This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Boosting MLLM Reasoning with Text-Debiased Hint-GRPO

Qihan Huang[1,2], Weilong Dai[2], Jinlong Liu[2], Wanggui He[2], Hao Jiang[2],
Mingli Song[1], Jingyuan Chen[1], Chang Yao[1], Jie Song[1,†]
[1] Zhejiang University, [2] Alibaba Group
{qh.huang,brooksong,jingyuanchen,changy,sjie}@zju.edu.cn,
{chenlong0104.chen,aoshu.jh}@alibaba-inc.com,
LJLwykqh@126.com,wanggui.hwg@taobao.com

Abstract

MLLM reasoning has drawn widespread research for its excellent problem-solving capability. Current reasoning methods fall into two types: PRM, which supervises the intermediate reasoning steps, and ORM, which supervises the final results. Recently, DeepSeek-R1 has challenged the traditional view that PRM outperforms ORM, which demonstrates strong generalization performance using an ORM method (i.e., *GRPO). However, current MLLM's GRPO algorithms still struggle to handle challenging and complex multimodal reasoning tasks* (e.g., *mathematical reasoning). In this work, we reveal two problems that impede the performance of GRPO on the MLLM: Low data utilization and Text-bias. Low data utilization refers to that GRPO cannot acquire positive rewards to update the MLLM on difficult samples, and text-bias is a phenomenon that the MLLM bypasses image condition and solely relies on text condition for generation after GRPO training. To tackle these problems, this work proposes Hint-GRPO that improves data utilization by adaptively providing hints for samples of varying difficulty, and text-bias calibration that mitigates text-bias by calibrating the token prediction logits with image condition in test-time. Experiment results on three base MLLMs across eleven datasets demonstrate that our proposed methods advance the reasoning capability of original MLLM by a large margin, exhibiting superior performance to existing MLLM reasoning methods. Our code is available at `https://github.com/hqhQAQ/Hint-GRPO`.*

1. Introduction

MLLM (Multimodal LLM) reasoning has attracted wide research interest for its exceptional problem-solving capability, especially after the release of OpenAI's o1 model. Existing reasoning methods can be categorized into two types: PRM (Process Reward Method), which supervises the intermediate reasoning steps, and ORM (Outcome Reward Method), which supervises the final reasoning results. Previously, most methods [5, 17, 32, 33] consider PRM to be superior to ORM, leveraging numerous strategies (*e.g.*, MCTS [5, 17, 33], DPO [37]) for PRM training.

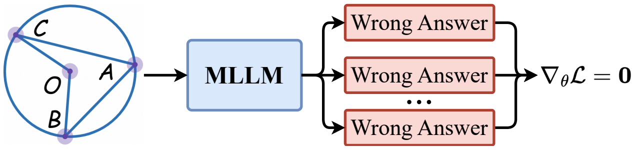
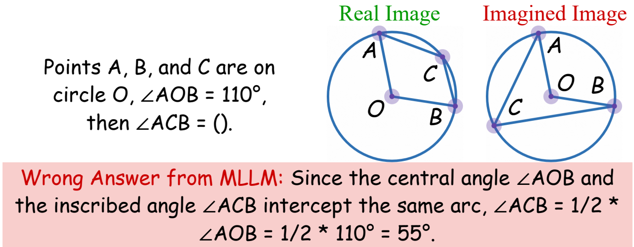


Figure 1. **Low data utilization** of GRPO: If all answers are incorrect, the zero loss gradients ($\nabla_\theta \mathcal{L} = \mathbf{0}$) will invalidate the sample.



Points A, B, and C are on circle O, ∠AOB = 110°, then ∠ACB = ().

Wrong Answer from MLLM: Since the central angle ∠AOB and the inscribed angle ∠ACB intercept the same arc, ∠ACB = 1/2 * ∠AOB = 1/2 * 110° = 55°.

Figure 2. **Text-bias** of GRPO, where the MLLM ignores real image and uses its imagined image from text to generate outputs.

Recently, DeepSeek-R1 [11] has overturned the conventional belief that PRM is superior to ORM, which employs the GRPO (Group Relative Policy Optimization [25]) reinforcement learning algorithm that calculates rewards for model predictions solely based on the model's reasoning results, without supervision of intermediate reasoning steps. DeepSeek-R1 discovers that ORM training can induce models to engage in self-reflection, extending the chain of thought to reach correct answers, thereby achieving excellent reasoning performance. After the emergence of DeepSeek-R1, many researchers explore ORM methods for LLM & MLLM reasoning, with most focusing

---

[1] † Corresponding author.

on the GRPO algorithm (*e.g.*, Open-R1 [7], simpleRL-reason [35], R1-V [4], R1-Multimodal-Journey [20]). Some research (Open-Reasoner-Zero [15]) also finds that PPO-based [24] ORM algorithm could achieve similar performance to GRPO.

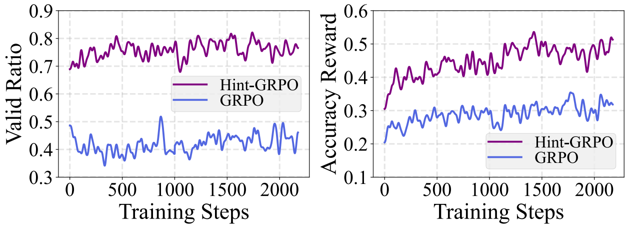
However, while current MLLM's ORM methods (*e.g.*, R1-V [4]) perform well on simple visual tasks (*e.g.*, counting objects in the image), they are insufficient for more challenging and complex multimodal reasoning tasks (*e.g.*, mathematical reasoning). In this work, we identify two problems that hinder the performance of ORM methods on MLLM reasoning: **(1) Low data utilization**; **(2) Text-bias**.

For the first problem, GRPO requires to first prompt the MLLM to generate multiple predictions for the same question, and then assign rewards to the predictions based on their correctness. However, due to the insufficient reasoning capability of the original MLLM, all the predictions for the difficult question could be incorrect in MLLM. In this situation, GRPO cannot update the model as the calculated advantages in GRPO are all zero, rendering the training sample invalid, as shown in Figure 1. Figure 3 (a) also shows the ratio of valid samples in each batch during training (Qwen2-VL-7B [30] on the mathematical reasoning dataset), verifying the severity of low data utilization.

To address this problem, this work proposes *Hint-GRPO*, which provides additional hints for MLLM to solve questions of high difficulty. Specifically, for difficult questions where MLLM cannot find the correct answer (invalid samples), Hint-GRPO provides the initial part of correct reasoning steps (according to a certain ratio) to the MLLM as hints, allowing it to complete the remaining reasoning steps and arrive at the final answer. As shown in Figure 3 (a), Hint-GRPO effectively leads MLLM to generate correct answers, thereby improving the data utilization. Furthermore, Hint-GRPO can adaptively adjust hint ratios for questions of varying difficulty, **avoiding excessive hints** for simple questions, thereby allowing for optimal utilization of the dataset and achieving better performance.

