# VL-Rethinker: Incentivizing Self-Reflection of Vision-Language Models with Reinforcement Learning

**Haozhe Wang**$^{\diamond\heartsuit\ddagger}$, **Chao Qu**$^{\ddagger}$, **Zuming huang**$^{\ddagger}$, **Wei Chu**$^{\ddagger}$, **Fangzhen Lin**$^{\diamond}$, **Wenhu Chen**$^{\heartsuit\dagger}$

HKUST$^{\diamond}$, University of Waterloo$^{\heartsuit}$, INF.AI$^{\ddagger}$, Vector Institute$^{\dagger}$

Corresponding to: jasper.whz@outlook.com, wenhuchen@uwaterloo.ca

Project Page: `https://tiger-ai-lab.github.io/VL-Rethinker/`
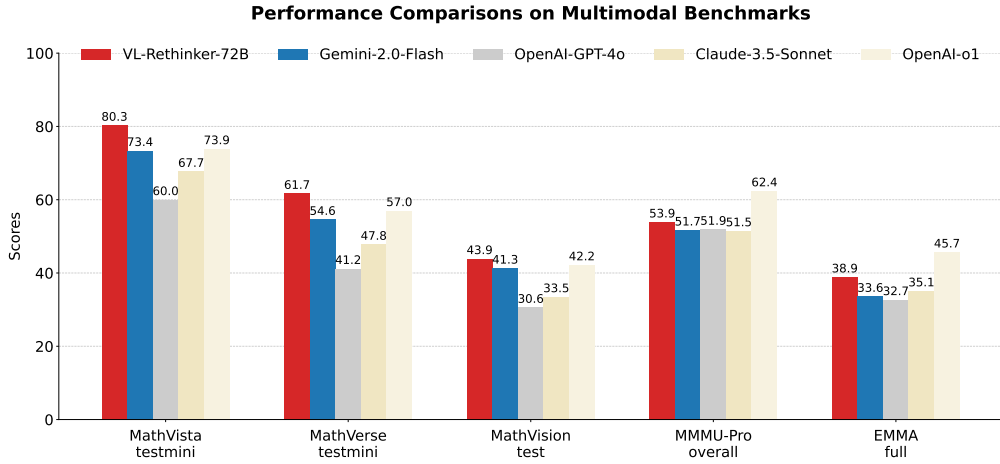
Figure 1: Performance comparison between VL-RETHINKER and other SoTA models on different multimodal reasoning benchmarks.

## Abstract

Recently, slow-thinking systems like GPT-o1 and DeepSeek-R1 have demonstrated great potential in solving challenging problems through explicit reflection. They significantly outperform the best fast-thinking models, such as GPT-4o, on various math and science benchmarks. However, their multimodal reasoning capabilities remain on par with fast-thinking models. For instance, GPT-o1's performance on benchmarks like MathVista, MathVerse, and MathVision is similar to fast-thinking models. In this paper, we aim to enhance the slow-thinking capabilities of vision-language models using reinforcement learning (without relying on distillation) to advance the state of the art. First, we adapt the GRPO algorithm with a novel technique called Selective Sample Replay (SSR) to address the vanishing advantages problem. While this approach yields strong performance, the resulting RL-trained models exhibit limited self-reflection or self-verification. To further encourage slow-thinking, we introduce Forced Rethinking, which appends a textual rethinking trigger to the end of initial rollouts in RL training, explicitly enforcing a self-reflection reasoning step. By combining these two techniques, our model, VL-RETHINKER, advances state-of-the-art scores on MathVista, MathVerse, and MathVision to achieve significantly to achieve 80.3%, 61.8% and 43.9% respectively. VL-RETHINKER also achieves open-source SoTA on multi-disciplinary benchmarks such as MMMU-Pro, EMMA, and MEGA-Bench, narrowing the gap with GPT-o1.

# 1  Introduction

Recently, slow-thinking systems such as OpenAI-o1 [Jaech et al., 2024], DeepSeek-R1 [Guo et al., 2025], Kimi-1.5 [Team et al., 2025], Gemini-Thinking [Team et al., 2023], and QwQ/QvQ [Bai et al., 2025] have significantly advanced the performance of language models in solving challenging math and science problems. These models engage in extended reasoning and reflection before arriving at a final answer, in contrast to fast-thinking models like GPT-4o [Hurst et al., 2024] and Claude-3.5-Sonnet [Anthropic, 2024], which produce answers rapidly without such deliberation. Through this reflective process, slow-thinking models outperform the best fast-thinking models by over 30% on math datasets such as AIME24 and AMC23 [Hendrycks et al.], and by around 10% on general science benchmarks like GPQA [Rein et al., 2024].

However, their multimodal reasoning capabilities remain on par with fast-thinking models. For example, GPT-o1 achieves 73.9% on MathVista [Lu et al., 2023] and 57.0% on MathVerse [Wang et al., 2024a], which is slightly worse than Qwen2.5-VL-72B [Wang et al., 2024b] scoring 74.8% and 57.2% on the same benchmarks. This raises an important research question:

*How can we effectively incentivize multimodal slow-thinking capabilities in Vision-Language Models?*

To address this, we explore how to directly train multimodal reasoning models through reinforcement learning (RL), without relying on distillation from stronger teacher models [Yang et al., 2025, Deng et al., 2025]. Our main contributions are as follows:

**GRPO with SSR:** We construct a diverse dataset of 16,000 queries covering perception, complex reasoning, and multiple disciplines. To train our vision-language model (VLM), we adapt the Group Relative Policy Optimization (GRPO) algorithm [Guo et al., 2025], which computes advantages by comparing responses within the same query group and normalizes rewards to guide policy updates. However, we identify a key limitation: the vanishing advantages problem. This occurs when all responses in a group receive identical rewards (either all correct or all incorrect), leading to zero advantage signals and ineffective gradient updates. As training progresses, this reward uniformity becomes more common, causing instability and constraining the model to shallow reasoning patterns.

