Beyond Where to Look: Trajectory-Guided Reinforcement Learning for Multimodal RLVR

Jinda Lu[1], Junkang Wu[1], Jinghan Li[1], Kexin Huang[1], Shuo Yang[2], Mingzhu Chen[1], Jiancan Wu[1], Kuien Liu[3], and Xiang Wang[1†]

[1] University of Science and Technology of China
[2] Peking University
[3] Institute of Software, Chinese Academy of Sciences

Abstract. Recent advances in Reinforcement Learning with Verifiable Rewards (RLVR) for multimodal large language models (MLLMs) have mainly focused on improving final answer correctness and strengthening visual grounding. However, a critical bottleneck remains: although models can attend to relevant visual regions, they often fail to effectively incorporate visual evidence into subsequent reasoning, leading to reasoning chains that are weakly grounded in visual facts. To address this issue, we propose **T**rajectory-**G**uided **R**einforcement **L**earning (TGRL), which guides the policy model to integrate visual evidence into fine-grained reasoning processes using expert reasoning trajectories from stronger models. We further introduce token-level reweighting and trajectory filtering to ensure stable and effective policy optimization. Extensive experiments on multiple multimodal reasoning benchmarks demonstrate that TGRL consistently improves reasoning performance and effectively bridges the gap between visual perception and logical reasoning.

Keywords: Multimodal Large Language Models · Reinforcement Learning with Verifiable Rewards · Trajectory Guidance

1 Introduction

Reinforcement Learning with Verifiable Rewards (RLVR) has recently emerged as an effective paradigm for enhancing the reasoning capabilities of large language models (LLMs) [8,24,34]. By leveraging automatically verifiable outcomes as reward signals, RLVR enables complex reasoning behaviors such as chain-of-thought (CoT) reasoning and self-reflection [2,5,7]. Building on these advances, recent studies extend RLVR to Multimodal Large Language Models (MLLMs), aiming to improve reasoning over visual inputs in multimodal tasks [6,13,32].

However, unlike purely linguistic reasoning in LLMs, multimodal reasoning requires jointly modeling **visual perception** and **coherent reasoning**. Recent works have identified perception as a key bottleneck in multimodal reasoning, as standard RLVR frameworks primarily optimize outcome-level correctness

† Corresponding author. Email: xiangwang@ustc.edu.cn.

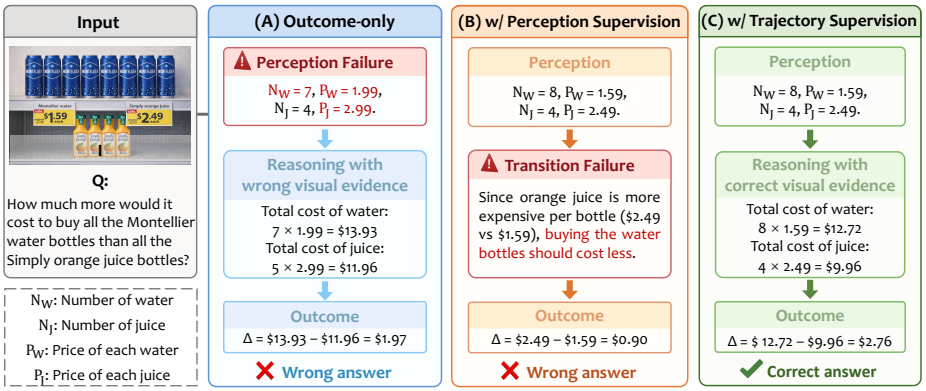

Fig. 1: Comparison of different supervision strategies in multimodal RLVR. (A) Outcome-only supervision may lead to perception failures, where incorrect visual evidence is used for reasoning. (B) Adding perception supervision improves visual grounding but does not guarantee correct reasoning transitions. (C) Our trajectory supervision aligns the perception–reasoning process, enabling reasoning grounded in correct visual evidence and leading to correct answers.

without enforcing visual grounding [29, 32]. As illustrated in Fig. 1(A), without explicit supervision of the perceptual process, models may misinterpret visual content and produce reasoning chains based on incorrect perceptual signals. To address this, recent studies attempt to strengthen the perceptual stage of multimodal reasoning through two main directions:

- **Image Augmentation**: input-level transformations such as cropping, randomly masking, and noise injection that enhance the model's sensitivity to critical visual regions [12, 29];
- **Perception Reward**: auxiliary reward signals that explicitly guide the model toward task-relevant visual cues, often derived by comparing the model's outputs with those from a stronger model (*e.g.*, Gemini-2.5 Pro) [36].

However, these approaches mainly address **where** the model should look, rather than **how** visual evidence is incorporated into the reasoning process. Even when perception is accurate, existing RLVR frameworks do not explicitly enforce how perceived visual evidence should be translated into textual descriptions and utilized during reasoning, allowing models to produce answers without faithfully grounding on visual evidence, as illustrated in Fig. 1. (B).

In this paper, we revisit the optimization flow of multimodal RLVR and show that existing approaches either rely solely on outcome-level supervision or introduce supervision only at the perception stage, leaving the transition from perception to reasoning unconstrained (Fig. 1. (C)). To address this limitation, we propose **T**rajectory-**G**uided **R**einforcement **L**earning (TGRL), which incorporates expert reasoning trajectories to align the perception–reasoning transition during RLVR training. Expert trajectories provide explicit perception–reasoning

paths that serve as structured supervision signals, enabling more effective credit assignment along the multimodal reasoning trajectory.

In summary, our contributions are threefold:

- We systematically analyze the optimization flow of multimodal RLVR and identify a critical bottleneck in the perception–reasoning transition.
- We propose Trajectory-Guided Reinforcement Learning (TGRL), which introduces trajectory-level supervision to align the perception–reasoning transition during RLVR training.
- Extensive experiments on various benchmarks demonstrate that TGRL consistently improves performance over existing RLVR baselines.

2 Preliminaries

In this section, we revisit Reinforcement Learning with Verifiable Rewards (RLVR) procedure and representative RLVR optimization strategies for multi-modal large language models (MLLMs).

2.1 Task Formulation

Reinforcement Learning with Verifiable Rewards (RLVR) for MLLMs: RLVR enhances multi-modal large language models (MLLMs) by aligning model outputs with verifiable answers. Given a multi-modal input with ground truth from a batch of B samples, $(\mathrm{q}^b, \mathrm{I}^b) \in \{\mathrm{q}^b, \mathrm{I}^b\}_{b=1}^B$, and $y^b \in \{y^b\}_{b=1}^B$, where I^b, q^b, and y^b denotes the image, question, and ground-truth respectively, the model $\boldsymbol{\pi}_\theta$ generates output $\boldsymbol{o}^b$ containing a reasoning process and a prediction $\hat{y}^b$. The prediction is enclosed in `\boxed{.}` while reasoning is delimited by `<think>...</think>`, enabling automated verification against the ground truth answers. RLVR employs a binary reward function $\mathbf{R}(\cdot)$ to determine whether the answer is correct by comparing the model prediction $\hat{y}^b$ with ground truth y^b. The goal of RLVR is to maximize the reward function, formalized as:

$$\mathcal{J}_{\mathrm{RLVR}}(\theta) = \max_{\theta} \mathbb{E}_{\{(\mathrm{q}^b, \mathrm{I}^b), y^b)\}_{b=1}^B} \mathbb{E}_{\boldsymbol{o}^b \sim \boldsymbol{\pi}_\theta(\cdot|\mathrm{q}^b, \mathrm{I}^b)} [\mathbf{R}(\hat{y}^b, y^b)]. \tag{1}$$