For the second problem (text-bias), this work observes a phenomenon that during GRPO training, MLLM learns to directly infer the final answer from text condition while ignoring image condition, as shown in Figure 2. Besides, Figure 5 (a) demonstrates that as GRPO training progresses, the accuracy of MLLM (with the image condition removed) on the test set also increases.

To tackle this problem, this work proposes a *text-bias calibration* method **in test-time** to reduce MLLM reasoning errors caused by ignoring image condition. Inspired by CFG (classifier-free guidance) [13] in image generation, this text-bias calibration method first uses the MLLM to generate prediction results (token logits) with and without image condition, and then calibrates the token logits using the difference between them.



(a) Data Utilization Rate **(b) Accuracy Reward**

Figure 3. Data utilization rate & reward of GRPO & Hint-GRPO.

We perform comprehensive experiments to validate the performance of the proposed methods. Specifically, we apply text-debiased Hint-GRPO to three base MLLMs across eleven datasets (mathematical reasoning & universal multimodal reasoning), and the experiment results demonstrate that our methods improve the reasoning capability of original MLLM by a large margin, achieving significantly superior performance to existing MLLM reasoning methods.

To sum up, the main contributions of this work can be summarized as follows:

- We identify and thoroughly analyze two problems that hinder the performance of GRPO on MLLM reasoning: (1) Low data utilization; (2) Text-bias.
- We propose two methods (Hint-GRPO and text-bias calibration), which effectively mitigate these two problems.
- Experiment results show that our proposed methods achieve significantly superior performance to existing MLLM reasoning methods.

2. Related Work

PRM for MLLM Reasoning. Most existing MLLM reasoning methods follow the PRM (Process Reward Method) paradigm, which employs fine-grained supervision on the intermediate reasoning steps. LLaVA-o1 [32] trains the MLLM to generate structured reasoning steps. AtomThink [31] directly trains a process reward model to assess the reasoning steps, using the framework of Markov decision process (MDP) learning. Mulberry [33], M-STAR [17], and AR-MCTS [5] utilize Monte Carlo Tree Search (MCTS) to estimate the reasoning steps. LLaVA-Reasoner-DPO [37] employs direct preference optimization (DPO) [22] to improve the intermediate reasoning process. Virgo [6] constructs a long-thought dataset to enable the MLLM with long reasoning capability. LlamaV-o1 [28] adopts curriculum learning to train the MLLM in an easy-to-hard manner. However, DeepSeek-R1 [11] points out current PRM methods struggle to accurately evaluate the reasoning steps and suffer from a serious reward hacking problem, resulting in suboptimal performance. On the contrary, ORM enables accurate evaluation by simply comparing the reasoning results with the ground truth, thus avoid-

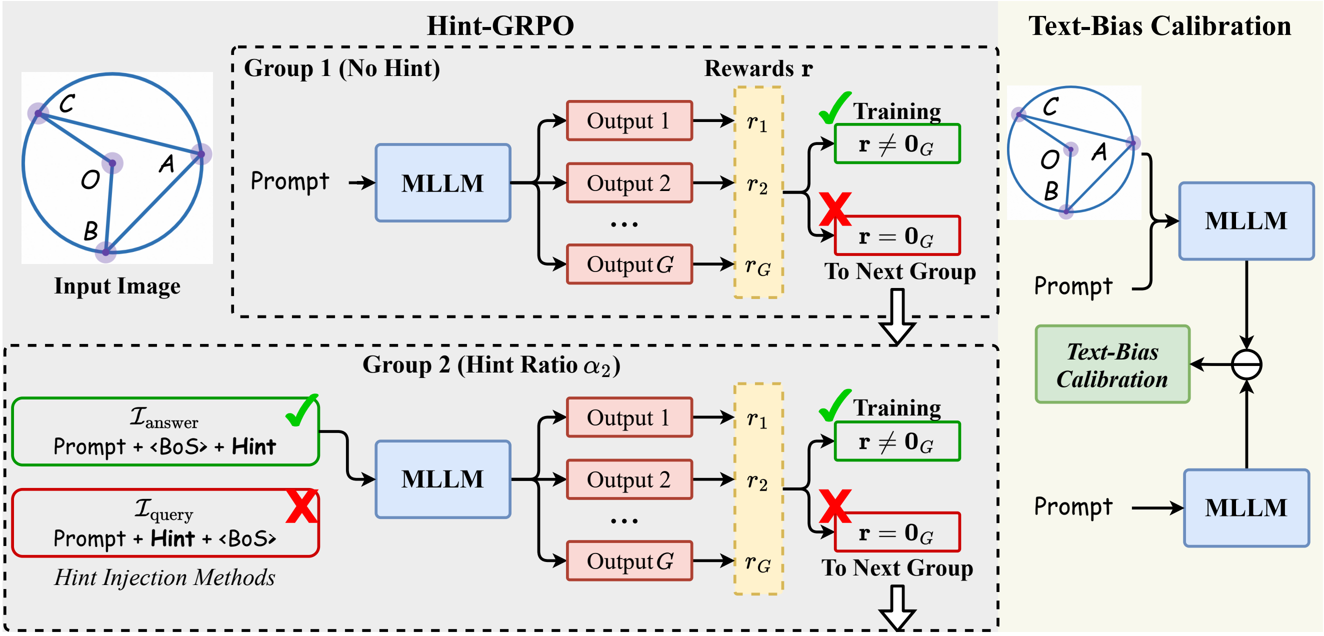


Figure 4. Framework of Hint-GRPO and text-bias calibration. Specifically, Hint-GRPO adaptively provides hints to the samples and selects the most eligible group for training, mitigating the low data utilization problem. Text-bias calibration increases the intensity of image condition **in test-time**, alleviating the text-bias problem. Note that "<BoS>" denotes the *beginning-of-sequence* token in MLLM.

ing the reward hacking problem.

ORM for MLLM Reasoning. ORM (Outcome Reward Method) only supervises the reasoning results using the ground truth, regardless of the intermediate reasoning steps. ORM methods such as GRPO can achieve strong generalization performance, verified by DeepSeek-R1. After the emergence of DeepSeek-R1, more researchers are shifting towards the ORM paradigm (*i.e.*, GRPO) for MLLM reasoning. Specifically, Open-R1-Multimodal [2] establishes the first GRPO baseline for MLLM reasoning. R1-V [4] demonstrates that GRPO performs well in simple multimodal reasoning tasks, *e.g.*, counting objects in the image. R1-Multimodal-Journey [20] significantly accelerates the training speed using the vLLM package [16]. Video-R1 [8] generalizes Open-R1-Multimodal to video reasoning. However, current methods are still limited in addressing more challenging multimodal reasoning tasks (*e.g.*, mathematical reasoning). Our work identifies two problems that hinder the GRPO performance in MLLM reasoning (low data utilization and text-bias), and proposes two methods to address them (Hint-GRPO and text-bias calibration).