To mitigate this, we introduce Selective Sample Replay (SSR), which augments the on-policy training batch with high-quality examples from a replay buffer. SSR counters the emergence of uniform reward patterns, maintains more informative learning signals, and promotes deeper reasoning exploration. Despite its simplicity, our direct RL approach achieves strong empirical performance on multiple multimodal reasoning benchmarks. However, we find that the resulting models still lack explicit reflective behavior, limiting their potential.

**Forced Rethinking:** To address this, we propose a simple yet effective technique called forced rethinking. We append a textual rethinking trigger to the end of roll-out responses and train the model using the same RL setup. This strategy prompts the model to engage in self-reflection and self-verification before producing the final answer. We name the resulting model VL-RETHINKER. As shown in Fig. 1, VL-RETHINKER significantly outperforms GPT-o1 on mathematical benchmarks such as MathVista, MathVerse, and MathVision. Furthermore, on general-purpose multimodal benchmarks like EMMA and MMMU-Pro, VL-RETHINKER achieves a new open-source state of the art performance, closely approaching GPT-o1's performance.

**Observations:** We observe a notable discrepancy between modalities: while RL training often induces slow-thinking behaviors such as longer reasoning traces in math-focused tasks [Zeng et al., 2025, Wen et al., 2025], vision-language tasks rarely exhibit such development. Specifically, models trained on multimodal data do not naturally adopt longer chains of thought or spontaneous wait patterns. Understanding why RL incentivizes reflection differently in multimodal contexts versus math-only settings is an important avenue for future work.

**In summary, our contributions are threefold:** (1) We propose and validate a simple, direct RL approach for enhancing VLM reasoning, offering a viable alternative to complex supervised fine-tuning and distillation pipelines. (2) We introduce Selective Sample Replay (SSR) to improve the training stability and effectiveness of GRPO-based RL for VLMs. (3) We propose Forced Rethinking, a lightweight yet powerful strategy to incentivize self-reflection in VLMs.

Our final model, VL-RETHINKER, sets a new state of the art on key multimodal reasoning benchmarks, demonstrating the value of slow-thinking reinforcement in vision-language modeling.

## 2 Preliminaries

This section outlines the key concepts and training setup for multimodal reasoning. We first formulate the multimodal reasoning problem and define our learning objective. Then, we describe the standard Reinforcement Learning (RL) algorithm used in our framework.

### 2.1 Problem Formulation

We define the multimodal reasoning task as follows: given a multimodal input consisting of one or more images $I$ and a textual query $Q$, the goal is to generate a textual response $y$ that correctly answers the query by reasoning over both visual and textual information.

Let $\mathcal{V}$ denote the visual input space and $\mathcal{T}$ the textual input space. The input is denoted as $x \in \mathcal{V} \times \mathcal{T}$, where $x = (I, Q)$ captures both modalities. The output is a textual response $y \in \mathcal{Y}$, where $\mathcal{Y}$ represents the response space. The challenge lies in building a vision-language model (VLM) that can integrate multimodal information and perform deep, multi-step reasoning—especially for complex queries requiring extended deliberation or external knowledge.

Our goal is to improve the reasoning capabilities of an instruction-tuned VLM that initially exhibits *fast-thinking* behavior, i.e., producing shallow, immediate responses. We aim to shift the model toward *slow-thinking* behavior—engaging in deeper, more deliberate reasoning—to significantly improve performance on downstream multimodal tasks. We achieve this via direct reinforcement learning (RL), which encourages the generation of accurate, thorough, and well-reasoned responses by assigning higher rewards to such outputs.

Formally, we train a policy $\pi_\theta(y|x)$, parameterized by $\theta$, to maximize the expected reward $r(y, x)$ for generating a response $y$ given an input $x$. The reward function $r(y, x)$ is designed to prioritize correctness, often represented as a binary label Guo et al. [2025]. The learning objective is:

$$\max_\theta \mathbb{E}_{x \sim \mathcal{D}} \mathbb{E}_{y \sim \pi_\theta(\cdot|x)}[r(y, x)]$$

where $\mathcal{D}$ is a dataset of multimodal queries and their corresponding answers, drawn from existing sources. It contains 16K examples covering perception, complex reasoning, and multiple subject areas, sourced from datasets such as Virgo [Du et al., 2025], R1-OneVision [Yang et al., 2025], AI2D [Kembhavi et al., 2016], ScienceQA [Saikh et al., 2022], and MMEureka [Meng et al., 2025].

### 2.2 Group Relative Policy Optimization (GRPO)

Group Relative Policy Optimization (GRPO) is a policy optimization algorithm that computes token-level advantages by comparing responses within a query-specific group. For a given input $x = (I, Q)$, the behavior policy $\pi_{\theta_{\text{old}}}$ generates a group of $G$ candidate responses $\{y_i\}_{i=1}^G$. The advantage for the $i$-th response at time step $t$ is computed by normalizing the rewards across the group:

$$\hat{A}_{i,t} = \frac{r(x, y_i) - \text{mean}(\{r(x, y_1), \ldots, r(x, y_G)\})}{\text{std}(\{r(x, y_1), \ldots, r(x, y_G)\})}$$

The GRPO objective incorporates a clipped surrogate loss with a KL-penalty to stabilize updates:

$$\frac{1}{G} \sum_{i=1}^G \frac{1}{|y_i|} \sum_{t=1}^{|y_i|} \min\left[ \frac{\pi_\theta(y_{i,t}|x, y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t}|x, y_{i,<t})} \hat{A}_{i,t}, \text{clip}\left( \frac{\pi_\theta(y_{i,t}|x, y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t}|x, y_{i,<t})}, 1 - \epsilon, 1 + \epsilon \right) \hat{A}_{i,t} \right]$$

Here, $\epsilon$ is a hyperparameter controlling the tolerance for policy deviation. The `clip` function prevents large updates by ensuring that the ratio between the current and reference policy stays within a predefined range. This stabilizes training and helps avoid performance collapse due to overly aggressive updates. (Indeed we don't have std)

# 3 Our Method

Thid section outlines our contribution, including Selective Sample Replay (SSR) and Forced rethinking, two techniques to incentivize slow-thinking capabilities.