2.2 RLVR Algorithms

Group Relative Policy Optimization (GRPO). As a widely adopted RLVR optimization strategy, GRPO stabilizes training by computing advantages within response groups [24]. Concretely, given a batch of samples $\{(\mathrm{q}^b, \mathrm{I}^b), y^b\}_{b=1}^B$, the GRPO objective is:

$$\begin{aligned} \mathcal{J}_{\mathrm{GRPO}}(\theta) = \mathbb{E}_{\substack{\mathrm{q}^b, \mathrm{I}^b \sim \mathcal{D} \\ \boldsymbol{o}_i^b \sim \boldsymbol{\pi}_{\theta_{\mathrm{old}}}(\cdot)}} \Bigg[\frac{1}{G} \sum_{i=1}^{G} \frac{1}{|o_i^b|} \sum_{t=1}^{|o_i^b|} \min \Bigg(\frac{\boldsymbol{\pi}_\theta(\boldsymbol{o}_{i,t}^b \mid \boldsymbol{o}_{i,<t}^b, \mathrm{q}^b, \mathrm{I}^b)}{\boldsymbol{\pi}_{\theta_{\mathrm{old}}}(\boldsymbol{o}_{i,t}^b \mid \boldsymbol{o}_{i,<t}^b, \mathrm{q}^b, \mathrm{I}^b)} \hat{A}_i^b, \\ \mathrm{clip} \left(\frac{\boldsymbol{\pi}_\theta(\boldsymbol{o}_{i,t}^b \mid \boldsymbol{o}_{i,<t}^b, \mathrm{q}^b, \mathrm{I}^b)}{\boldsymbol{\pi}_{\theta_{\mathrm{old}}}(o_{i,t}^b \mid \boldsymbol{o}_{i,<t}^b, \mathrm{q}^b, \mathrm{I}^b)}, 1 \pm \epsilon \right) \hat{A}_i^b \Bigg) - \beta \cdot \mathbb{D}_{KL}(\boldsymbol{\pi}_\theta \| \boldsymbol{\pi}_{\mathrm{ref}}) \Bigg], \end{aligned} \tag{2}$$

where G is the rollout group size for each input, and the clip function restricts the importance ratio within $[1-\epsilon, 1+\epsilon]$. Moreover, the advantage is:

$$\begin{cases} \hat{A}_i^b = \dfrac{\mathbf{R}_i^b - \text{mean}(\mathbf{R}^b)}{\text{std}(\mathbf{R}^b)}, \\ \mathbf{R}_i^b = \mathbb{I}(\texttt{is\_equivalent}(\hat{y}^b, y^b)), \end{cases} \tag{3}$$

where $\mathbb{I}(\cdot)$ is the indicator function, $\hat{y}^b$ is the extracted predictions from the model output $\boldsymbol{o}_i^b$, and $\texttt{is\_equivalent}(\cdot)$ compares the extracted prediction $\hat{y}^b$ with the ground truth y^b.

Decoupled Clip and Dynamic Sampling Policy Optimization (DAPO). DAPO [37] improves upon GRPO by removing KL regularization and introducing clip-higher, dynamic sampling, and token-level loss from the GRPO loss, achieving new state-of-the-art performance. Specifically, the DAPO objective for the batch of samples $\{(\mathrm{I}^b, \mathrm{q}^b), y^b\}_{b=1}^B$ can be formalized as:

$$\begin{aligned} \mathcal{J}_{\text{DAPO}}(\theta) = \mathbb{E}_{\substack{\mathrm{q}^b, \mathrm{I}^b \sim \mathcal{D} \\ \boldsymbol{o}_i^b \sim \boldsymbol{\pi}_{\theta_{\text{old}}}(\cdot)}} \Bigg[\frac{1}{\sum |\boldsymbol{o}_i^b|} \sum_{i=1}^{G} \sum_{t=1}^{|\boldsymbol{o}_i^b|} \min \Bigg(\frac{\boldsymbol{\pi}_\theta(\boldsymbol{o}_{i,t}^b \mid \boldsymbol{o}_{i,<t}^b, \mathrm{q}^b, \mathrm{I}^b)}{\boldsymbol{\pi}_{\theta_{\text{old}}}(o_{i,t}^b \mid \boldsymbol{o}_{i,<t}^b, \mathrm{q}^b, \mathrm{I}^b)} \hat{A}_i^b, \\ \text{clip} \left(\frac{\boldsymbol{\pi}_\theta(\boldsymbol{o}_{i,t}^b \mid \boldsymbol{o}_{i,<t}^b, \mathrm{q}^b, \mathrm{I}^b)}{\boldsymbol{\pi}_{\theta_{\text{old}}}(\boldsymbol{o}_{i,t} \mid \boldsymbol{o}_{i,<t}^b, \mathrm{q}^b, \mathrm{I}^b)}, 1-\epsilon_{\text{low}}, 1+\epsilon_{\text{high}} \right) \hat{A}_i^b \Bigg) \Bigg], \end{aligned} \tag{4}$$

where ϵ_{low} and ϵ_{high} decouple the clipping thresholds for negative and positive advantages, respectively. In this work, we apply our token-reweighting strategy to both GRPO and DAPO, demonstrating its general applicability across various RLVR optimization strategies.

3 Revisiting Optimization Flow in Multimodal RLVR

3.1 The Multimodal Reasoning Pipeline

To understand the limitations of current paradigms, we formalize the generation process of an MLLM $\boldsymbol{\pi}_\theta$. Given a question q^b and an image I^b from a batch $\{(\mathrm{q}^b, \mathrm{I}^b), y^b\}_{b=1}^B$, the generation process of a MLLM $\boldsymbol{\pi}_\theta$ can be decomposed into:

$$(\mathrm{q}^b, \mathrm{I}^b) \xrightarrow{\boldsymbol{\pi}_\theta} \mathrm{p}^b \xrightarrow{\boldsymbol{\pi}_\theta} \mathrm{r}^b \xrightarrow{g(\cdot)} \hat{y}^b, \tag{5}$$

where p^b represents responses describing the image I^b, and r^b denotes the subsequent reasoning responses conditioned on $(\mathrm{p}^b, \mathrm{q}^b)$. Together, they form the complete response $\boldsymbol{o}^b = \{\mathrm{p}^b, \mathrm{r}^b\}$, from which the final prediction $\hat{y}^b$ is derived as $\hat{y}^b = g(\mathrm{r}^b)$ using an extraction function $g(\cdot)$. Under this formulation, the distribution of generating the trajectory $\boldsymbol{o}^b$ can be factorized as:

$$\boldsymbol{\pi}_\theta(\boldsymbol{o}^b|\mathrm{q}^b, \mathrm{I}^b) = \underbrace{\boldsymbol{\pi}_\theta(\mathrm{p}^b|\mathrm{q}^b, \mathrm{I}^b)}_{\text{Perception}} \cdot \underbrace{\boldsymbol{\pi}_\theta(\mathrm{r}^b|\mathrm{p}^b, \mathrm{q}^b)}_{\text{Reasoning}}. \tag{6}$$

This factorization exposes a two-stage dependency:

– Perception modeling: $\boldsymbol{\pi}_\theta(\mathrm{p}^b|\mathrm{q}^b, \mathrm{I}^b)$
– Perception to reasoning transition: $\boldsymbol{\pi}_\theta(\mathrm{r}^b|\mathrm{p}^b, \mathrm{q}^b)$

3.2 Bottlenecks in Current Optimization Flows

The essential difference among existing approaches lies in which components of this pipeline are optimized, and how reward is assigned along ($\mathrm{p} \to \mathrm{r} \to \hat{y}$) during RLVR. We show how existing RLVR strategies distribute reward signals along the multimodal reasoning trajectory.