3. Method

3.1. Preliminaries

Supervised Fine-tuning (SFT). SFT trains the LLM on curated query-output pairs to improve its instruction-following ability. The objective of SFT is to maximize the following objective:

$$\mathcal{J}_{\text{SFT}}(\theta) = \mathbb{E}[q, o \sim P(Q,O)] \left(\frac{1}{|o|} \sum_{t=1}^{|o|} \log \pi_\theta(o_t | q, o_{<t}) \right),$$

where q, o is the query-output pair sampled from the SFT dataset $P(Q,O)$, θ denotes the model parameters, $\pi_\theta(o_t|q, o_{<t})$ represents the logit of the model predicting the next token o_t from q and previous tokens $o_{<t}$.

Proximal Policy Optimization (PPO). PPO is an actor-critic RL algorithm that is widely used in the RL fine-tuning stage of LLM. In particular, it optimizes the model by maximizing the following objective:

$$\mathcal{J}_{\text{PPO}}(\theta) = \mathbb{E}[q \sim P(Q), o \sim \pi_{\theta_{\text{old}}}(O|q)] \frac{1}{|o|} \sum_{t=1}^{|o|} \frac{\pi_\theta(o_t|q, o_{<t})}{\pi_{\theta_{\text{old}}}(o_t|q, o_{<t})} A_t,$$

where $\pi_{\theta_{\text{old}}}$ is the old model, A_t represents the advantage function measuring how the t-th token's prediction deviates from average, based on the rewards $\{r_{\geq t}\}$ and a learned value function V_ψ. The **min & clip operations** for avoiding extreme values are omitted here for simplicity.

Group Relative Policy Optimization (GRPO). GRPO directly calculates A_t using the average reward of multiple sampled outputs, eliminating the additional value function V_ψ in PPO. Specifically, GRPO samples a group of outputs $\{o_1, o_2, ..., o_G\}$ from the old model $\pi_{\theta_{\text{old}}}$, and then optimizes the model by maximizing the following objective (**min & clip operations** are also omitted here):

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{\text{old}}}(O|q)]$$

$$\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\left\{\frac{\pi_\theta(o_{i,t}|q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q,o_{i,<t})}\hat{A}_{i,t} - \beta D_{\text{KL}}[\pi_\theta||\pi_{\text{ref}}]\right\},$$

where $D_{\text{KL}}[\pi_\theta||\pi_{\text{ref}}]$ serves as a regularization term that prevents the new model π_θ from deviating too far from the original model π_{ref} (the model before training). As an ORM method, GRPO provides the reward r_i at the end of each output o_i ($r_i = 1$ if the reasoning result is correct, otherwise $r_i = 0$), and sets the advantage $\hat{A}_{i,t}$ of all tokens in o_i as the normalized reward ($\mathbf{r} = \{r_1, r_2, ..., r_G\}$, mean$(\cdot)$ denotes the average, and std$(\cdot)$ denotes the standard deviation):

$$\hat{A}_{i,t} = \tilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}. \tag{1}$$

3.2. Hint-GRPO

3.2.1. Low Data Utilization

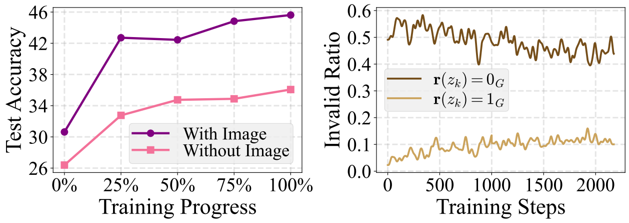
Low data utilization refers to the problem that in current GRPO training, many training samples fail to provide effective feedback to the MLLM, as shown in Figure 1. Specifically, due to the limited reasoning ability of the original MLLM, the generated reasoning results on these training samples are **all incorrect**. This situation (*i.e.*, $r_i = 0$ for all $\{r_i\}_{i=1}^{G}$) causes each advantage $\hat{A}_{i,t} = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})} = 0$, thus hindering the MLLM training (note that std$(\mathbf{r})$ has a small offset to avoid division by zero). Furthermore, we can analyze the invalidity of these training samples by examining the gradients of the optimizing objective $\mathcal{J}_{\text{GRPO}}(\theta)$ on the MLLM parameters θ:

$$\nabla_\theta \mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{\text{old}}}(O|q)]$$

$$\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\left\{\frac{\hat{A}_{i,t}\nabla_\theta\pi_\theta(o_{i,t}|q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q,o_{i,<t})} - \beta\nabla_\theta D_{\text{KL}}[\pi_\theta||\pi_{\text{ref}}]\right\}.$$

When all $\hat{A}_{i,t}$ equal 0, the update of MLLM parameters θ only depends on the less important KL divergence $D_{\text{KL}}[\pi_\theta||\pi_{\text{ref}}]$ and is unrelated to the accuracy of reasoning results. This prevents such training samples from providing effective feedback for model optimization.

In this work, we conduct an in-depth analysis of the low data utilization problem. Specifically, we propose a *data utilization rate* to measure the proportion of effective samples in each batch during GRPO training. Let $\{z_k\}_{k=1}^{B}$ denote a training batch of B samples, and the MLLM generates G outputs for each sample, then $\mathbf{r}(z_k) \in \mathbb{R}^G$ represents the correctness of all G outputs of z_k, *i.e.*, $\mathbf{r}(z_k)_i = 1$ if the output result is correct, otherwise $\mathbf{r}(z_k)_i = 0$. Next, we can determine the sample z_k is valid **if and only if** std$(\mathbf{r}(z_k)) \neq 0$. In other words, std$(\mathbf{r}(z_k)) = 0$ indicates that $\mathbf{r}(z_k)$ equals a zero vector $\mathbf{0}_G$ or a ones vector $\mathbf{1}_G$, resulting in the calculated advantages $\hat{A}_{i,t}$ being all zero and thus invalidating the sample z_k. Finally, the data utilization rate $S_{\text{valid}} \in [0, 1]$ for this training batch is calculated as below ($\mathbb{1}\{\cdot\}$ denotes the indicator function):



(a) Accuracy w/o Image **(b) Invalid Ratio**

Figure 5. (a) Qwen2-VL-7B's test accuracy w/ & w/o image in GRPO training. (b) The ratio of two types of invalid samples.

