EXTENDED ABSTRACT: Learning to Think Fast and Slow for Visual Language Models

1. Introduction

Human cognition is widely recognized to operate through two thinking systems: System 1 and System 2 [5, 6, 8]. System 1 enables fast, automatic responses to simple scenarios, while System 2 engages in slow, deliberate reasoning for intricate challenges. Such adaptive allocation of reasoning effort offers valuable inspiration for designing more cognitively-aligned visual language models (VLMs).

Current research on visual reasoning models primarily emphasizes step-by-step reasoning [3, 4, 7, 20, 23, 26–31], encouraging behaviors such as detailed image description or reflective reasoning to elongate their reasoning chains. However, existing approaches ignore the dual-mode thinking mechanism, causing excessive reasoning on all queries and leading to redundant token usage. While such step-by-step reasoning benefits challenging problems, it incurs unnecessary computational overhead on simpler ones.

This limitation raises a natural question: can VLMs adaptively switch between fast and slow thinking as humans do? To explore this possibility, we first examine the response length patterns of pre-trained VLMs. As shown in Fig. 1, different VLMs exhibit generally consistent response lengths within the same task while varying across tasks, suggesting that an implicit prior on response length emerges during large-scale pre-training. However, most existing reasoning methods override this prior by uniformly encouraging extended reasoning. As a consequence, these methods lose adaptive thinking and rely on long output tokens to get the right answer. As evidenced in Fig. 2, reasoning models significantly underperform the base model under limited token budgets and only start to show advantages when the token budget becomes sufficiently large.

In this work, we introduce **DualMindVLM**, a dual-mode thinking VLM that leverages the model's intrinsic prior on response length to develop two controllable thinking modes, enabling automatic switching between fast and slow thinking. The approach consists of two stages. In the first stage, each training instance is anchored to a thinking prefix following the model's natural response length tendency. In the second stage, we employ GRPO with partially-constrained rollouts where half of the trajectories are generated with a thinking mode prefix while the other half are freely generated. This design reinforces both thinking modes while allowing the model to autonomously select the appropriate prefix at inference time.

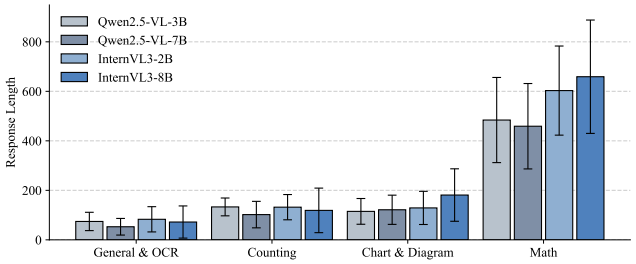


Figure 1. **Average response lengths of four pre-trained general-purpose VLMs across a variety of VQA tasks.** Response lengths remain relatively consistent within the same task while varying across tasks, suggesting the presence of an implicit response-length prior inherited from large-scale pre-training.

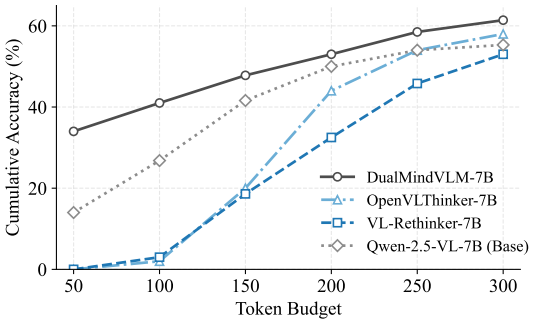


Figure 2. Cumulative accuracy under varying token budgets on MMStar [2].

In summary, our main contributions are threefold: 1) We identify an implicit prior on response length in pre-trained VLMs and show that it can be leveraged to develop an explicit dual-mode thinking mechanism. 2) We propose a two-stage training framework that stabilizes and strengthens both thinking modes while enabling automatic mode selection at inference time. 3) Extensive experiments on six multimodal benchmarks demonstrate the effectiveness of DualMindVLM.