## 3.1 Vanishing Advantages in GRPO

We identify a critical limitation in GRPO, which we term the "Vanishing Advantages" problem. In GRPO, a simple binary reward signal is used to indicate the correctness of a response $y$ to a given vision-language query $x$. When all responses within a query group are uniformly correct or uniformly incorrect, the calculated advantages become zero for every response in that group. Consequently, such examples cease to provide effective policy gradients, as the gradient signal relies on non-zero advantages to guide learning.
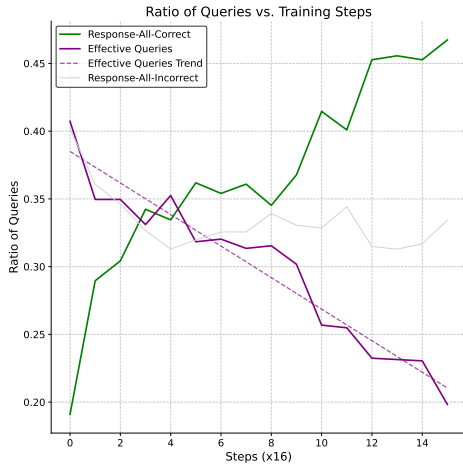


Figure 2: Illustration of the Vanishing Advantages problem. Training of 72B rapidly saturates, leading to a significant decrease of effective queries to only 20% within 256 steps.
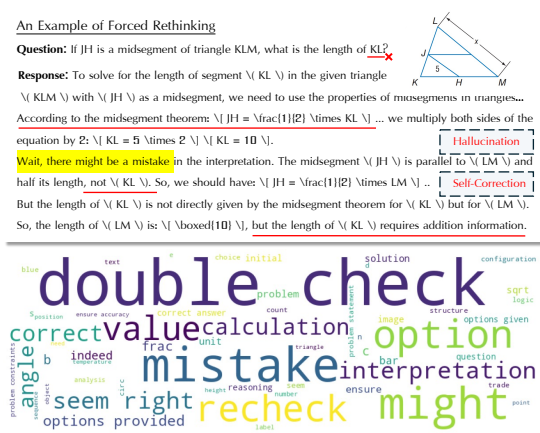


Figure 3: An example of Forced Rethinking (Top). VL-Rethinker discovers a flawed problem via rethinking upon its hallucinations. The word cloud of VL-Rethinker (Bottom) shows the learned rethinking pattern of self-verification, self-correction and self-questioning.

This issue becomes increasingly pronounced as training progresses. As illustrated in Fig. 2, the percentage of examples exhibiting non-zero advantages steadily declines from approximately 40% at the beginning of training to below 20% after $16 \times 16$ gradient steps. This also reflects the policy's tendency to converge towards generating responses with uniform rewards within group – either consistently correct or consistently incorrect – over time. As the policy updates, the diversity in rewards within the group diminishes, thereby exacerbating the problem. Similar trends are also observed in text LLMs [Yu et al., 2025], concurrent to our work.

The "vanishing advantages" phenomenon poses a significant impediment to our goal of transitioning the VLM from "fast-thinking" to "slow-thinking". As more query groups yield zero advantages, the effective batch size for training decreases, reducing the number of examples that contribute meaningfully to the policy update. This can lead to training instability, causing the policy to stop improving prematurely and potentially become trapped in suboptimal local optima. Such optima often correspond to shallow reasoning traces, discouraging the model from exploring complex, multi-step reasoning pathways.

## 3.2 Selective Sample Replay (SSR)

To counteract the Vanishing Advantages problem and maintain learning efficiency, we introduce Selective Sample Replay (SSR). SSR enhances GRPO by integrating an experience replay mechanism that selectively displays informative samples from past iterations.

SSR maintains a replay buffer $\mathcal{B}_{\text{replay}}$ storing tuples $(x, y_i, \hat{A}_i)$ that will persist for $K$ training steps, specifically retaining only those samples where the query group exhibited non-zero advantages
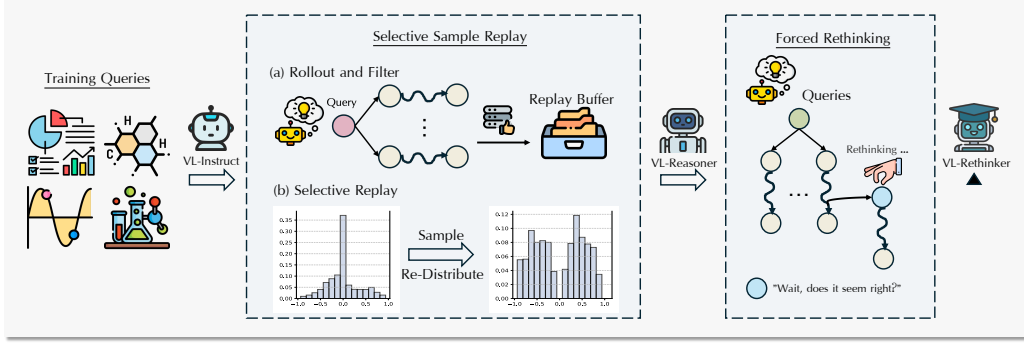
Figure 4: **Method Overview.** We present a two-stage RL method based on Qwen-VL-2.5-Instruct. The first stage enhances general reasoning through GRPO with Selective Sample Replay (SSR), which retains explored trajectories with non-zero advantages and selectively replay samples based on their advantages. The second stage promotes deliberate reasoning using forced rethinking, where we append a specific rethinking trigger to RL rollouts.

