is left lower. **Hmm, not sure**.

Alternatively, **maybe** the sequence is moving in a certain direction. Let's think of the figures as arranged in rows. The first row has five figures, and the second row has A-E. The correct answer is among A-E, which are the second row.

[Hypothesis] Alternatively, the sequence might be based on rotation or the position of the black square. For example, starting with diagonal, then vertical splits. Let's see:…… **Wait**, the options A-E are all vertical splits except B and C, which have diagonals. **Wait,** original figures 1-5 have a mix: 1,3,5 have diagonals; 2,4 have vertical splits. The options A-E: A is vertical split left; B is diagonal with black in upper right; C is diagonal upper left; D is vertical split right upper; E is vertical split left lower.

So after figure 5 (diagonal lower right), the next figure is likely a vertical split. Now, in the original vertical splits (figures 2 and 4), figure 2 is right lower, figure 4 is right upper. **So maybe the next vertical split alternates.** If figure 4 was right upper, then next could be right lower again (figure 2's position).

[Reflection] **But** option E is left lower. **Hmm. Alternatively, the vertical splits in the original sequence go from right lower to right upper.** If the next vertical split is right lower again, **but** that's figure 2. **But E is left lower.**

[Hypothesis] Alternatively, **maybe the direction is moving left**.

Wait, maybe the pattern is that after two diagonal figures (1,3,5), the vertical splits alternate between right and left. Let's see:\n\nOriginal vertical splits: figure 2 (right lower), figure 4 (right upper). Then the next vertical split might be left lower (E) or left upper (A). **But** figure A is left black square, which is a vertical split left. If the pattern is moving from right to left, then after figure 4 (right upper), next would be left lower (E).

[Self-Critic] But I need to check. Alternatively, looking at the positions of the black squares in the vertical splits:\n\nFigure 2: right third, lower part.\n\nFigure 4: right third, upper part.

So the vertical split's black square moves from lower to upper in the same right third. Then, if the next vertical split is in the left third, **maybe lower part (E)**. So after moving right upper, next is left lower. That would make the pattern: right lower, right upper, left lower. So E. Alternatively, the answer is E.

[Self-Critic] Let's check the options: Original sequence: diagonal, vertical split (right lower), diagonal, vertical split (right upper), diagonal. Then next is vertical split (left lower, E). That seems plausible.Another way: looking at the positions of the black squares in the vertical splits. The vertical splits in figures 2 and 4 have the black square in the right third. Figure 2's black square is at the lower part of the right third, figure 4's is at the upper part.

So if we follow that pattern, the next vertical split would be in the left third, perhaps at the lower part (E). So the answer is E.\n

</think>

<summary>

The sequence involves alternating between diagonal and vertical split figures. After the diagonal figure (5), the next is a vertical split. The vertical splits in the original sequence (figures 2 and 4) have black squares in the right third, moving from lower to upper. The next vertical split follows this pattern but in the left third, at the lower part.

Answer: \boxed{E} ✓

</summary>

Figure 6: Our Revisual-R1 model reasoning case, showcasing its exceptional reasoning ability. The model generates long responses, continuously hypothesizing, reflecting, verifying, and correcting to arrive at the final answer, while also providing a summary answer.