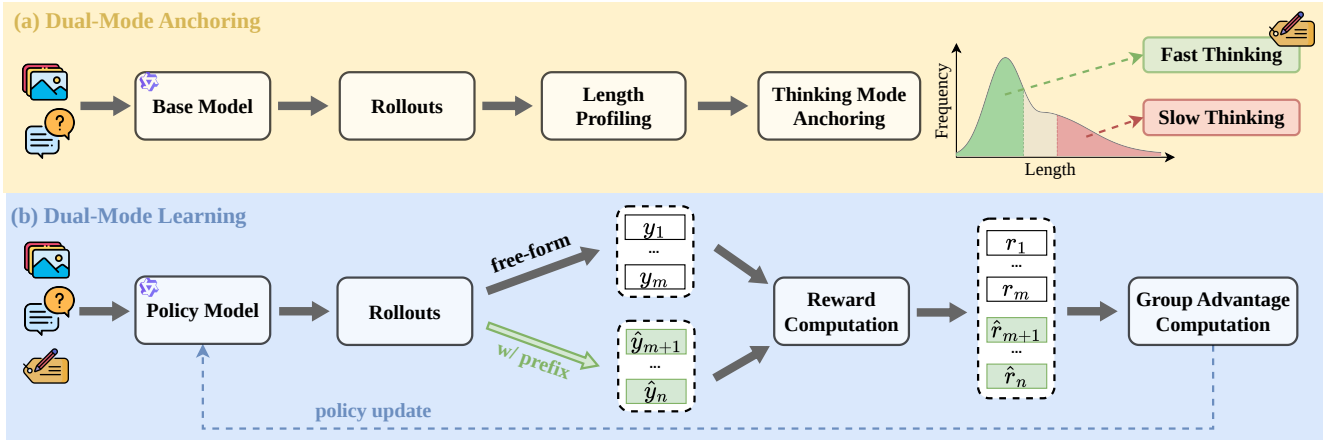


Figure 3. **Overview of DualMindVLM**. (a) For each VQA pair, we collect multiple rollouts from the base model and profile their response lengths. Based on the observed length tendency, we anchor the pair to a thinking mode by assigning the corresponding control prefix. (b) We then perform GRPO with partially constrained rollouts, where one group of candidates is generated using the assigned prefix, while the other is freely generated. A group-wise advantage is computed over all candidate responses to reinforce dual-mode reasoning behaviors and consistent prefix usage.

2. Methodology

We propose DualMindVLM, a dual-mode thinking model that is trained using RL and simple visual question-answer pairs. As shown in Fig. 3, the training pipeline consists of two stages. The first stage, *dual-mode anchoring*, maps the model's implicit length prior to two thinking modes by partitioning the training data into two subsets. Instances associated with short responses are used to develop fast thinking, while those exhibiting long responses are used to develop slow thinking. Each thinking mode is associated with a control prefix that triggers the corresponding reasoning behavior. The second stage, *dual-mode learning*, builds upon the anchored thinking modes to optimize reasoning in each mode and enable autonomous mode selection. Specifically, we employ GRPO with partially constrained rollouts, where one group is generated using the assigned control prefix, while the other is freely generated. Below we detail these two stages.

2.1. Dual-Mode Anchoring

Anchoring data to thinking modes. For each sample, we collect multiple rollouts (e.g., 8) from the base model and compute the average response length as a proxy for the model's length tendency. A sample is anchored to fast thinking if the average length is below τ_{fast}, or to slow thinking if it exceeds τ_{slow}. We set $\tau_{fast} = 100$ tokens and $\tau_{slow} = 200$ tokens by default. To mitigate vanishing advantages [18, 23], we exclude samples with an average accuracy of 0 or 1.

Binding thinking modes to prefixes. We bind each mode to a control prefix. p^{fast} = `"Short Thinking:"` is associated with concise processing, whereas p^{slow} = `"Long Thinking:"` triggers detailed reasoning. This prefix-to-mode binding turns anchored thinking modes into steerable behaviors.