$$S_{\text{valid}} = \frac{1}{B}\sum_{k=1}^{B}\mathbb{1}\{\text{std}(\mathbf{r}(z_k)) \neq 0\}. \tag{2}$$

As shown in Figure 3 (a), the original GRPO exhibits a low S_{valid} (40% to 50%) for Qwen2-VL-7B during training. In addition, we also calculate the ratio of two situations where the training sample is invalid: (1) $\mathbf{r}(z_k)$ equals $\mathbf{0}_G$, *i.e.*, all outputs of z_k are incorrect. (2) $\mathbf{r}(z_k)$ equals $\mathbf{1}_G$, *i.e.*, all outputs of z_k are correct. Figure 5 (b) presents that situation (2) only accounts for a small proportion (within 10%), which corresponds to the phenomenon of low rewards during GRPO training in Figure 3 (b). These findings verify that GRPO suffers from the low correctness of MLLM outputs, thus constraining the training of MLLM reasoning.

3.2.2. Hint-GRPO Implementation

To tackle the problem of low data utilization, this work proposes Hint-GRPO, which provides **reasoning hints** to lead the MLLM to generate correct answers for difficult training samples, thus improving the number of samples with std$(\mathbf{r}(z_k)) \neq 0$. Specifically, let h denote the reasoning hint, then the MLLM predicts $\pi_\theta(o_{i,t}|q, h, o_{i,<t})$ (the next-token prediction logit) with h as condition in Hint-GRPO. The implementation of Hint-GRPO consists of three parts: (1) Dataset construction; (2) Hint injection method; and (3) Hint adaptation strategy.

(1) Dataset construction

The training of Hint-GRPO requires a dataset containing multiple samples with image, query text, reasoning steps, and the ground-truth answer. Here, we choose the LLaVA-CoT dataset [32], a high-quality and influential training dataset for MLLM reasoning, as our base dataset. Nevertheless, the original LLaVA-CoT dataset still necessitates **two modifications** before being used for Hint-GRPO:

First, we use GPT-4o to split the original reasoning steps (a long text) of each sample into multiple structured steps, which allows us to control the hint level by directly adjusting the number of reasoning steps in the hint.

Second, we convert multi-choice questions to fill-in-the-blank format by removing the options, stopping the MLLM from getting correct answers through random guessing.

(2) Hint injection method

In $\pi_\theta(o_{i,t}|q, h, o_{i,<t})$ (the next-token prediction logit of Hint-GRPO), the MLLM requires to inject h (the reasoning hint split from the correct reasoning steps in the dataset) into the model. The simplest baseline $\mathcal{I}_{\text{query}}$ (named *hint injection in query*) in Figure 4 is to append the hint to the original query text, *e.g.*, appending "Solve the question following the hint: {*Reasoning Hint*}". However, this baseline has two problems: First, even with hints in the prompt, the model sometimes ignores them and starts reasoning from scratch. Second, and more critically, query text **with hint** in training-time is inconsistent with query text **without hint** in test-time, leading to poor test performance.

To address these problems, we propose $\mathcal{I}_{\text{answer}}$ (named *hint injection in answer*) in Figure 4, which keeps the query text unchanged while using the reasoning hint as the beginning of model output, letting the model complete the remaining steps to reach the final answer. $\mathcal{I}_{\text{answer}}$ makes the MLLM faithfully continue reasoning based on the hint, and the query text requires no hint in both training and testing, performing significantly better than the baseline $\mathcal{I}_{\text{query}}$.

(3) Hint adaptation strategy

Hint-GRPO requires adjusting the hint level to achieve optimal results. Specifically, if the hint level is too low, it still cannot lead the MLLM to the correct answers, remaining trapped in the low data utilization problem; If the hint level is too high, the MLLM can reach correct answers without reasoning, hindering effective use of the data for MLLM reasoning training. Therefore, based on our constructed dataset, we employ a hint ratio $\alpha \in [0, 1]$ to adjust the hint level. In detail, for the total L correct reasoning steps of a sample, we extract the **first** $L \cdot \alpha$ steps as hint (denoted as h_α), and thus a higher α indicates a higher hint level. Next, we explore three strategies to set the hint level: (1) Fixed hint level; (2) Random hint level; (3) Adaptive hint level.

Fixed hint level sets the same hint ratio α for each training sample. However, this strategy can lead to simple questions being given excessive hints, resulting in insufficient training for MLLM reasoning. Experiments in Table 3 show that this strategy achieves suboptimal performance.

Random hint level samples $\alpha \sim \text{Uniform}(0, 1)$ randomly for each training sample, inspired by diffusion model training [14, 26]. Experiments in Table 3 imply that this strategy is still suboptimal, because it also cannot provide an appropriate hint level for different samples.

Adaptive hint level. To address this problem, we propose to adaptively adjust the hint level based on the difficulty level of different samples. To this end, this strategy extends the original GRPO's single group output per sample (G outputs per group) to M groups, and assigns different hint ratios to each group. Specifically, let $\{\alpha_i\}_{i=1}^{M}$ denote the hint ratios for these M groups, then α_i is set to $\frac{i-1}{M}$. As the hint ratio progressively increases, the sample difficulty steadily decreases from group 1 through group M. Finally, this strategy selects the first group with existing correct answers (in the order from group 1 to group M) for training. By selecting the most appropriate hint level, this strategy both avoids the low-data utilization problem and mitigates the issue of excessive hints preventing the MLLM from reasoning, thereby achieving optimal results.

Efficiency analysis: Although this strategy increases the group number to M, the MLLM still uses only one group for training. Besides, we use vLLM [16] to significantly speed up generation. Therefore, compared to the original strategy, this strategy only increases the training time by 20.5% when M is 2, as shown in S2.1 of the appendix.

3.3. Text-Bias Calibration

This work reveals a phenomenon (named text-bias) that the MLLM trained with GRPO tends to directly reason from text condition while ignoring image condition, as shown in Figure 2. Besides, Figure 5 (a) demonstrates that as GRPO training continues, the accuracy of MLLM (with the image condition removed) on the test set also improves. We suspect that this phenomenon stems from that many query texts in current MLLM reasoning datasets can fully describe the questions, leading the MLLM to rely solely on text. However, when the query text is insufficient to describe the entire question, this type of reasoning will result in errors and lower performance.