❶ Outcome-Level Supervision. Standard outcome-level RLVR optimizes the model solely based on the correctness of the final prediction:

$$\mathcal{J}(\theta) = \max_\theta \ \mathbb{E}_{\boldsymbol{o}^b \sim \boldsymbol{\pi}_\theta(\cdot|\mathrm{q}^b, \mathrm{I}^b)} \big[\mathbf{R}(\hat{y}^b, y^b)\big], \tag{7}$$

where $\boldsymbol{o}^b = (\mathrm{p}^b, \mathrm{r}^b)$ is the generated trajectory, $\hat{y}^b = g(\mathrm{r}^b)$ is the extracted prediction, and $\mathbf{R}$ is a binary reward. Under the simple REINFORCE objective [31], the gradient is derived as:

$$\begin{aligned}\nabla_\theta \mathcal{J}(\theta) &= \mathbb{E}_{\boldsymbol{o}^b \sim \boldsymbol{\pi}_\theta(\cdot|\mathrm{q}^b, \mathrm{I}^b)} \Big[\mathbf{R}(\hat{y}^b, y^b) \cdot \nabla_\theta \log \boldsymbol{\pi}_\theta(\boldsymbol{o}^b \mid \mathrm{q}^b, \mathrm{I}^b)\Big] \\ &= \mathbb{E}_{\boldsymbol{o}^b \sim \boldsymbol{\pi}_\theta(\cdot|\mathrm{q}^b, \mathrm{I}^b)} \Big[\mathbf{R}(\hat{y}^b, y^b) \cdot \underbrace{\Big(\nabla_\theta \log \boldsymbol{\pi}_\theta(\mathrm{r}^b|\mathrm{p}^b, \mathrm{q}^b) + \nabla_\theta \log \boldsymbol{\pi}_\theta(\mathrm{p}^b|\mathrm{q}^b, \mathrm{I}^b)\Big)}_{\text{Outcome gradient on full trajectory}} \Big]\end{aligned} \tag{8}$$

Limitations. Although the gradient propagates through both perception and reasoning components, the credit assignment signal is entirely determined by the final correctness. As a result: **(1)** The perception module is not explicitly grounded in verifiable visual evidence. **(2)** The reasoning module is not required to faithfully utilize the perceived facts. This outcome-level supervision allows the model to obtain correct answers via spurious correlations or implicit language priors, rather than through consistent perception—reasoning alignment.

❷ Perception-Level Supervision. To provide denser signals for the perceptual stage, perceptual reward $\mathbf{R}_{\text{per}}$ is introduced to compare the perceived representation p^b with a supervision target $\mathcal{G}^b$ (*e.g.*, grounding annotations):

$$\mathcal{J}(\theta) = \max_\theta \ \mathbb{E}_{\boldsymbol{o}^b \sim \boldsymbol{\pi}_\theta(\cdot|\mathrm{q}^b, \mathrm{I}^b)} \Big[\underbrace{\mathbf{R}(\hat{y}^b, y^b)}_{\text{Outcome}} + \lambda \underbrace{\mathbf{R}_{\text{per}}(\mathrm{p}^b, \mathcal{G}^b)}_{\text{Perception}}\Big]. \tag{9}$$

Its gradient becomes:

$$\begin{aligned}\nabla_\theta \mathcal{J}(\theta) = \mathbb{E}_{\boldsymbol{o}^b \sim \boldsymbol{\pi}_\theta(\cdot|\mathrm{q}^b, \mathrm{I}^b)} \Big[&\underbrace{\mathbf{R}(\hat{y}^b, y^b) \cdot \nabla\theta \log \boldsymbol{\pi}_\theta(\boldsymbol{o}^b \mid \mathrm{q}^b, \mathrm{I}^b)}_{\text{Outcome Gradient}} \\ &+ \underbrace{\lambda \cdot \mathbf{R}_{\text{per}}(p^b, \mathcal{G}^b) \cdot \nabla_\theta \log \boldsymbol{\pi}_\theta(\mathrm{p}^b \mid \mathrm{q}^b, \mathrm{I}^b)}_{\text{Perception Gradient}} \Big],\end{aligned} \tag{10}$$

Limitations. Perception-level supervision improves what is observed, but leaves the perception—reasoning coupling unconstrained. Although the auxiliary reward explicitly supervises the perception module, it does not constrain how the perceived evidence p^b is subsequently utilized in reasoning. In particular, the transition distribution $\boldsymbol{\pi}_\theta(\mathrm{r}^b \mid \mathrm{p}^b, \mathrm{q}^b)$ remains optimized solely through outcome-level feedback. Therefore, even with accurate perceptual grounding, the reasoning process is not required to faithfully ground on p^b.

3.3 The Necessity of Trajectory-Level Supervision

The fundamental weakness of the above approaches lies in their structural decoupling of perception and reasoning. While outcome-level supervision relies on final rewards and perception-level supervision constrains only intermediate representations, neither explicitly regulates the transition from perception to reasoning. To explicitly supervise this transition, we introduce trajectory-level supervision that aligns the perception—reasoning path ($\mathrm{p} \to \mathrm{r}$) with expert trajectories.

Let $\boldsymbol{o}_\phi^b = (\mathrm{p}_\phi^b, \mathrm{r}_\phi^b)$ denote an expert-generated trajectory sampled from $\pi_\phi(\cdot \mid \mathrm{q}^b, \mathrm{I}^b)$. We optimize the following objective:

$$\mathcal{J}(\theta) = \max_\theta \mathbb{E}_{\substack{\boldsymbol{o}_\theta^b \sim \boldsymbol{\pi}_\theta(\cdot|\mathrm{q}^b,\mathrm{I}^b) \\ \boldsymbol{o}_\phi^b \sim \boldsymbol{\pi}_\phi(\cdot|\mathrm{q}^b,\mathrm{I}^b)}} \Big[\underbrace{\mathbf{R}(\hat{y}_\theta^b, y^b)}_{\text{Outcome}} + \underbrace{\boldsymbol{f}(\boldsymbol{o}_\phi^b;\theta)\, \mathbf{R}(\hat{y}_\phi^b, y^b)}_{\text{Expert Trajectory}} \Big], \tag{11}$$

where $\hat{y}_\theta^b = g(r_\theta^b)$, $\hat{y}_\phi^b = g(r_\phi^b)$. The token-level weighting function $\boldsymbol{f}(\boldsymbol{o}_\phi^b;\theta)$ modulates the contribution of each token in expert trajectories, encouraging the student policy to align with structurally coherent perception–reasoning paths. Since $\boldsymbol{o}_\phi^b = \{\mathrm{p}_\phi^b, \mathrm{r}_\phi^b\}$ combines perception and reasoning tokens, we decompose $\boldsymbol{f}(\boldsymbol{o}_\phi^b;\theta)$ according to p_ϕ^b and r_ϕ^b as $\boldsymbol{f}(\mathrm{p}_\phi^b;\theta)$ and $\boldsymbol{f}(\mathrm{r}_\phi^b;\theta)$, respectively. Under the policy gradient framework, the gradient decomposes as:

$$\begin{aligned}\nabla_\theta \mathcal{J}(\theta) = \mathbb{E}_{\substack{\boldsymbol{o}_\theta^b \sim \boldsymbol{\pi}_\theta(\cdot|\mathrm{q}^b,\mathrm{I}^b) \\ \boldsymbol{o}_\phi^b \sim \boldsymbol{\pi}_\phi(\cdot|\mathrm{q}^b,\mathrm{I}^b)}} \Big[& \mathbf{R}(\hat{y}_\theta^b, y^b) \cdot \nabla_\theta \log \boldsymbol{\pi}_\theta(\boldsymbol{o}_\theta^b \mid \mathrm{q}^b, \mathrm{I}^b) + \mathbf{R}(\hat{y}_\phi^b, y^b) \cdot \\ & \Big(\underbrace{\boldsymbol{f}(\mathrm{p}_\phi^b;\theta) \cdot \nabla_\theta \log \boldsymbol{\pi}_\theta(\mathrm{p}_\phi^b \mid \mathrm{q}^b, \mathrm{I}^b)}_{\text{Perception-aligned Gradient}} + \underbrace{\boldsymbol{f}(\mathrm{r}_\phi^b;\theta) \cdot \nabla_\theta \log \boldsymbol{\pi}_\theta(\mathrm{r}_\phi^b \mid \mathrm{p}_\phi^b, \mathrm{q}^b)}_{\text{Reasoning-aligned Gradient}} \Big) \Big].\end{aligned} \tag{12}$$

Unlike outcome-level or perception-level supervision, trajectory-level alignment directly constrains the transition from perception to reasoning, enabling structured credit assignment across the entire multimodal reasoning pipeline.

4 Trajectory-Guided Reinforcement Learning

Motivated by the gradient analysis, we propose a **T**rajectory **G**uided **R**einforcement **L**earning (TGRL) that implements such trajectory-level alignment into the RLVR framework. An overview of TGRL is illustrated in Fig. 2.