($|\hat{A}_k| > 0$). During each training step, we draw samples from the replay buffer $\mathcal{B}_{\text{replay}}$, from which the sampling is prioritized based on the absolute advantages, promoting the replay of experiences that previously indicated strong positive or negative advantages. Specifically, a sample $j$ from the buffer is selected with probability:

$$P(\text{select } j) = \frac{|\hat{A}_j|^\alpha}{\sum_{k \in \mathcal{B}_{\text{replay}}} |\hat{A}_k|^\alpha}$$

where $\alpha$ is a hyperparameter controlling the prioritization intensity ($\alpha = 1$ for direct proportionality).

By selectively sampling informative experiences, SSR not only counteracts the reward uniformity but also ensures a richer diversity of advantages within each training batch, thus providing more consistent gradient signals. This stabilizes training and prevents premature stagnation, as we further detail in the ablation studies (Fig. 5).

### 3.3 Forced Rethinking

While GRPO with SSR improves optimization stability, we observe that *complex, deliberative "slow-thinking" patterns, such as explicit self-correction, did not consistently emerge as a direct result of standard RL on VLMs*, a divergence from trends observed in large text-only models. To explicitly cultivate these deeper reasoning processes within our VLM framework, we introduce a training technique termed *Forced Rethinking*. This method aims to proactively encourage the model to engage in more extensive internal deliberation before arriving at a final prediction.

As depicted in Fig. 4, the central principle of Forced Rethinking involves a targeted intervention within the RL rollout procedure. Following the VLM's initial generation of a response $y_1$ to a given input $x$, we append a specific textual "rethinking trigger" to $y_1$. This augmented sequence is then fed back into the model, prompting it to generate a subsequent response segment $y_2$. Consequently, the complete generated sequence becomes $y = y_1 \oplus \text{trigger} \oplus y_2$.

To elicit a diverse range of reasoning behaviors, we designed three distinct categories of triggers: self-verification, self-correction, and self-questioning. To minimize disruption to the learned policy, we apply the strategy to a sampled fraction $q < 1$ of the generated responses. Notably, we observed that relying solely on the standard RL loss was insufficient for the model to consistently adopt the desired rethinking behavior. Therefore, we augment the training objective with an additional negative log-likelihood loss specifically applied to the $y_2$ segments that lead to a correct final answer. This auxiliary loss directly incentivizes the model to generate the desired rethinking patterns.

This approach offers a key distinction from methods [Deng et al., 2025, Yang et al., 2025] that rely on Supervised Fine-tuning (SFT) distillation from existing slow-thinking systems. Our VL-RETHINKER, trained with this strategy, does not necessitate a rethinking step for every query. Instead, it learns to strategically engage in this process only when it implicitly determines it to be necessary, potentially

leading to more efficient inference. Intriguingly, as illustrated in the example provided in Fig. 3, our VL-Rethinker demonstrates the capability to even identify flaws in the given problem when checking its initial reasoning through rethinking, showcasing a form of emergent metacognitive ability (similar to the findings in Wang et al. [2025]).

## 4    Experiments

Our experiments investigate the following key questions:

*Q1: Method Effectiveness.* How does our approach enhance performance on comprehensive multi-modal benchmarks compared to existing MLLMs?

*Q2: Ablation Studies.* How do the proposed Selective Sample Replay (SSR), Forced Rethinking, and curated data affect performance?

*Q3: Effectiveness of the learned rethinking behaviors.* Do the model learn to effectively and spontaneously perform deliberate thinking?

**Datasets and Benchmarks.** We source initial seed queries from established datasets, including Virgo [Du et al., 2025], R1-OneVision [Yang et al., 2025], and MM-Eureka [Meng et al., 2025]. To ensure data integrity, we implement a rigorous process of cleaning and quality control, retaining solely those queries that can be objectively verified. Our analysis indicates that RL training rapidly reaches saturation with the initial queries, as the majority are relatively easy to learn. Consequently, we refine the dataset further by applying a filtering mechanism based on the mean accuracy derived from eight random samplings [Wang et al., 2025]. Following these cleaning and filtering procedures, we derive a refined query set comprising approximately 16K queries which are readily evaluated for correctness using rule-based functions.

For evaluation, we employ a diverse set of challenging multimodal benchmarks:

- Math-related reasoning: MathVista [Lu et al., 2023], MathVerse [Zhang et al., 2024], and MathVision [Wang et al., 2024a].
- Multi-discipline understanding and reasoning: MMMU [Yue et al., 2024a], MMMU-Pro [Yue et al., 2024b], and EMMA [Hao et al., 2025].
- Large-scale long-tailed real-world tasks: MegaBench [Chen et al., 2024a].

This benchmark suite covers a wide range of complex multimodal reasoning challenges. We report the Pass@1 accuracy using greedy decoding.

**Baselines and Implementation.** We compare against several categories of models:

- Proprietary models: GPT-4o [Hurst et al., 2024], o1 [Jaech et al., 2024], Claude 3.5 Sonnet [Anthropic, 2024], Gemini-2.0-Flash [Team et al., 2023].
- State-of-the-art open-source models: Qwen-2.5-VL-72B [Bai et al., 2025], QvQ-72B [Wang et al., 2024b], InternVL-2.5-78B [Chen et al., 2024b], Llava-Onevision [Li et al., 2024], Llama-4-Scout and Kimi-VL [Team et al., 2025].
- Representative open-source reasoning-focused models: OpenVLThinker [Deng et al., 2025], R1-OneVision [Yang et al., 2025], R1-VL [Zhang et al., 2025] and MM-Eureka [Meng et al., 2025]. These models are mainly trained on multimodal reasoning dataset.