To mitigate this text-bias problem, this work proposes text-bias calibration, which can directly emphasize the image condition **in test-time**. Inspired by CFG (classifier-free guidance) [13] in image generation, text-bias calibration first uses the MLLM to predict token logits with and without image condition, and then calibrates the token logits according to their differences. Specifically, let q_{img} and q_{text} denote the image condition and text condition, then $\hat{\pi}_\theta(o_t|q_{\text{img}}) = \pi_\theta(o_t|q_{\text{img}}, q_{\text{text}}, o_{<t})$ and $\hat{\pi}_\theta(o_t) = \pi_\theta(o_t|q_{\text{text}}, o_{<t})$ represent the token logit predicted with and without image condition (note that $\hat{\pi}_\theta$ abbreviates the original π_θ). Finally, the calibrated token logit $\hat{\pi}_\theta^{\text{calibrated}}(o_t|q_{\text{img}})$ is calculated as below following CFG:

$$\hat{\pi}_\theta^{\text{calibrated}}(o_t|q_{\text{img}}) = \hat{\pi}_\theta(o_t|q_{\text{img}}) + \gamma \cdot (\hat{\pi}_\theta(o_t|q_{\text{img}}) - \hat{\pi}_\theta(o_t)),$$

where γ is a hyper-parameter controlling the intensity of image condition. Intuitively, this text-bias calibration method makes the calibrated token logits move away from $\hat{\pi}_\theta(o_t)$ and closer to the real $\hat{\pi}_\theta(o_t|q_{\text{img}})$, thereby alleviating the problem of ignoring image condition. Furthermore, we provide a theoretical analysis on text-bias calibration in S1 of the appendix, for a more comprehensive understanding.

4. Experiments

Implementation details. Following existing GRPO methods (*e.g*., R1-V [4], Open-R1-Multimodal [2]), we conduct the main experiments on mathematical (geometry) reasoning tasks, using Qwen2-VL-7B and Qwen2.5-VL-3B as base models. Besides, we also follow LLaVA-o1 [32] to conduct experiments on the universal multimodal reasoning tasks, using Llama-3.2-11B-Vision as base model. During training, we adopt AdamW optimizer with a learning rate of 5e-5, and train the model on 8 GPUs for 2 epochs (following R1-V) with a batch size of 1 per GPU. A system prompt is used to instruct the model to generate responses in the format of "<think>{*Reasoning Steps*}</think> <answer>{*Reasoning Result*}</answer>". We also use DeepSpeed [1, 23] to facilitate the model training through ZeRO-3 optimization. Besides, we use the vLLM package [16] to accelerate the generation process in GRPO, allocating 1 GPU for generation and 7 GPUs for training. The hyper-parameter M for adaptive hint is set to 3, and the hyper-parameter γ for text-bias calibration is set to 0.8.

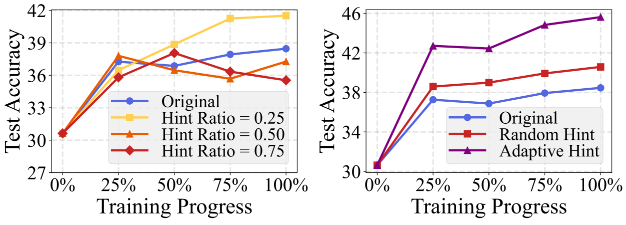
Training dataset. For the geometry reasoning, we extract geometry samples from the LLaVA-CoT dataset [32] for training, with a total size of 7840. For the universal multimodal reasoning, we use the whole LLaVA-CoT dataset for training, with a total size of 100,000.

Test benchmark. For the geometry reasoning, we follow R1-V to use the same subset of Geo170K dataset [9] for evaluation. Besides, we also incorporate other geometry data from existing datasets (MathVista [19], MMStar [3], MathVerse [36], Math-Vision [29], MM-Math [27], and WeMath [21]) for evaluation, towards a comprehensive comparison. For the universal multimodal reasoning, we follow LLaVA-o1 to evaluate the model on 6 benchmarks: MMStar [3], MMBench [18], MMVet [34], MathVista [19], AI2D [12], and Hallusion [10].

Baseline methods. For the geometry reasoning, we compare our method with the SFT baseline, PRM methods (Mulberry), and existing GRPO methods (Open-R1-Multimodal, R1-V). Specifically, Open-R1-Multimodal and R1-V are both GRPO methods but are trained on different datasets: GEOQA_R1V and open-r1-8k-verified respectively. For the universal multimodal reasoning, we compare our method with the influential LLaVA-o1.

4.1. Comparison Analysis

Mathematical (Geometry) reasoning. Table 1 shows the comparison results of different methods on two base



(a) Fixed **(b) Random & Adaptive**

Figure 6. Ablation experiments of hint adaptation strategy for Qwen2-VL-7B on the Geo170K dataset.

MLLMs (Qwen2-VL-7B & Qwen2.5-VL-3B) across seven datasets. Several conclusions can be drawn from Table 1:

(1) Supervised fine-tuning (SFT) reduces the performance of original MLLMs on **out-of-domain data**, indicating that SFT merely memorizes knowledge mechanically, without learning patterns of reasoning that can generalize to new problems.

(2) PRM method (Mulberry) outperforms SFT, demonstrating that PRM can enhance the intermediate reasoning steps of MLLM to some extent.

(3) GRPO methods (Open-R1-Multimodal, R1-V, and GRPO on our dataset) demonstrate superior performance to both PRM and SFT methods, implying that GRPO enables the MLLM to independently learn how to think and reason for tackling new problems.

(4) As analyzed in section 3, GRPO methods suffer from the low data utilization and text-bias problems, hindering the model training. Our proposed text-debiased Hint-GRPO can alleviate these two problems, achieving significantly superior performance to existing GRPO methods.

Universal multimodal reasoning. Table 2 presents the comparison results of different methods on Llama-3.2-11B-Vision across six datasets. Both trained on the LLaVA-CoT dataset, our text-debiased Hint-GRPO **outperforms** the original LLaVA-o1 (trained with SFT). Nevertheless, the improvement of the GRPO method on universal multimodal reasoning is not as significant as in geometry reasoning. We discover that this issue may stem from the accuracy estimation in GRPO training, *e.g*., in samples requiring bounding box localization, the model's responses can hardly match the ground-truth answers perfectly, making such samples all incorrect and invalid for GRPO training. Therefore, a more robust accuracy estimation method is required for universal multimodal reasoning. For example, regarding the aforementioned samples, IoU values can be used to measure the accuracy of model responses.