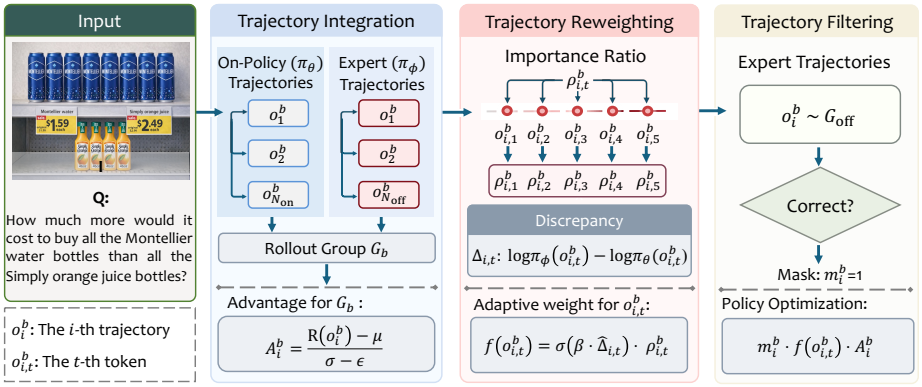

Fig. 2: Overview of **T**rajectory **G**uided **R**einforcement **L**earning (TGRL). Given an image-question pair, we construct a rollout group by integrating on-policy and expert trajectories. Expert trajectories are then reweighted at the token level and filtered by correctness, enabling trajectory-level alignment within RLVR framework.

4.1 Trajectory Integration

As discussed in Section 3, trajectory-level supervision requires aligning current policy with expert trajectories. However, directly fine-tuning on expert trajectories suffers from distribution mismatch [18,33], leading to error accumulation in reasoning trajectories, as early deviations push the model toward states outside its training distribution. Instead of directly fine-tuning, we incorporate expert trajectories into a reinforcement learning framework, allowing expert rollouts to serve as guidance signals while preserving on-policy exploration.

• **Group Construction.** Given a question q^b with image I^b, we construct the rollout group consisting of both on-policy and expert-generated trajectories:

$$\begin{aligned} \mathcal{G}_{\text{on}}^b &= \{\boldsymbol{o}_i^b \sim \boldsymbol{\pi}_{\theta_{\text{old}}}(\cdot \mid \mathrm{q}^b, \mathrm{I}^b)\}_{i=1}^{N_{\text{on}}}, \\ \mathcal{G}_{\text{off}}^b &= \{\boldsymbol{o}_i^b \sim \boldsymbol{\pi}_{\phi}(\cdot \mid \mathrm{q}^b, \mathrm{I}^b)\}_{i=1}^{N_{\text{off}}}. \end{aligned} \tag{13}$$

During optimization, trajectories are sampled from the previous policy $\boldsymbol{\pi}_{\theta_{\text{old}}}$, following standard PPO-style updates. Then, the combined rollout group is:

$$\mathcal{G}^b = \mathcal{G}_{\text{on}}^b \cup \mathcal{G}_{\text{off}}^b. \tag{14}$$

Note that $\mathcal{G}_{\text{off}}^b$ typically contains a small number of expert trajectories, ensuring that the optimization remains predominantly on-policy.

• **Group-Based Advantage Computation.** For each trajectory $\boldsymbol{o}_i^b \in \mathcal{G}^b$, we compute its reward $\mathbf{R}(\boldsymbol{o}_i^b) = \mathbf{R}(\hat{y}_i^b, y^b)$. To enable relative comparison within the group, we normalize the rewards with the mean $\mu_{\mathcal{G}^b}$ and standard deviation $\sigma_{\mathcal{G}^b}$:

$$\mu_{\mathcal{G}^b} = \frac{1}{|\mathcal{G}^b|} \sum_{\boldsymbol{o}_i^b \in \mathcal{G}^b} \mathbf{R}(\boldsymbol{o}_i^b), \quad \sigma_{\mathcal{G}^b} = \mathrm{Std}\big(\{\mathbf{R}(\boldsymbol{o}_i^b)\}_{\boldsymbol{o}_i^b \in \mathcal{G}^b}\big). \tag{15}$$

Then, the normalized advantage is defined as:

$$\bar{A}_i^b = \frac{\mathbf{R}(\boldsymbol{o}_i^b) - \mu_{\mathcal{G}^b}}{\sigma_{\mathcal{G}^b} + \epsilon}. \tag{16}$$

Since advantages are normalized across the joint rollout group, trajectories with relatively higher rewards receive proportionally larger policy updates. In practice, expert trajectories often exhibit higher relative advantages during early training, thus contributing more strongly to the updates. As training progresses, the relative advantage of on-policy rollouts increases, gradually transitioning from expert guidance to on-policy exploration.

4.2 Trajectory Reweighting

To instantiate the weighting function introduced in Section 3, we design a two-stage reweighting mechanism. We first perform importance ratio correction to account for off-policy expert trajectories. Moreover, we introduce adaptive token-level reweighting to emphasize positions where the policy model deviates most from the expert.

• **Importance Ratio Correction.** Expert trajectories are sampled from a stronger model $\boldsymbol{\pi}_\phi$ rather than the previous policy $\boldsymbol{\pi}_{\theta_{\text{old}}}$. Consequently, the standard on-policy importance sampling ratio $\boldsymbol{\pi}_\theta / \boldsymbol{\pi}_{\theta_{\text{old}}}$ is not applicable to these trajectories [23, 24]. To account for this behavior mismatch, we define the importance ratio with respect to the expert policy $\boldsymbol{o}_i^b \sim \mathcal{G}_{\text{off}}^b$ as:

$$\rho_{\theta,i,t}^b = \frac{\boldsymbol{\pi}_\theta(\boldsymbol{o}_{i,t}^b \mid \boldsymbol{o}_{i,<t}^b, \mathrm{q}^b, \mathrm{I}^b)}{\boldsymbol{\pi}_\phi(\boldsymbol{o}_{i,t}^b \mid \boldsymbol{o}_{i,<t}^b, \mathrm{q}^b, \mathrm{I}^b)}. \tag{17}$$

This ratio performs token-level off-policy correction, compensating for the distribution shift between expert-generated trajectories and the current policy.

• **Adaptive Token Reweighting.** While the modified importance sampling ratio corrects for distribution mismatch, not all tokens in expert trajectories provide equally informative guidance. Intuitively, tokens where the current policy significantly underestimates the expert probability indicate larger discrepancies and therefore contain stronger corrective signals. To capture this effect, we introduce a token-level reweighting coefficient. For an expert trajectory $\boldsymbol{o}_i^b = (\boldsymbol{o}_{i,1}^b, \ldots, \boldsymbol{o}_{i,T}^b) \sim \mathcal{G}^b$, we define the discrepancy at step t as:

$$\Delta_{i,t}^b = \log \boldsymbol{\pi}_\phi(\boldsymbol{o}_{i,t}^b \mid \boldsymbol{o}_{i,<t}^b, \mathrm{q}^b, \mathrm{I}^b) - \log \boldsymbol{\pi}_\theta(\boldsymbol{o}_{i,t}^b \mid \boldsymbol{o}_{i,<t}^b, \mathrm{q}^b, \mathrm{I}^b), \tag{18}$$

which measures how much the current policy underestimates the expert token under the same prefix. To stabilize optimization, we normalize the discrepancy within each trajectory:

$$\tilde{\Delta}_{i,t}^b = \frac{\Delta_{i,t}^b - \mu_{\Delta i}^b}{\sigma_{\Delta i}^b + \epsilon}. \tag{19}$$

We then compute a bounded token weight as:

$$w_{i,t}^b = \sigma(\beta \tilde{\Delta}_{i,t}^b), \tag{20}$$

where $\sigma(\cdot)$ denotes the sigmoid function and β controls the sensitivity to discrepancy magnitude. Finally, we define the modulated token-level importance weight, the weight at step t is:

$$\boldsymbol{f}(\boldsymbol{o}_{i,t}^b;\theta) = \rho_{\theta,i,t}^b \cdot w_{i,t}^b, \tag{21}$$

This formulation preserves token-level off-policy correction, while adaptively scaling gradient magnitudes according to the discrepancy between the current and expert policies.