Our algorithm was implemented using the OpenRLHF framework. Training was conducted on the curated 16K query set for a maximum of 3 epochs. The final checkpoint was selected based on the mean reward achieved on a held-out validation set. We employed a near on-policy RL paradigm, where the behavior policy was synchronized with the improvement policy after every 1024 queries, which we define as an episode. The replay buffer for SSR persisted for the duration of each episode before being cleared. For each query, we sampled 8 responses. The training batch size was set to 512 query-response pairs. The code, models, and data will be made available via the project page.

### 4.1    Main Results

Our approach demonstrates significant performance gains, as evidenced by the quantitative results. For the 72B models (Table 1), VL-Rethinker-72B achieved 3.5% improvement on average compared to the base model, Qwen-2.5-VL-72B. Notably, VL-Rethinker-72B achieved state-of-the-art results

| Model | Math-Related | | | Multi-Discipline | | | Real-World |
|---|---|---|---|---|---|---|---|
| | MathVista testmini | MathVerse testmini | MathVision test | MMMU-Pro overall | MMMU val | EMMA full | MEGA core |
| Proprietary Model | | | | | | | |
| OpenAI-o1 | 73.9 | 57.0 | 42.2 | 62.4 | 78.2 | 45.7 | 56.2 |
| OpenAI-GPT-4o | 60.0 | 41.2 | 30.6 | 51.9 | 69.1 | 32.7 | 52.7 |
| Claude-3.5-Sonnet | 67.7 | 47.8 | 33.5 | 51.5 | 68.3 | 35.1 | 52.3 |
| Gemini-2.0-Flash | 73.4 | 54.6 | 41.3 | 51.7 | 70.7 | 33.6 | 54.1 |
| Open-Source Models | | | | | | | |
| Llama4-Scout-109B | 70.7 | - | - | 52.2 | 69.4 | 24.6 | 31.8 |
| InternVL-2.5-78B | 72.3 | 51.7 | 34.9 | 48.6 | 61.8 | 27.1 | 44.1 |
| QvQ-72B | 71.4 | 48.6 | 35.9 | 51.5 | 70.3 | 32.0 | 8.8 |
| LLava-OV-72B | 67.5 | 39.1 | 30.1 | 31.0 | 56.8 | 23.8 | 29.7 |
| Qwen-2.5-VL-32B | 74.7 | 48.5 | 38.4 | 49.5 | 70.0 | 31.1 | 13.3 |
| Qwen-2.5-VL-72B | 74.8 | 57.2 | 38.1 | 51.6 | 70.2 | 34.1 | 49.0 |
| VL-RETHINKER-72B | **80.3** | **61.7** | **43.9** | **53.9** | 68.8 | **38.9** | **51.3** |
| Δ (Ours - Open SoTA) | +5.5 | +4.5 | +5.5 | +1.7 | -1.4 | +5.8 | +2.3 |

Table 1: Comparison between our 72B model and other state-of-the-art models.

| Model | Math-Related | | | Multi-Discipline | | | Real-World |
|---|---|---|---|---|---|---|---|
| | MathVista testmini | MathVerse testmini | MathVision test | MMMU-Pro overall | MMMU val | EMMA full | MEGA core |
| General Vision-Language Models | | | | | | | |
| InternVL2-8B | 58.3 | - | 17.4 | 29.0 | 51.2 | 19.8 | 26.0 |
| InternVL2.5-8B | 64.4 | 39.5 | 19.7 | 34.3 | 56.0 | - | 30.4 |
| QwenVL2-7B | 58.2 | - | 16.3 | 30.5 | 54.1 | 20.2 | 34.8 |
| QwenVL2.5-7B | 68.2 | 46.3 | 25.1 | 36.9 | †54.3 | 21.5 | 35.0 |
| Llava-OV-7B | 63.2 | 26.2 | - | 24.1 | 48.8 | 18.3 | 22.9 |
| Kimi-VL-16B | 68.7 | 44.9 | 21.4 | - | †55.7 | - | - |
| Vision-Language Reasoning Models | | | | | | | |
| MM-Eureka-8B | 67.1 | 40.4 | 22.2 | 27.8 | 49.2 | - | - |
| R1-VL-7B | 63.5 | 40.0 | 24.7 | 7.8 | 44.5 | 8.3 | 29.9 |
| R1-Onevision-7B | 64.1 | 46.4 | 29.9 | 21.6 | - | 20.8 | 27.1 |
| OpenVLThinker-7B | 70.2 | 47.9 | 25.3 | 37.3 | 52.5 | 26.6 | 12.0 |
| VL-RETHINKER-7B | 74.9 | 54.2 | 32.3 | 41.7 | 56.7 | 29.7 | 37.2 |
| Δ (Ours - Prev SoTA) | +4.7 | +6.3 | +2.4 | +4.4 | +0.7 | +3.1 | +2.2 |

Table 2: Comparison between our 7B model and other general and reasoning vision-language models. † means that the results are reproduced by us.

on math-related benchmarks among all models, including OpenAI-o1. For the 7B models (Table 2), VL-RETHINKER-7B outperforms competitor 7B models that also employ RL, e.g., OpenVLThinker, R1-OneVision, by a large margin. These results underscore the effectiveness of our proposed approach in enhancing performance across various challenging benchmarks.

## 4.2 Ablation Study

**Ablation on Data.** Our training queries are comprised of three major genres: math-related vision-language queries, science-related queries and text-only ones. we conducted ablation studies on these components. As shown in Table. 3, removing text-only queries does not cause significant differences. As we further remove queries from the broader scientific domains, we observe a more pronounced drop in performance. This significant reduction underscores the importance of scientific data in improving the model's general reasoning ability.