4.2. Ablation Experiments

4.2.1. Hint-GRPO

This section provides the ablation experiments of three parts in Hint-GRPO: (1) Dataset construction; (2) Hint injection

Method	Geo170K	MathVista (Geometry)	MMStar (Geometry)	MathVerse (Geometry)	Math-Vision (Geometry)	MM-Math (Geometry)	WeMath (Geometry)	Average
Qwen2-VL-7B								
Original	30.63	44.50	40.52	27.92	10.89	8.73	35.52	30.40
SFT	37.53	41.66	37.07	14.47	2.86	1.95	26.84	25.50
Mulberry	33.55	52.17	42.24	17.68	6.06	10.69	42.07	32.08
Open-R1-Multimodal	35.68	45.55	40.52	28.78	11.43	6.78	38.22	31.56
R1-V	38.72	47.26	41.38	28.12	12.51	8.83	41.44	33.19
GRPO	38.46	48.82	42.24	30.10	12.02	10.37	40.52	33.92
Hint-GRPO	45.62	52.77	43.97	31.68	14.38	14.35	45.23	37.60
Debiased Hint-GRPO	**46.68**	**54.19**	**45.69**	**32.18**	**14.99**	**14.61**	**45.86**	**38.55**
Qwen2.5-VL-3B								
Original	35.41	47.50	41.38	26.17	8.08	11.69	39.66	32.17
SFT	43.24	46.64	43.10	20.03	8.62	2.29	32.64	30.40
Open-R1-Multimodal	48.67	45.88	44.83	27.44	12.71	14.68	40.75	35.10
R1-V	47.48	48.67	47.41	31.42	14.58	12.10	42.07	36.55
GRPO	45.49	49.84	48.27	30.88	14.48	12.35	43.45	36.83
Hint-GRPO	53.32	54.79	51.72	33.68	17.09	16.73	44.89	40.88
Debiased Hint-GRPO	**55.31**	**56.11**	**52.59**	**34.09**	**17.39**	**17.51**	**46.78**	**41.95**

Table 1. Experiment results of two base MLLMs on the geometry reasoning tasks. Bold font denotes the best result.

Method	MMStar	MMBench	MMVet	MathVista	AI2D	Hallusion	Average
Original	49.8	65.8	57.6	48.6	77.3	40.3	56.6
LLaVA-o1	57.6	75.0	60.3	54.8	85.7	47.8	63.5
Ours	**60.7**	**75.8**	**64.2**	**56.8**	**86.6**	**50.7**	**65.8**

Table 2. Experiment results of Llama-3.2-11B-Vision on the universal multimodal reasoning tasks. Bold font denotes the best result.

method; (3) Hint adaptation strategy.

Dataset construction. We convert the multi-choice questions to fill-in-the-blank format in the original LLaVA-CoT dataset, and the original dataset and converted dataset are denoted as $\mathcal{D}_{\text{original}}$ and $\mathcal{D}_{\text{new}}$, respectively. Table 3 shows that GRPO trained on $\mathcal{D}_{\text{new}}$ outperforms GRPO trained on $\mathcal{D}_{\text{original}}$ notably, by addressing the problem that the MLLM takes shortcuts (*i.e.*, random guessing) to correct answers rather than through reasoning. Therefore, we use $\mathcal{D}_{\text{new}}$ as the training dataset in the subsequent experiments.

Hint injection method. This work proposes two methods to inject the hint into the MLLM: *hint injection in query* (denoted as $\mathcal{I}_{\text{query}}$) and *hint injection in answer* (denoted as $\mathcal{I}_{\text{answer}}$). Table 3 demonstrates that Hint-GRPO $\mathcal{I}_{\text{query}}$ severely degrades performance, resulting from the inconsistency between query text **with hint** during training and query text **without hint** during testing.

Hint adaptation strategy. Figure 6 (a) shows the effect of hint ratio α on the strategy of **fixed hint level**, implying that a low-level hint ratio (*e.g.*, 0.25) can improve MLLM performance compared to the original GRPO, as hints enhance data utilization and facilitate the training. However, excessive hint levels (*e.g.*, 0.50, 0.75) impair MLLM performance, which causes the MLLM to skip reasoning and undermine their reasoning capability.

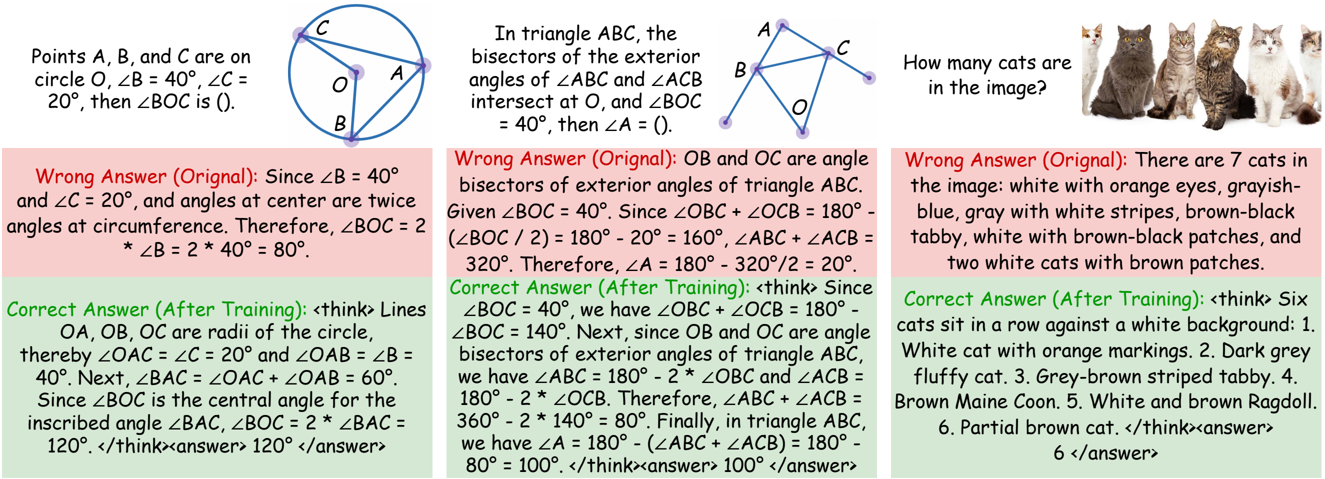
Method	MMStar	MathVista (Geometry)	MMStar (Geometry)	Avg.
Original	30.63	44.50	40.52	40.04
GRPO + $\mathcal{D}_{\text{original}}$	35.81	43.94	37.93	40.40
GRPO + $\mathcal{D}_{\text{new}}$	**38.46**	**48.82**	**42.24**	**44.59**
Hint-GRPO + $\mathcal{I}_{\text{query}}$	41.64	47.17	39.66	43.91
Hint-GRPO + $\mathcal{I}_{\text{answer}}$	**45.62**	**52.77**	**43.97**	**48.78**

Table 3. Ablation experiments of training dataset and hint injection method on Qwen2-VL-7B.

Base Model	0.0	0.4	0.8†	1.2	1.6
Qwen2-VL-7B	45.62	46.29	**46.68**	46.02	44.69
Qwen2.5-VL-3B	53.32	54.77	**55.31**	54.38	53.58

Table 4. Ablation experiments of γ for text-bias calibration on the Geo170K dataset. † denotes the selected one.