4.3 Trajectory Filtering

Although expert-generated trajectories generally exhibit higher reasoning quality, they do not guarantee correct final predictions. Expert trajectories with incorrect final predictions represent failed reasoning paths. Supervising the model with these samples introduces gradient noise that contradicts the outcome-based reward, thereby destabilizing the optimization process and hindering policy improvement. In contrast, negative samples are more reliably obtained through on-policy exploration, which better reflects the model's current uncertainty and failure modes [11]. Therefore, we restrict expert supervision to trajectories that yield correct outcomes.

• **Positive Trajectory Filtering.** We apply a correctness-based filtering mechanism that selectively incorporates only successful expert trajectories. For an expert-generated trajectory $\boldsymbol{o}_\phi^b$, we define a binary mask:

$$m_\phi^b = \begin{cases} 1, & \mathbf{R}(\hat{y}_\phi^b, y^b) = 1, \\ 0, & \mathbf{R}(\hat{y}_\phi^b, y^b) = 0, \end{cases} \tag{22}$$

where $\hat{y}_\phi^b = g(\mathrm{r}_\phi^b)$. The masked expert contribution is then incorporated into the optimization only when $m_\phi^b = 1$, ensuring that trajectory-level alignment is guided by correct reasoning paths.

4.4 Training Objective

To incorporate expert-generated trajectories into the RLVR framework, TGRL modifies the optimization in three aspects: **(1)** the rollout distribution, **(2)** the advantage normalization scope, and **(3)** the token-level importance weighting. We incorporate trajectory integration, adaptive reweighting, and positive filtering into GRPO [24] and DAPO [37] by replacing the standard importance ratio with a unified token-level coefficient. For each trajectory $\boldsymbol{o}_i^b \in \mathcal{G}^b$, we define the unified token-level coefficient as:

$$\tilde{r}_{i,t}^b = \begin{cases} \dfrac{\boldsymbol{\pi}_\theta(\boldsymbol{o}_{i,t}^b \mid \boldsymbol{o}_{i,<t}^b, \mathrm{q}^b, \mathrm{I}^b)}{\boldsymbol{\pi}_{\theta_{\text{old}}}(\boldsymbol{o}_{i,t}^b \mid \boldsymbol{o}_{i,<t}^b, \mathrm{q}^b, \mathrm{I}^b)}, & \boldsymbol{o}_i^b \in \mathcal{G}_{\text{on}}^b, \\ m_\phi^b \cdot \boldsymbol{f}(\boldsymbol{o}_{\phi,t}^b;\theta), & \boldsymbol{o}_i^b \in \mathcal{G}_{\text{off}}^b, \end{cases} \tag{23}$$

where $\boldsymbol{f}(\boldsymbol{o}^b_{\phi,t};\theta) = \rho^b_{\theta,t} \cdot w^b_t$ combines importance ratio correction and token-level reweighting strategy, and m^b_ϕ filters out incorrect expert trajectories.

• **TGRL-GRPO.** We extend GRPO by performing optimization over the joint rollout group $\mathcal{G}^b = \mathcal{G}^b_{\text{on}} \cup \mathcal{G}^b_{\text{off}}$. By normalizing advantages across this unified group as in Equation 16 and replacing the standard importance ratio with the coefficient $\tilde{r}^b_{i,t}$, we obtain the TGRL-GRPO objective as:

$$\begin{aligned}\mathcal{J}_{\text{TGRL-GRPO}}(\theta) =& \mathbb{E}_{\substack{(\mathrm{q}^b,\mathrm{I}^b)\sim\mathcal{D} \\ \boldsymbol{o}^b_i\sim\mathcal{G}^b}}\Bigg[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|\boldsymbol{o}^b_i|}\sum_{t=1}^{|\boldsymbol{o}^b_i|}\min\Big(\tilde{r}^b_{i,t}\bar{A}^b_i, \\ & \text{clip}(\tilde{r}^b_{i,t}, 1-\epsilon, 1+\epsilon)\bar{A}^b_i\Big) - \beta\mathbb{D}_{KL}(\boldsymbol{\pi}_\theta\|\boldsymbol{\pi}_{\text{ref}})\Bigg].\end{aligned} \tag{24}$$

• **TGRL-DAPO.** Similarly, we extend DAPO under the same trajectory-guided framework. The resulting TGRL-DAPO objective is:

$$\begin{aligned}\mathcal{J}_{\text{TGRL-DAPO}}(\theta) =& \mathbb{E}_{\substack{(\mathrm{q}^b,\mathrm{I}^b)\sim\mathcal{D} \\ \boldsymbol{o}^b_i\sim\mathcal{G}^b}}\Bigg[\frac{1}{\sum|\boldsymbol{o}^b_i|}\sum_{i=1}^{G}\sum_{t=1}^{|\boldsymbol{o}^b_i|}\min\Big(\tilde{r}^b_{i,t}\bar{A}^b_i, \\ & \text{clip}(\tilde{r}^b_{i,t}, 1-\epsilon_{\text{low}}, 1+\epsilon_{\text{high}})\bar{A}^b_i\Big)\Bigg].\end{aligned} \tag{25}$$

Discussion. TGRL incorporates expert trajectories into RLVR by modifying the rollout distribution, advantage normalization, and token-level importance weighting. The resulting objectives preserve the underlying gradient structure of GRPO-style RLVR objectives while enabling trajectory-level alignment, achieving a principled balance between expert guidance and on-policy exploration.

5 Experiments

In this section, we validate the effectiveness of **T**rajectory-**G**uided **R**einforcement **L**earning (TGRL) by answering the following **R**esearch **Q**uestions (RQs):
❶ How do different components of TGRL affect overall performance?
❷ How sensitive is TGRL to the hyperparameter β?
❸ How does TGRL compare with state-of-the-art approaches?

5.1 Experimental Settings

Training Data. We conduct training on two multimodal reasoning datasets: Geo3K [20], which contains 2.1K geometry reasoning samples, and ViRL39K [25], a large-scale dataset with 39K multimodal reasoning instances.

Implementation Details. All experiments are implemented using the EasyR1 training framework[†]. We adopt Qwen2.5-VL-3B and Qwen2.5-VL-7B as the student policy π_θ and Qwen2.5-VL-32B as the expert policy π_ϕ for trajectory generation. For Geo3K training, the rollout group size is set to 8, consisting of

[†] https://github.com/hiyouga/EasyR1

Table 1: Component ablation of TGRL on Qwen2.5-VL-7B under GRPO and DAPO. Removing trajectory reweighting or filtering consistently degrades performance, demonstrating the importance of properly utilizing expert trajectories. We set $\beta = 5$ for GRPO and $\beta = 14$ for DAPO.

Method	WeMath	MathVista	MathVerse	MathVision	HalluBench	Avg.
TGRL-GRPO	70.29	71.50	51.08	27.41	72.02	58.46
w/o Filter	68.74	70.80	50.65	26.68	71.04	57.50
w/o Reweight	67.58	70.40	50.14	27.11	69.90	57.03
w/o Expert (GRPO)	67.40	70.50	50.80	27.30	69.80	57.16
TGRL-DAPO	71.05	71.40	55.21	28.61	71.29	59.51
w/o Filter	69.80	70.20	51.74	27.57	70.35	57.93
w/o Reweight	67.93	70.90	50.93	27.20	70.56	57.50
w/o Expert (DAPO)	69.30	70.30	50.60	26.50	67.90	56.92

$N_{\text{on}} = 7$ on-policy trajectories and $N_{\text{off}} = 1$ expert trajectory. For ViRL39K training, the group size is 5, with $N_{\text{on}} = 4$ and $N_{\text{off}} = 1$. Models are trained for 15 epochs on Geo3K and 2 epochs on ViRL39K. Unless otherwise specified, all other hyperparameters follow the default settings.