**Ablation on Selective Sample Replay (SSR).** To address vanishing advantages, we introduce Selective Sample Replay (SSR) based on GRPO. GRPO-SSR filters out queries causing zero advantages and perform selective sampling with a probability proportional to the absolute advantage. To investigate the impact of filtering and selective replay, we establish two corresponding baselines for comparison against our full GRPO-SSR method (as used in VL-RETHINKER): GRPO-Filter and

| Model | RL-Algo | Data | MathVision | MathVista | MathVerse | MMMU-Pro | EMMA |
|---|---|---|---|---|---|---|---|
| VL-Rethinker-7B | SSR | 16K | 32.3 | 74.9 | 54.2 | 41.7 | 29.7 |
| w/o 'Forced-Rethinking' | SSR | 16K | 29.8 | 72.4 | 53.2 | 40.9 | 29.5 |
| - no SSR | Filter | 16K | 28.5 | 72.0 | 50.0 | 40.0 | 26.9 |
| - no SSR& Filter | GRPO | 16K | 26.0 | 70.9 | 51.4 | 38.8 | 26.2 |
| - no Text | SSR | 13K | 29.1 | 73.5 | 53.5 | 41.1 | 28.7 |
| - no Science&Text | SSR | 11K | 28.0 | 71.6 | 50.3 | 39.7 | 28.0 |

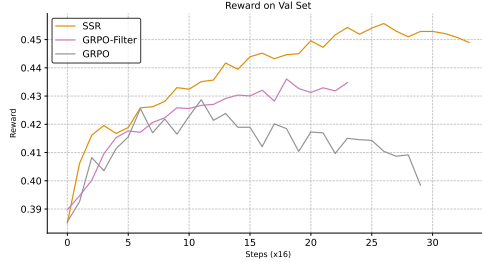Table 3: Ablation Results to show the impact of SSR and Data Mix.



Figure 5: Comparisons of training dynamics of GRPO, GRPO-Filter and GRPO-SSR. GRPO baseline exhibits significant overfit, and GRPO-Filter are more stabilized. GRPO-SSR achieves the best convergence.
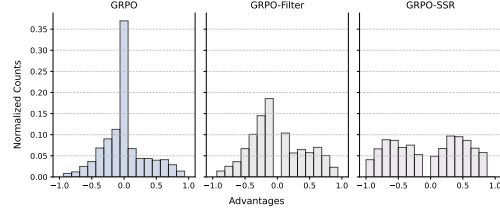


Figure 6: Comparisons of training batch advantage distribution. Standard GRPO and GRPO-Filter has biased advantage distribution, with mass centered around zero. In contrast, GRPO-SSR re-distribute the probability mass over training examples evenly across different advantage values.

GRPO. GRPO-Filter removes the SSR component from GRPO-SSR (similar to the dynamic filtering in DAPO [Yu et al., 2025], but don't involve an online re-sampling), while GRPO further removes the filtering of examples with zero advantages.

The results presented in Table. 3 highlight the effectiveness of our proposed components. The models trained with the full GRPO-SSR algorithm consistently achieves superior performance compared to the ablated versions, strongly supporting the benefits of both filtering and selective replay.

Further insights into the behavior of these algorithms are revealed by analyzing the training dynamics, as shown in Fig. 5. the GRPO baseline exhibits the most pronounced overfitting, eventually leading to performance degradation. This can be attributed to the vanishing advantages problem, where the number of training examples with near-zero advantages increases as training progresses. These examples provide minimal learning signal, effectively reducing the batch size and destabilizing the training process. In contrast, GRPO-SSR demonstrates a more stable training process and achieves better convergence compared to GRPO-Filter, suggesting the beneficial role of SSR.

The underlying reason for these differences is illuminated by the advantage distributions during training (Fig. 6). Standard GRPO displays a highly skewed distribution, with a pronounced peak at zero advantage, confirming that a large fraction of samples provides ineffective gradients. GRPO-Filter alleviates the extreme peak at zero, yet it still retains a strong central bias, indicating that many examples with very small advantages persist.

Conversely, GRPO-SSR significantly alters the advantage distribution by redistributing the probability mass away from zero and placing greater emphasis on examples with large absolute advantages. These examples, such as a correct response to a challenging query or an incorrect response to a simple one, are intuitively more informative as they likely lie closer to the decision boundary. By selectively replaying these high-advantage examples, GRPO-SSR ensures a more balanced and effective learning process, ultimately leading to improved convergence as evidenced by the reward curves.