2.2. Dual-Mode Learning

Hybrid group response sampling. Given an input $x = (I, Q)$, we sample n candidate responses from the sampling model $\pi_{\theta_{\text{old}}}$ and split them into two subgroups: a free-form subgroup $\{y_i\}_{i=1}^{m}$ and a prefix-conditioned subgroup $\{\hat{y}_i\}_{i=m+1}^{n}$. For each $\hat{y}_i$, we manually insert the anchored prefix p^* before generation to enforce the desired output format. Below we use the notation y for clarity.

Because the anchored prefix is aligned with the model's inherent length tendency, the prefix-conditioned subgroup provides stable behavioral anchors within each group. This helps the free-form subgroup learn the mapping between prefixes and reasoning behaviors through group-wise comparisons.

Reward design. Each candidate response is scored by a joint reward:

$$r(y_i) = r_a(y_i) + r_f(y_i) \tag{1}$$

This reward is used to optimize answer correctness while promoting mode consistency.

- Accuracy Reward (r_a): We set $r_a(y) = 1$ if the predicted answer is correct and 0 otherwise.
- Format Reward (r_f): To encourage prefix usage and

mode consistency, we define

$$r_f(y_i) = \begin{cases} 1, & \text{if } \texttt{prefix}(y_i) = p^*, \\ 0.5, & \text{if } \texttt{prefix}(y_i) \neq p^* \\ & \text{and } \quad \texttt{prefix}(y_i) \in \{p^{\text{fast}}, p^{\text{slow}}\}, \\ 0, & \text{otherwise}, \end{cases} \tag{2}$$

where $\texttt{prefix}(y_i)$ denotes the prefix extracted from the generated response.

Optimization objective. We optimize the policy model π_θ using the GRPO objective over the n candidates in each group. For each response y_i, we compute a relative advantage by subtracting the group mean reward:

$$A_i = r(y_i) - \text{mean}\big(r(y_1), r(y_2), \ldots, r(y_n)\big).$$

The GRPO objective is computed as:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \frac{1}{n}\sum_{i=1}^{n}\Bigg[\min\left(\frac{\pi_\theta(y_i \mid x)}{\pi_{\theta_{\text{old}}}(y_i \mid x)}A_i, \text{clip}\left(\frac{\pi_\theta(y_i \mid x)}{\pi_{\theta_{\text{old}}}(y_i \mid x)}, 1-\epsilon, 1+\epsilon\right)A_i\right) + \beta\mathcal{D}_{\text{KL}}(\pi_\theta \,\|\, \pi_{\text{ref}})\Bigg], \tag{3}$$

where ϵ and β are hyper-parameters.

3. Experiments

3.1. Experimental Setup

We train on 37,506 visual question-answer pairs collected from multiple public datasets covering general visual understanding [13, 19, 21], spatial reasoning [10], chart and document understanding [9, 12, 14, 16, 17], and mathematical reasoning [18, 23]. The anchored training set is nearly balanced, with 18,778 slow-thinking samples and 18,728 fast-thinking samples. We evaluate on six benchmarks: MathVista [15], MathVision [24], MMStar [2], MMBench [11], ScienceQA [13], and AI2D [9]. We use Qwen2.5-VL-7B [1] as the base model and train with TRL [22], using $n = 8$ completions, a learning rate of 1×10^{-6}, rollout batch size 256, KL coefficient 1×10^{-3}, and maximum generation length 2,048.

3.2. Main Results

Table 1 compares DualMindVLM with state-of-the-art visual reasoning models of similar sizes. Except for LLaVA-CoT and R1-VL, all compared models are based on Qwen2.5-VL. Overall, DualMindVLM achieves state-of-the-art performance while maintaining exceptionally high token efficiency. Compared with the base model Qwen2.5-VL, it improves accuracy on all benchmarks, with gains of +7.4% on MathVista, +5.1% on MathVision, +1.4% on MMStar, +5.3% on MMBench, +3.2% on ScienceQA, and +3.0% on AI2D, while also producing shorter outputs across all benchmarks. Compared with leading reasoning models, including VL-Rethinker [23], ThinkLite [25], MM-Eureka [18], OpenVLThinker [3], R1-VL [31], R1-Onevision [29], and LLaVA-CoT [28], DualMindVLM achieves the best accuracy on four out of six benchmarks and remains competitive on the other two, while reducing token usage by 40% on average relative to the strongest rival on each benchmark. These results demonstrate that DualMindVLM achieves a strong balance between accuracy and token efficiency.