As shown in Figure 6 (b), while the random hint level improves MLLM performance over the original GRPO framework by increasing data diversity and utilization during training, the adaptive hint level strategy achieves even better performance. It does so by intelligently assessing the difficulty of each sample and dynamically assigning the most suitable hint level, thus facilitating a more effective and targeted learning process.



Points A, B, and C are on circle O, ∠B = 40°, ∠C = 20°, then ∠BOC is ().

Wrong Answer (Orignal): Since ∠B = 40° and ∠C = 20°, and angles at center are twice angles at circumference. Therefore, ∠BOC = 2 * ∠B = 2 * 40° = 80°.

Correct Answer (After Training): <think> Lines OA, OB, OC are radii of the circle, thereby ∠OAC = ∠C = 20° and ∠OAB = ∠B = 40°. Next, ∠BAC = ∠OAC + ∠OAB = 60°. Since ∠BOC is the central angle for the inscribed angle ∠BAC, ∠BOC = 2 * ∠BAC = 120°. </think><answer> 120° </answer>

In triangle ABC, the bisectors of the exterior angles of ∠ABC and ∠ACB intersect at O, and ∠BOC = 40°, then ∠A = ().

Wrong Answer (Orignal): OB and OC are angle bisectors of exterior angles of triangle ABC. Given ∠BOC = 40°. Since ∠OBC + ∠OCB = 180° - (∠BOC / 2) = 180° - 20° = 160°, ∠ABC + ∠ACB = 320°. Therefore, ∠A = 180° - 320°/2 = 20°.

Correct Answer (After Training): <think> Since ∠BOC = 40°, we have ∠OBC + ∠OCB = 180° - ∠BOC = 140°. Next, since OB and OC are angle bisectors of exterior angles of triangle ABC, we have ∠ABC = 180° - 2 * ∠OBC and ∠ACB = 180° - 2 * ∠OCB. Therefore, ∠ABC + ∠ACB = 360° - 2 * 140° = 80°. Finally, in triangle ABC, we have ∠A = 180° - (∠ABC + ∠ACB) = 180° - 80° = 100°. </think><answer> 100° </answer>

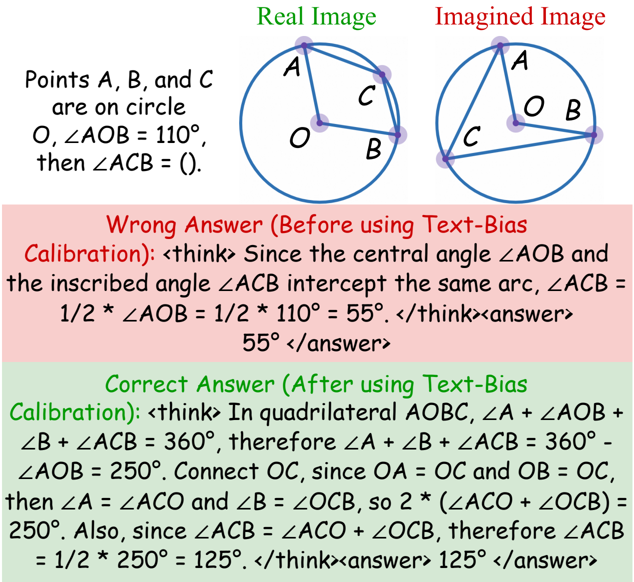
How many cats are in the image?

Wrong Answer (Orignal): There are 7 cats in the image: white with orange eyes, grayish-blue, gray with white stripes, brown-black tabby, white with brown-black patches, and two white cats with brown patches.

Correct Answer (After Training): <think> Six cats sit in a row against a white background: 1. White cat with orange markings. 2. Dark grey fluffy cat. 3. Grey-brown striped tabby. 4. Brown Maine Coon. 5. White and brown Ragdoll. 6. Partial brown cat. </think><answer> 6 </answer>

(a) Geometry Reasoning **(b) Geometry Reasoning** **(c) Universal Multimodal Reasoning**

Figure 7. Visualization examples of MLLMs' outputs before/after Hint-GRPO.



Points A, B, and C are on circle O, ∠AOB = 110°, then ∠ACB = ().

Wrong Answer (Before using Text-Bias Calibration): <think> Since the central angle ∠AOB and the inscribed angle ∠ACB intercept the same arc, ∠ACB = 1/2 * ∠AOB = 1/2 * 110° = 55°. </think><answer> 55° </answer>

Correct Answer (After using Text-Bias Calibration): <think> In quadrilateral AOBC, ∠A + ∠AOB + ∠B + ∠ACB = 360°, therefore ∠A + ∠B + ∠ACB = 360° - ∠AOB = 250°. Connect OC, since OA = OC and OB = OC, then ∠A = ∠ACO and ∠B = ∠OCB, so 2 * (∠ACO + ∠OCB) = 250°. Also, since ∠ACB = ∠ACO + ∠OCB, therefore ∠ACB = 1/2 * 250° = 125°. </think><answer> 125° </answer>

Figure 8. Visualization example of Qwen2-VL-7B's outputs before/after text-bias calibration.

4.2.2. Text-Bias Calibration

As shown in Table 1, text-debiased Hint-GRPO outperforms Hint-GRPO using the calibration operation in test-time, which increases the image conditioning intensity for alleviating the text-bias problem. Besides, Table 4 provides the ablation experiments for the effect of γ on text-bias calibration, revealing that an excessively high γ value degrades performance due to over-correction.

4.3. Visualization Analysis

Visualization before/after Hint-GRPO. Figure 7 demonstrates the visualization examples of MLLMs' outputs before/after Hint-GRPO. Figure 7 (a) and (b) show that after Hint-GRPO training, the MLLM (Qwen2-VL-7B) conducts more thorough analysis of difficult geometry problems, reaching correct solutions through extended reasoning processes (within "<think>" and "</think>" symbols). Besides, Figure 7 (c) presents that Hint-GRPO also achieves excellent performance in universal multimodal reasoning on Llama-3.2-11B-Vision.

Visualization before/after text-bias calibration. As shown in Figure 8, before text-bias calibration, the MLLM (Qwen2-VL-7B) ignores the real image and uses the imagined image from text to generate the wrong answer. After increasing the image conditioning intensity with text-bias calibration, the MLLM successfully concentrates on the real image and generates the correct answer from it.