Evaluation Benchmarks. We evaluate models on both multimodal reasoning and perception benchmarks. *Reasoning benchmarks* include MathVista [19], WeMath [22], MathVision [26], and MathVerse [40]. *Perception benchmarks* include HallusionBench [4].

5.2 Ablation Studies

• **Component Analysis (RQ1).** Table 1 shows the contribution of each component in TGRL. Interestingly, directly introducing expert trajectories does not consistently improve performance, highlighting the importance of properly utilizing expert supervision. Under the GRPO framework, simply introducing expert trajectories brings only marginal gains over the baseline, and removing the reweighting module even leads to performance degradation. This observation suggests that expert trajectories alone may introduce noisy supervision signals.

The proposed trajectory reweighting and filtering mechanisms address this issue by emphasizing informative tokens and discarding incorrect expert trajectories. By combining these components, TGRL achieves the best overall performance. Specifically, TGRL improves the average score 57.16 → 58.46 under GRPO and 56.92 → 59.51 under DAPO, demonstrating the effectiveness of properly leveraging expert trajectories.

• **Hyperparameter Analysis (RQ2).** We analyze the sensitivity of the token-reweighting coefficient β, which controls the strength of discrepancy-based token weighting. Table 2 reports the performance under different β values.

Moderate values of β generally yield the best performance. Specifically, the optimal value is $\beta = 5$ under GRPO and $\beta = 14$ under DAPO. This difference arises from their optimization structures: GRPO performs trajectory-level

Table 2: Sensitivity analysis of the token reweighting coefficient β on the Qwen2.5-VL-7B under GRPO and DAPO. Baseline denotes vanilla GRPO/DAPO training without trajectories. Best results within each block are highlighted in **bold**.

Method	WeMath	MathVista	MathVerse	MathVision	HalluBench	Avg.
GRPO	67.40	70.50	50.80	27.30	69.80	57.16
$\beta = 1$	68.70	70.60	51.03	27.68	71.40	57.88
$\beta = 4$	69.37	70.80	**51.14**	27.60	71.71	58.12
$\beta = \mathbf{5}$	**70.29**	**71.50**	51.08	27.41	**72.02**	**58.46**
$\beta = 7$	68.10	70.54	50.53	27.50	70.14	57.36
$\beta = 10$	68.36	70.24	50.81	**27.83**	69.62	57.37
$\beta = 13$	68.41	70.37	50.53	27.43	69.93	57.33
$\beta = 15$	68.74	69.80	50.78	27.77	71.29	57.68
DAPO	69.30	70.30	50.60	26.50	67.90	56.92
$\beta = 1$	70.50	70.20	51.34	26.81	69.82	57.73
$\beta = 4$	70.20	70.90	50.93	26.67	70.56	57.85
$\beta = 7$	70.97	70.90	51.86	26.24	69.72	57.94
$\beta = 10$	69.22	**71.50**	51.84	**30.26**	70.24	58.61
$\beta = 13$	69.98	70.47	54.10	27.84	70.24	58.53
$\beta = \mathbf{14}$	**71.05**	71.40	**55.21**	28.61	**71.29**	**59.51**
$\beta = 15$	70.90	71.10	51.40	25.67	70.35	57.88

averaging with group-normalized advantages, while DAPO averages gradients across all tokens, resulting in more uniformly distributed gradient signals. Consequently, DAPO requires a stronger token-weighting coefficient to highlight informative tokens. Overall, TGRL remains robust across a wide range of β values and consistently improves over the baseline RLVR methods.

5.3 Comparison with State-of-the-Art Approaches (RQ3)

Table 3 presents the comparison with existing multimodal reasoning approaches on five benchmarks using Qwen-2.5-VL-7B as the base model. TGRL consistently improves both GRPO and DAPO across different training scales, indicating that trajectory-level guidance effectively facilitates multimodal RLVR optimization. With 39K training samples, TGRL-DAPO achieves the best average score of 60.68, outperforming state-of-the-art methods. Notably, our trajectories are generated by Qwen-2.5-VL-32B, whereas methods like Perception-R1 rely on stronger proprietary models (*e.g.*, Gemini 2.5 pro) for supervision, highlighting the effectiveness of our approach even with comparatively weaker expert models.

5.4 Performance Comparisons on Additional Backbones.

In this section, we analyze the impact of the token reweighting coefficient β on the performance of the Qwen2.5-VL-3B model under the GRPO and DAPO frameworks. Table 4 shows the results for different β values. **(1)** Compared to

Table 3: Comparison with state-of-the-art multimodal reasoning methods on five benchmarks using Qwen-2.5-VL-7B as base model. TGRL consistently improves both GRPO and DAPO across various training scales and achieves the best performance.

Method	Data	WeMath	MathVista	MathVerse	MathVision	HalluBench	Avg.
VLAA-Thinker [1]	25K	67.90	68.80	49.90	26.90	68.60	56.42
Perception-R1 [32]	1.4K	72.00	74.20	54.30	28.60	-	-
ThinkLite [28]	11K	69.20	72.70	50.20	27.60	71.00	58.14
NoisyRollout [12]	6.4K	70.30	74.50	53.0	30.60	72.20	60.12
R1-Onevision [35]	165K	62.10	63.90	46.10	22.50	65.60	52.04
OpenVLThinker [3]	50K	67.80	71.50	48.00	25.00	70.80	56.62
Qwen-2.5-VL-7B	-	63.10	67.50	46.20	25.00	64.60	53.28
+ GRPO	2.1K	67.40	70.50	50.80	27.30	69.80	57.16
+ TGRL-GRPO	2.1K	70.29	71.50	51.08	27.41	72.02	58.46
+ GRPO	39K	68.20	70.90	50.46	28.32	70.66	57.71
+ TGRL-GRPO	39K	70.23	73.20	52.50	29.87	71.08	59.38
+ DAPO	2.1K	69.30	70.30	50.60	26.50	67.90	56.92
+ TGRL-DAPO	2.1K	71.05	71.40	**55.21**	28.61	71.29	59.51
+ DAPO	39K	68.84	71.30	50.80	28.94	70.87	58.15
+ TGRL-DAPO	39K	**72.20**	**75.00**	53.51	**31.26**	**72.93**	**60.68**

the baseline GRPO and DAPO models, all configurations with different β values show significant improvements. This clearly demonstrates the effectiveness of the proposed TGRL approach. **(2)** GRPO performs best with a smaller $\beta = 6$, while DAPO requires a larger $\beta = 9$ for optimal performance. This aligns with our analysis: GRPO benefits from smaller reweighting to avoid over-prioritizing tokens, whereas DAPO needs a larger β to better highlight informative tokens across all gradients. **(3)** The greatest improvements over the baseline occur with TGRL-GRPO ($\beta = 6$) and TGRL-DAPO ($\beta = 9$), showing increases of 1.3 and 2.6 points in average score, respectively. These results highlight the importance of the token reweighting mechanism in TGRL, which helps the model focus on the most informative tokens and significantly boosts performance.

6 Related Works

We review related work along three directions: **(1)** RLVR-based multimodal reasoning, **(2)** perception-augmented RLVR, and **(3)** trajectory-guided strategies. We then situate our TGRL within this landscape and clarify its distinctions.

6.1 RLVR for Multimodal Reasoning

Multimodal large language models (MLLMs) have developed rapidly in recent years [10, 14, 15, 17], and more recently, RLVR has further advanced MLLMs' reasoning capabilities. In particular, group-based policy optimization methods

Table 4: Sensitivity analysis of the token reweighting coefficient β over Qwen-2.5-VL 3B. **Bold values**: the best performance within each column.