**Analysis on Forced Rethinking.** To evaluate the effectiveness of our Forced Rethinking training technique in fostering deliberate reasoning, we compared its impact against baseline models and theoretical limits, as illustrated in Fig. 7. Our primary objective was to examine whether training with Forced Rethinking encourages VL-Rethinker to develop internal metacognitive awareness, enabling it to strategically decide when rethinking is beneficial, rather than applying it rigidly.
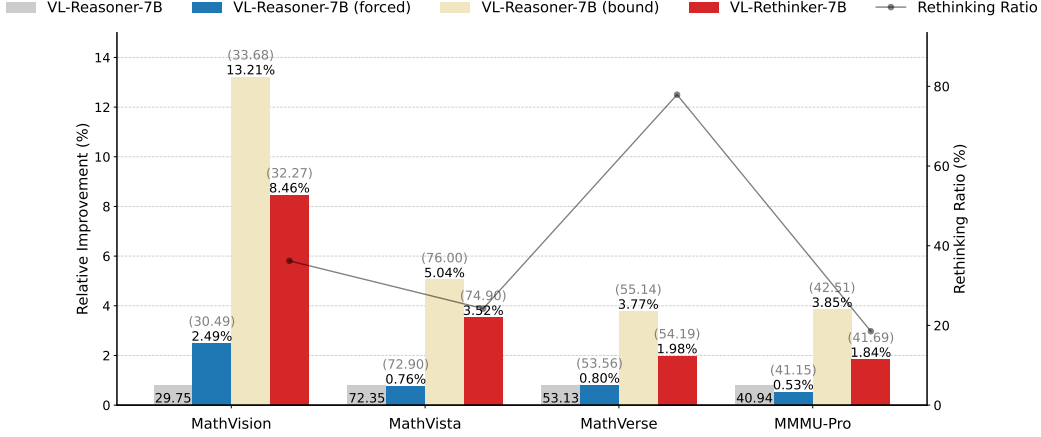
Figure 7: **Relative Improvement with Different Re-thinking Strategies.** We compare: (a) VL-Reasoner (forced), which is forced to rethink at test time; (b) VL-Reasoner (bound), represents the upper bound of test-time forced re-thinking; and (c) VL-RETHINKER is trained for self-reflection. The results indicate that forcing VL-Reasoner to rethink at test time yields positive performance gains. Training for self-reflection significantly enhances performance, achieving closer results to the upper bound of forced re-thinking. The overlaid line plot shows the rethinking ratio (right y-axis) of VL-RETHINKER across different benchmarks, showing VL-RETHINKER adaptively performs re-thinking, unlike the fixed forced re-thinking strategy.

Fig. 7 compares the performance of VL-RETHINKER against several configurations. The baseline is "w/o Forced Rethinking", which we dub *VL-Reasoner*. We first assessed the inherent potential of rethinking via *VL-Reasoner (forced)*, where the baseline model is compelled to perform a rethinking step at test time for every instance. The results (blue bars) show positive relative improvements across all benchmarks. This indicates that the baseline model already possesses latent rethinking capabilities that can lead to correct answers. However, this approach is suboptimal, as the baseline struggles to effectively leverage this ability, sometimes even corrupting initially correct answers through flawed rethinking. We also compute an upper bound, *VL-Reasoner (bound)* (yellow bars), which represents the maximum achievable improvement if test-time rethinking is only applied to the wrong outputs.

Crucially, VL-RETHINKER (red bars), trained using our Forced Rethinking technique, consistently outperforms the *VL-Reasoner (forced)* baseline. For example, on MathVision, *VL-Rethinker* achieves an 8.46% relative improvement, significantly higher than the 2.49% gained by passively forcing the baseline to re-think. This demonstrates that integrating rethinking into the training phase markedly enhances the model's capacity for effective self-reflection.

Importantly, the analysis highlights the adaptive nature of the learned rethinking behavior. The overlaid line plot (right y-axis) shows the "Rethinking Ratio" for VL-RETHINKER – the fraction of test instances where it spontaneously engaged in the rethinking process. This ratio varies substantially across benchmarks, in stark contrast to the rigid, 100% application in the *VL-Reasoner (forced)* scenario. It suggests that VL-RETHINKER has learned to selectively trigger re-thinking based on the query's perceived difficulty or its initial confidence, embodying the targeted metacognitive awareness rather than relying on a fixed, potentially inefficient strategy.

## 5 Related Work

### 5.1 Multimodal Instruction Tuning

Instruction tuning has become a central technique for aligning large language models (LLMs) with human intent, enabling them to better follow open-ended natural language instructions. In the multimodal setting, however, aligning both language and vision modalities presents unique challenges. Building upon the success of unimodal instruction tuning methods such as FLAN [Wei et al., 2022], Self-Instruct [Wang et al., 2023], and Direct Preference Optimization (DPO) [Rafailov

et al., 2023], researchers have extended these strategies to vision-language models (VLMs). These models must reason over visual semantics, resolve cross-modal references, and produce grounded, coherent responses—all within the framework of natural language instructions.

Initial efforts such as InstructBLIP [Dai et al., 2023], LLaVA [Liu et al., 2023], and MiniGPT-4 [Zhu et al., 2024] demonstrated the feasibility of aligning VLMs using instruction-following data. More recent advances, including Llava-OV [Li et al., 2024], Infinity-MM [Gu et al., 2024], MAmmoTH-VL [Guo et al., 2024], and VisualWebInstruct [Jia et al., 2025], show that scaling up instruction tuning datasets and introducing diverse tasks can significantly enhance generalization across a wide range of multimodal benchmarks.

## 5.2 Reasoning with Reinforcement Learning

The release of GPT-o1 [Jaech et al., 2024] and DeepSeek-R1 [Guo et al., 2025] has sparked renewed interest in incentivizing reasoning capabilities in LLMs via reinforcement learning (RL). Recent works like SimpleRL-Zoo [Zeng et al., 2025] and Open-Reasoner-Zero [Hu et al., 2025] explore direct RL fine-tuning from base models without relying on additional supervised instruction-tuning phases. Building on this foundation, approaches such as DeepScaler [Luo et al., 2025] and Light-R1 [Wen et al., 2025] incorporate cold-start datasets specifically designed to promote long-form reasoning and step-by-step thought processes.

In parallel, efforts such as DAPO [Yu et al., 2025] and Dr GRPO [Liu et al., 2025] aim to improve the original Group Relative Policy Optimization (GRPO) algorithm, refining reward structures and advantage estimation to more effectively elicit deep reasoning behaviors from LLMs during training.

## 5.3 Multimodal Reinforcement Learning

There is a growing body of work focused on bringing RL-based reasoning into the multimodal domain [Deng et al., 2025, Yang et al., 2025, Huang et al., 2025, Peng et al., 2025]. Inspired by models like DeepSeek-R1, these approaches typically follow a multi-stage pipeline. A common practice involves first performing supervised fine-tuning (SFT) on vision-language data that has been annotated or augmented with detailed reasoning traces, often derived from strong text-only LLMs after converting visual inputs into textual descriptions.