5. Conclusion

In this work, we identify and provide a thorough analysis of two problems that hinder the performance of GRPO on MLLM reasoning: Low data utilization and Text-bias. In detail, low data utilization occurs when GRPO fails to obtain positive rewards for updating the MLLM on difficult samples. Text-bias is a phenomenon where the MLLM disregards the image condition after GRPO training. To tackle these two problems, we propose two key contributions: (1) Hint-GRPO, which improves data utilization by providing adaptive hints for samples of varying difficulty, and (2) text-bias calibration, which mitigates text-bias in test-time by calibrating the token prediction logits with image condition. We conduct experiments on three base MLLMs across eleven datasets, showing that our proposed method achieves significantly superior performance to the original model, PRM methods, and existing GRPO methods. We hope our method and dataset (will be made publicly available) can contribute to the community of MLLM reasoning.

6. Acknowledgements

This work is supported by Zhejiang Province High-Level Talents Special Support Program "Leading Talent of Technological Innovation of Ten-Thousands Talents Program" (No. 2022R52046), Alibaba-Zhejiang University Joint Research Institute of Frontier Technologies, and Key Research and Development Project of Yunnan Province (Grant No.202402AD080006).

References

[1] Reza Yazdani Aminabadi, Samyam Rajbhandari, Ammar Ahmad Awan, Cheng Li, Du Li, Elton Zheng, Olatunji Ruwase, Shaden Smith, Minjia Zhang, Jeff Rasley, et al. Deepspeed-inference: enabling efficient inference of transformer models at unprecedented scale. In *SC22: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–15. IEEE, 2022. 6

[2] Li Bo. Open-r1-multimodal: Investigate r1's paradigm in multimodal models. `https://github.com/EvolvingLMMs-Lab/open-r1-multimodal`, 2025. 3, 6

[3] Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? *arXiv preprint arXiv:2403.20330*, 2024. 6

[4] 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. 2, 3, 6

[5] Guanting Dong, Chenghao Zhang, Mengjie Deng, Yutao Zhu, Zhicheng Dou, and Ji-Rong Wen. Progressive multimodal reasoning via active retrieval. *arXiv preprint arXiv:2412.14835*, 2024. 1, 2

[6] 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. 2

[7] Hugging Face. Open r1: A fully open reproduction of deepseek-r1. `https://github.com/huggingface/open-r1`, 2025. 2

[8] Kaituo Feng. Video-r1: Towards super reasoning ability in video understanding. `https://github.com/tulerfeng/Video-R1`, 2025. 3

[9] Jiahui Gao, Renjie Pi, Jipeng Zhang, Jiacheng Ye, Wanjun Zhong, Yufei Wang, Lanqing Hong, Jianhua Han, Hang Xu, Zhenguo Li, et al. G-llava: Solving geometric problem with multi-modal large language model. *arXiv preprint arXiv:2312.11370*, 2023. 6

[10] Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, et al. Hallusionbench: an advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14375–14385, 2024. 6

[11] 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. 1, 2

[12] Tuomo Hiippala, Malihe Alikhani, Jonas Haverinen, Timo Kalliokoski, Evanfiya Logacheva, Serafina Orekhova, Aino Tuomainen, Matthew Stone, and John A Bateman. Ai2d-rst: a multimodal corpus of 1000 primary school science diagrams. *Language Resources and Evaluation*, 55:661–688, 2021. 6

[13] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 2, 5

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 5

[15] Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, and Heung-Yeung Shum Xiangyu Zhang. Open-reasoner-zero: An open source approach to scaling reinforcement learning on the base model. `https://github.com/Open-Reasoner-Zero/Open-Reasoner-Zero`, 2025. 2

[16] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023. 3, 5, 6

[17] Wei Liu, Junlong Li, Xiwen Zhang, Fan Zhou, Yu Cheng, and Junxian He. Diving into self-evolving training for multimodal reasoning. *arXiv preprint arXiv:2412.17451*, 2024. 1, 2

[18] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pages 216–233. Springer, 2024. 6

[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. 6

[20] Fanqing Meng. R1-multimodal-journey: A jourery to real multimodel r1. `https://github.com/FanqingM/R1-Multimodal-Journey`, 2025. 2, 3

[21] Runqi Qiao, Qiuna Tan, Guanting Dong, Minhui Wu, Chong Sun, Xiaoshuai Song, Zhuoma GongQue, Shanglin Lei, Zhe Wei, Miaoxuan Zhang, et al. We-math: Does your large multimodal model achieve human-like mathematical reasoning? *arXiv preprint arXiv:2407.01284*, 2024. 6

[22] 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. 2

[23] Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: Memory optimizations toward training

trillion parameter models. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–16. IEEE, 2020. 6

[24] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 2

[25] 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. 1

[26] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 5

[27] Kai Sun, Yushi Bai, Ji Qi, Lei Hou, and Juanzi Li. Mm-math: Advancing multimodal math evaluation with process evaluation and fine-grained classification. *arXiv preprint arXiv:2404.05091*, 2024. 6

[28] Omkar Thawakar, Dinura Dissanayake, Ketan More, Ritesh Thawkar, Ahmed Heakl, Noor Ahsan, Yuhao Li, Mohammed Zumri, Jean Lahoud, Rao Muhammad Anwer, et al. Llamav-o1: Rethinking step-by-step visual reasoning in llms. *arXiv preprint arXiv:2501.06186*, 2025. 2

[29] 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, 2025. 6

[30] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 2

[31] Kun Xiang, Zhili Liu, Zihao Jiang, Yunshuang Nie, Runhui Huang, Haoxiang Fan, Hanhui Li, Weiran Huang, Yihan Zeng, Jianhua Han, et al. Atomthink: A slow thinking framework for multimodal mathematical reasoning. *arXiv preprint arXiv:2411.11930*, 2024. 2

[32] Guowei Xu, Peng Jin, Li Hao, Yibing Song, Lichao Sun, and Li Yuan. Llava-o1: Let vision language models reason step-by-step. *arXiv preprint arXiv:2411.10440*, 2024. 1, 2, 4, 6

[33] Huanjin Yao, Jiaxing Huang, Wenhao Wu, Jingyi Zhang, Yibo Wang, Shunyu Liu, Yingjie Wang, Yuxin Song, Haocheng Feng, Li Shen, et al. Mulberry: Empowering mllm with o1-like reasoning and reflection via collective monte carlo tree search. *arXiv preprint arXiv:2412.18319*, 2024. 1, 2

[34] 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. 6

[35] 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. `https://hkust-nlp.notion.site/simplerl-reason`, 2025. Notion Blog. 2

[36] 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. 6

[37] Ruohong Zhang, Bowen Zhang, Yanghao Li, Haotian Zhang, Zhiqing Sun, Zhe Gan, Yinfei Yang, Ruoming Pang, and Yiming Yang. Improve vision language model chain-of-thought reasoning. *arXiv preprint arXiv:2410.16198*, 2024. 1, 2