Method	WeMath	MathVista	MathVerse	MathVision	HalluBench	Avg.
GRPO	61.78	60.1	40.03	23.30	61.09	49.26
$\beta = 1$	61.49	61.6	41.87	25.00	60.67	50.13
$\beta = 4$	62.93	61.2	41.40	24.45	61.30	50.26
$\beta = \mathbf{6}$	63.05	**62.6**	**42.36**	**25.20**	60.99	**50.84**
$\beta = 7$	**63.39**	60.7	41.40	24.80	**63.20**	50.70
$\beta = 10$	62.93	61.2	41.40	24.45	61.30	50.26
$\beta = 13$	62.93	60.7	42.21	25.04	61.41	50.46
$\beta = 15$	62.30	60.9	39.95	25.07	60.57	49.76
DAPO	61.49	58.1	41.52	23.69	59.20	48.80
$\beta = 1$	62.36	60.6	40.99	24.64	58.46	49.41
$\beta = 4$	61.95	61.6	41.25	24.08	61.20	50.02
$\beta = 7$	61.03	62.0	41.31	24.86	60.88	50.02
$\beta = \mathbf{9}$	61.78	**63.19**	41.76	24.54	**61.93**	**50.64**
$\beta = 10$	62.18	62.95	41.12	**25.33**	60.67	50.45
$\beta = 13$	**63.16**	60.64	**42.77**	23.52	60.46	50.11
$\beta = 15$	61.67	60.2	41.22	24.64	61.20	49.79

such as GRPO [24] and DAPO [37] stabilize reinforcement learning by computing relative advantages within groups of sampled responses. Building upon these foundations, several works directly apply large-scale RLVR to multimodal models, showing that verifiable rewards can elicit chain-of-thought reasoning and self-correction behaviors without additional supervision [9, 12, 13, 16].

Beyond direct RL optimization, some approaches incorporate agentic tool use to actively manipulate visual inputs and interleave image–text interactions during reasoning. By enabling iterative visual grounding and refinement, works like [41, 42] enhance the model's ability to utilize visual information. Others leverage reasoning priors from reasoning LLM backbones, such as the Skywork-R1V series [21, 27], or adopt cold-start strategies that warm up models with high-quality SFT data before large-scale RLVR [6, 30]. These works primarily focus on improving reasoning capacity under outcome-level sparse reward.

6.2 Perception-Augmented RLVR

A key challenge in multimodal reasoning is the perception–reasoning gap, where visual grounding errors propagate into downstream reasoning failures. To mitigate this issue, perception-augmented RL methods explicitly strengthen the visual grounding stage.

Existing approaches generally fall into two categories. **(1)** Some methods introduce stochastic perturbations or transformation strategies during training to improve visual robustness [12, 29]; **(2)** Others design auxiliary perception-level rewards to encourage faithful grounding before engaging in higher-level reason-

ing [32]. However, these methods primarily focus on improving perception quality, while relying on sparse outcome-level rewards for the subsequent reasoning process, leaving the perception-to-reasoning transition largely unexplored.

6.3 Trajectory-Guided RLVR

Trajectory-guided reinforcement learning introduces expert trajectories into the policy optimization process to enhance the reasoning capability of the current policy. For text-only LLMs, works such as [33, 38, 39] incorporate successful reasoning trajectories into the RL loop, stabilizing the reasoning distribution while preserving policy-gradient optimization.

However, extending trajectory guidance to multimodal reasoning remains challenging. Perception–reasoning coupling amplifies error propagation, while expert trajectory injection inevitably introduces off-policy distribution mismatch.

6.4 Differences

Unlike perception-oriented methods that primarily enhance visual grounding, TGRL explicitly supervises the transition from perception to reasoning through trajectory-level alignment. Moreover, compared to prior trajectory-guided RL approaches developed for text-only settings, TGRL adapts expert trajectory injection to multimodal RLVR by carefully handling perception–reasoning coupling and off-policy distribution mismatch. Overall, TGRL extends group-based multimodal RL optimization from outcome-level supervision to trajectory-level alignment, integrating expert guidance while preserving on-policy exploration.

7 Conclusion

In this work, we revisit the optimization flow of multimodal RLVR and identify the lack of supervision over the perception–reasoning transition as a critical bottleneck. To address this, we propose **T**rajectory-**G**uided **R**einforcement **L**earning (TGRL), which integrates expert trajectories into group-based RL optimization through token reweighting and trajectory filtering. By extending supervision from outcome-level rewards to trajectory-level alignment, TGRL enables more precise credit assignment while preserving on-policy exploration.

Limitations and Future Work. Future work includes reducing reliance on strong expert policies through self-generated trajectories and improving the stability of off-policy trajectory alignment. Extending trajectory-level supervision to broader multimodal settings would further validate its generality and advance more reliable perception–reasoning integration in multimodal foundation models, particularly for embodied intelligence and real-world reasoning tasks.

Acknowledgement

This research is supported by the National Natural Science Foundation of China (62572449).

References

1. Chen, H., Tu, H., Wang, F., Liu, H., Tang, X., Du, X., Zhou, Y., Xie, C.: SFT or RL? an early investigation into training r1-like reasoning large vision-language models. Transactions on Machine Learning Research (2025), https://openreview.net/forum?id=wZI5qkQeDF
2. Comanici, G., Bieber, E., Schaekermann, M., Pasupat, I., Sachdeva, N., Dhillon, I., Blistein, M., Ram, O., Zhang, D., Rosen, E., et al.: Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. arXiv preprint arXiv:2507.06261 (2025)
3. Deng, Y., Bansal, H., Yin, F., Peng, N., Wang, W., Chang, K.W.: OpenVLThinker: Complex vision-language reasoning via iterative SFT-RL cycles. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems (2025), https://openreview.net/forum?id=gfX1nqBKtu
4. Guan, T., Liu, F., Wu, X., Xian, R., Li, Z., Liu, X., Wang, X., Chen, L., Huang, F., Yacoob, Y., 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. pp. 14375–14385 (2024)
5. Guo, D., Yang, D., Zhang, H., Song, J., Wang, P., Zhu, Q., Xu, R., Zhang, R., Ma, S., Bi, X., et al.: Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. Nature **645**(8081), 633–638 (2025)
6. Huang, W., Jia, B., Cao, S., Ye, Z., zhao, F., Xu, Z., Hu, Y., Lin, S.: Vision-r1: Incentivizing reasoning capability in multimodal large language models. In: The Fourteenth International Conference on Learning Representations (2026), https://openreview.net/forum?id=UZIjskfbfU
7. Jaech, A., Kalai, A., Lerer, A., Richardson, A., El-Kishky, A., Low, A., Helyar, A., Madry, A., Beutel, A., Carney, A., et al.: Openai o1 system card. arXiv preprint arXiv:2412.16720 (2024)
8. Lambert, N., Morrison, J., Pyatkin, V., Huang, S., Ivison, H., Brahman, F., Miranda, L.J.V., Liu, A., Dziri, N., Lyu, X., Gu, Y., Malik, S., Graf, V., Hwang, J.D., Yang, J., Bras, R.L., Tafjord, O., Wilhelm, C., Soldaini, L., Smith, N.A., Wang, Y., Dasigi, P., Hajishirzi, H.: Tulu 3: Pushing frontiers in open language model post-training. In: Second Conference on Language Modeling (2025), https://openreview.net/forum?id=i1uGbfHHpH
9. Li, J., Fang, J., Lu, J., Wang, Y., Guo, X., Zhang, T., Wang, X., He, X.: Enhancing multi-modal LLMs reasoning via difficulty-aware group normalization. In: Forty-third International Conference on Machine Learning (2026), https://openreview.net/forum?id=jyOgpu5wfC
10. Li, J., Gao, Y., Lu, J., Fang, J., Wen, C., Lin, H., Wang, X.: Diffgad: A diffusion-based unsupervised graph anomaly detector. In: International Conference on Learning Representations. vol. 2025, pp. 37744–37766 (2025)
11. Liu, A., Mei, A., Lin, B., Xue, B., Wang, B., Xu, B., Wu, B., Zhang, B., Lin, C., Dong, C., et al.: Deepseek-v3. 2: Pushing the frontier of open large language models. arXiv preprint arXiv:2512.02556 (2025)
12. Liu, X., Ni, J., Wu, Z., Du, C., Dou, L., Wang, H., Pang, T., Shieh, M.Q.: Noisy-rollout: Reinforcing visual reasoning with data augmentation. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems (2025), https://openreview.net/forum?id=9zD2i7YRot