Following the SFT stage, reinforcement learning is used to further enhance the model's reasoning capabilities. While effective, these pipelines often require complex and resource-intensive processes, including visual captioning, teacher model distillation, and tightly coupled SFT+RL orchestration [Wang et al., 2025]. In contrast, our work investigates a more direct and lightweight RL-only approach, aiming to incentivize slow-thinking behavior without relying on large-scale supervision or teacher-based distillation.

# 6 Conclusion

In this paper, we investigated how to more effectively incentivize the reasoning capabilities of multimodal models. Our proposed approaches have shown effectiveness in multimodal reasoning benchmarks. However, our models are still lagging behind human expert performance on more general multimodal tasks like EMMA and MEGA-Bench. We conjecture that this is due to a lack of high-quality multimodal training dataset. In the future, we endeavor to further improve the data quality to improve multimodal reasoning capabilities.

# References

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

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.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025.

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.

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.

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.

Anthropic. Claude 3.5 sonnet model card addendum, 2024. URL `https://www.anthropic.com/claude-3-5-sonnet-model-card-addendum`.

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. *Sort*, 2(4): 0–6.

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.

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.

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, 2024a.

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

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.

Yihe Deng, Hritik Bansal, Fan Yin, Nanyun Peng, Wei Wang, and Kai-Wei Chang. Openvlthinker: An early exploration to complex vision-language reasoning via iterative self-improvement. *arXiv preprint arXiv:2503.17352*, 2025.

Weihao Zeng, Yuzhen Huang, Qian Liu, Wei Liu, Keqing He, Zejun Ma, and Junxian He. Simplerl-zoo: Investigating and taming zero reinforcement learning for open base models in the wild. *arXiv preprint arXiv:2503.18892*, 2025.

Liang Wen, Yunke Cai, Fenrui Xiao, Xin He, Qi An, Zhenyu Duan, Yimin Du, Junchen Liu, Lifu Tang, Xiaowei Lv, et al. Light-r1: Curriculum sft, dpo and rl for long cot from scratch and beyond. *arXiv preprint arXiv:2503.10460*, 2025.

Yifan Du, Zikang Liu, Yifan Li, Wayne Xin Zhao, Yuqi Huo, Bingning Wang, Weipeng Chen, Zheng Liu, Zhongyuan Wang, and Ji-Rong Wen. Virgo: A preliminary exploration on reproducing o1-like mllm. *arXiv preprint arXiv:2501.01904*, 2025.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images, 2016. URL `https://arxiv.org/abs/1603.07396`.

Tanik Saikh, Tirthankar Ghosal, Amish Mittal, Asif Ekbal, and Pushpak Bhattacharyya. Scienceqa: A novel resource for question answering on scholarly articles. *International Journal on Digital Libraries*, 23(3):289–301, 2022.

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.

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.

Haozhe Wang, Long Li, Chao Qu, Fengming Zhu, Weidi Xu, Wei Chu, and Fangzhen Lin. Learning autonomous code integration for math language models. *arXiv preprint arXiv:2502.00691*, 2025.

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, 2024.

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, 2024a.

Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, et al. Mmmu-pro: A more robust multi-discipline multimodal understanding benchmark. *arXiv preprint arXiv:2409.02813*, 2024b.

Yunzhuo Hao, Jiawei Gu, Huichen Will Wang, Linjie Li, Zhengyuan Yang, Lijuan Wang, and Yu Cheng. Can mllms reason in multimodality? emma: An enhanced multimodal reasoning benchmark. *arXiv preprint arXiv:2501.05444*, 2025.

Jiacheng Chen, Tianhao Liang, Sherman Siu, Zhengqing Wang, Kai Wang, Yubo Wang, Yuansheng Ni, Wang Zhu, Ziyan Jiang, Bohan Lyu, et al. Mega-bench: Scaling multimodal evaluation to over 500 real-world tasks. *arXiv preprint arXiv:2410.10563*, 2024a.

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*, 2024b.

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.

Jingyi Zhang, Jiaxing Huang, Huanjin Yao, Shunyu Liu, Xikun Zhang, Shijian Lu, and Dacheng Tao. R1-vl: Learning to reason with multimodal large language models via step-wise group relative policy optimization. *arXiv preprint arXiv:2503.12937*, 2025.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*, 2022.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, 2023.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741, 2023.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL `https://openreview.net/forum?id=vvoWPYqZJA`.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. In *ICLR*, 2024.

Shuhao Gu, Jialing Zhang, Siyuan Zhou, Kevin Yu, Zhaohu Xing, Liangdong Wang, Zhou Cao, Jintao Jia, Zhuoyi Zhang, Yixuan Wang, et al. Infinity-mm: Scaling multimodal performance with large-scale and high-quality instruction data. *arXiv preprint arXiv:2410.18558*, 2024.

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. *arXiv preprint arXiv:2412.05237*, 2024.

Yiming Jia, Jiachen Li, Xiang Yue, Bo Li, Ping Nie, Kai Zou, and Wenhu Chen. Visualwebinstruct: Scaling up multimodal instruction data through web search. *arXiv preprint arXiv:2503.10582*, 2025.

Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum. Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base model. *arXiv preprint arXiv:2503.24290*, 2025.

Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y. Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Tianjun Zhang, Li Erran Li, Raluca Ada Popa, and Ion Stoica. Deepscaler: Surpassing o1-preview with a 1.5b model by scaling rl. `https://pretty-radio-b75.notion.site/DeepScaleR-Surpassing-O1-Preview-with-a-1-5B-Model-by-Scaling-RL-19681902c1468005bed8ca303013a` 2025. Notion Blog.

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.

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.

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.

# Appendix