13. Liu, Z., Sun, Z., Zang, Y., Dong, X., Cao, Y., Duan, H., Lin, D., Wang, J.: Visual-rft: Visual reinforcement fine-tuning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 2034–2044 (October 2025)
14. Lu, J., Li, J., Gao, Y., Wu, J., Wu, J., Wang, X., He, X.: Advip: Aligning multimodal llms via adaptive vision-enhanced preference optimization. International Journal of Computer Vision **134**(5), 224 (2026)
15. Lu, J., Wang, S., Zhang, X., Hao, Y., He, X.: Semantic-based selection, synthesis, and supervision for few-shot learning. In: Proceedings of the 31st ACM International Conference on Multimedia. pp. 3569–3578 (2023)
16. Lu, J., Wu, J., Li, J., Huang, K., Yang, S., Wang, G., Wu, J., Wang, X., He, X.: Bridging perception and reasoning: Token reweighting for rlvr in multimodal llms. arXiv preprint arXiv:2603.25077 (2026)
17. Lu, J., Wu, J., Li, J., Jia, X., Wang, S., Zhang, Y., Fang, J., Wang, X., He, X.: Dama: Data-and model-aware alignment of multi-modal llms. arXiv preprint arXiv:2502.01943 (2025)
18. Lu, K., Lab, T.M.: On-policy distillation. Thinking Machines Lab: Connectionism (2025). https://doi.org/10.64434/tml.20251026, https://thinkingmachines.ai/blog/on-policy-distillation
19. Lu, P., Bansal, H., Xia, T., Liu, J., Li, C., Hajishirzi, H., Cheng, H., Chang, K.W., Galley, M., Gao, J.: Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In: The Twelfth International Conference on Learning Representations (2024), https://openreview.net/forum?id=KUNzEQMWU7
20. Lu, P., Gong, R., Jiang, S., Qiu, L., Huang, S., Liang, X., Zhu, S.C.: Inter-gps: Interpretable geometry problem solving with formal language and symbolic reasoning. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 6774–6786 (2021)
21. Peng, Y., Wang, P., Wang, X., Wei, Y., Pei, J., Qiu, W., Jian, A., Hao, Y., Pan, J., Xie, T., et al.: Skywork r1v: Pioneering multimodal reasoning with chain-of-thought. arXiv preprint arXiv:2504.05599 (2025)
22. Qiao, R., Tan, Q., Dong, G., MinhuiWu, M., Sun, C., Song, X., Wang, J., GongQue, Z., Lei, S., Zhang, Y., et al.: We-math: Does your large multimodal model achieve human-like mathematical reasoning? In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 20023–20070 (2025)
23. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347 (2017)
24. Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y., Wu, Y., et al.: Deepseekmath: Pushing the limits of mathematical reasoning in open language models. arXiv preprint arXiv:2402.03300 (2024)
25. Wang, H., Qu, C., Huang, Z., Chu, W., Lin, F., Chen, W.: VL-rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems (2025), https://openreview.net/forum?id=4oYxzssbVg
26. Wang, K., Pan, J., Shi, W., Lu, Z., Ren, H., Zhou, A., Zhan, M., Li, H.: Measuring multimodal mathematical reasoning with math-vision dataset. Advances in Neural Information Processing Systems **37**, 95095–95169 (2024)
27. Wang, P., Wei, Y., Peng, Y., Wang, X., Qiu, W., Shen, W., Xie, T., Pei, J., Zhang, J., Hao, Y., et al.: Skywork r1v2: Multimodal hybrid reinforcement learning for reasoning. arXiv preprint arXiv:2504.16656 (2025)

28. Wang, X., Yang, Z., Feng, C., Lu, H., Li, L., Lin, C.C., Lin, K., Huang, F., Wang, L.: SoTA with less: MCTS-guided sample selection for data-efficient visual reasoning self-improvement. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems (2025), https://openreview.net/forum?id=PHu9xJeAum
29. Wang, Z., Guo, X., Stoica, S., Xu, H., WANG, H., Ha, H., Chen, X., Chen, Y., Yan, M., Huang, F., Ji, H.: Perception-aware policy optimization for multimodal reasoning. In: The Fourteenth International Conference on Learning Representations (2026), https://openreview.net/forum?id=izbBqTL8vb
30. Wei, L., Li, Y., Zheng, K., Wang, C., Wang, Y., Kong, L., Sun, L., Huang, W.: Advancing multimodal reasoning via reinforcement learning with cold start. arXiv preprint arXiv:2505.22334 (2025)
31. Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine learning **8**(3), 229–256 (1992)
32. Xiao, T., Xu, X., Huang, Z., Gao, H., Liu, Q., Liu, Q., Chen, E.: Perception-r1: Advancing multimodal reasoning capabilities of MLLMs via visual perception reward. In: The Fourteenth International Conference on Learning Representations (2026), https://openreview.net/forum?id=KttCXdjj4w
33. Yan, J., Li, Y., Hu, Z., Wang, Z., Cui, G., Qu, X., Cheng, Y., Zhang, Y.: Learning to reason under off-policy guidance. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems (2025), https://openreview.net/forum?id=vO8LLoNWWk
34. Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al.: Qwen3 technical report. arXiv preprint arXiv:2505.09388 (2025)
35. Yang, Y., He, X., Pan, H., Jiang, X., Deng, Y., Yang, X., Lu, H., Yin, D., Rao, F., Zhu, M., Zhang, B., Chen, W.: R1-onevision: Advancing generalized multimodal reasoning through cross-modal formalization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 2376–2385 (October 2025)
36. Yu, E., Lin, K., Zhao, L., jisheng yin, Wei, Y., Peng, Y., Wei, H., Sun, J., Han, C., Ge, Z., Zhang, X., Jiang, D., Wang, J., Tao, W.: Perception-r1: Pioneering perception policy with reinforcement learning. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems (2025), https://openreview.net/forum?id=BeXcXrXetA
37. Yu, Q., Zhang, Z., Zhu, R., Yuan, Y., Zuo, X., YuYue, Dai, W., Fan, T., Liu, G., Liu, J., Liu, L., Liu, X., Lin, H., Lin, Z., Ma, B., Sheng, G., Tong, Y., Zhang, C., Zhang, M., Zhang, R., Zhang, W., Zhu, H., Zhu, J., Chen, J., Chen, J., Wang, C., Yu, H., Song, Y., Wei, X., Zhou, H., Liu, J., Ma, W.Y., Zhang, Y.Q., Yan, L., Wu, Y., Wang, M.: DAPO: An open-source LLM reinforcement learning system at scale. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems (2025), https://openreview.net/forum?id=2a36EMSSTp
38. Zhan, R., Li, Y., Wang, Z., Qu, X., Liu, D., Shao, J., Wong, D.F., Cheng, Y.: ExGRPO: Learning to reason from prior successes. In: The Fourteenth International Conference on Learning Representations (2026), https://openreview.net/forum?id=701tjQXWVk
39. Zhang, H., Fu, J., Zhang, J., Fu, K., Wang, Q., Zhang, F., Zhou, G.: Rlep: Reinforcement learning with experience replay for llm reasoning (2025), https://arxiv.org/abs/2507.07451
40. Zhang, R., Jiang, D., Zhang, Y., Lin, H., Guo, Z., Qiu, P., Zhou, A., Lu, P., Chang, K.W., Qiao, Y., et al.: Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? In: European Conference on Computer Vision. pp. 169–186. Springer (2024)

41. Zhang, Y., Lu, X., Yin, S., Fu, C., Chen, W., Hu, X., Wen, B., Jiang, K., Liu, C., Zhang, T., fan, H., Chen, K., Chen, J., Ding, H., Tang, K., Zhang, Z., Wang, L., Yang, F., Gao, T., Zhou, G.: Thyme: Think beyond images. In: The Fourteenth International Conference on Learning Representations (2026), https://openreview.net/forum?id=gCWLkqK450
42. Zheng, Z., Yang, M., Hong, J., Zhao, C., Xu, G., Yang, L., Shen, C., XingYu: Deepeyes: Incentivizing "thinking with images" via reinforcement learning. In: The Fourteenth International Conference on Learning Representations (2026), https://openreview.net/forum?id=xUyMXkI958