3.3. Ablation Study

Table 2 shows the effect of the two key components in DualMindVLM: dual-mode anchoring and dual-mode RL. By removing dual-mode anchoring—meaning that we rely only on the dual-mode system prompt to develop the dual-thinking systems. It leads to consistent performance drops on both MathVista and MathVision, with accuracy decreasing from 75.6 to 72.6 and from 30.2 to 28.5, respectively. This variant also produces much shorter responses. We find that the model collapses to the fast thinking mode during training, which prevents the development of effective System 2 reasoning and leads to lower performance. Detailed statistics are omitted due to space constraints.

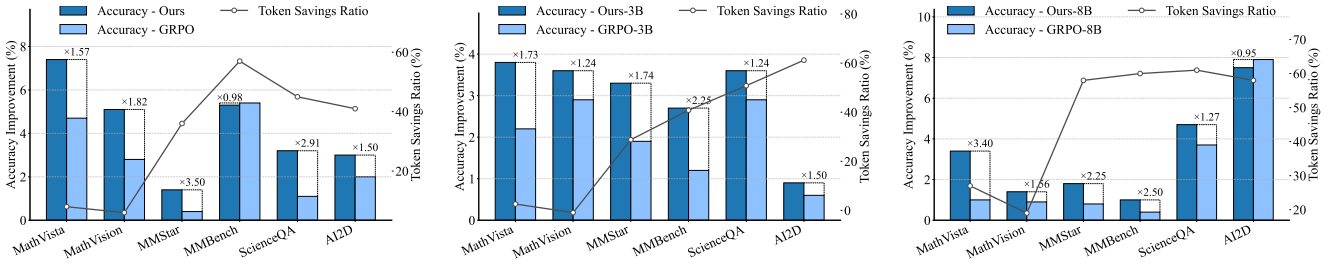
Removing dual-mode RL and instead training with standard GRPO using the prompt *"Please reason step by step"* also degrades performance, while substantially increasing response length. Compared with the full model, this variant drops from 75.6 to 75.0 on MathVista and from 30.2 to 28.9 on MathVision, while increasing output length from 184 to 271 and from 446 to 584 tokens, respectively. This suggests that part of the performance gains may already stem from the mode anchoring stage itself. Since anchoring relies on length-based thresholding, it effectively alters the training data distribution by selecting samples with more pronounced output length tendencies.

3.4. Further Analysis

Generalization. Fig. 4 compares DualMindVLM with GRPO on three backbones of different architectures and scales. The bar charts report the accuracy improvement over the corresponding base model together with the token savings ratio relative to GRPO. Across all three backbones, DualMindVLM consistently achieves higher accuracy than GRPO on most benchmarks while using fewer tokens on all benchmarks. These results show that the proposed dual-mode training strategy generalizes well beyond a single model configuration and consistently yields a better

Table 1. **Comparison of DualMindVLM with state-of-the-art visual reasoning models**. For each benchmark, we report accuracy (acc, %) and average response length (len, #tokens). The best result is highlighted in **bold**. DualMindVLM strikes the best balance between accuracy and token efficiency among all models.

Model	Size	Strategy	MathVista		MathVision		MMStar		MMBench		ScienceQA		AI2D		Avg	
			acc ↑	len ↓	acc ↑	len ↓	acc ↑	len ↓	acc ↑	len ↓	acc ↑	len ↓	acc ↑	len ↓	acc ↑	len ↓
Qwen2.5-VL	7B	-	68.2	205	25.1	511	63.9	155	83.0	73	84.0	156	80.8	145	67.5	208
LLaVA-CoT	11B	SFT	54.8	350	-	-	57.6	464	-	-	-	-	-	-	-	-
R1-Onevision	7B	SFT+RL	64.1	279	29.9	560	-	-	-	-	-	-	-	-	-	-
R1-VL	7B	SFT+RL	63.5	263	24.7	363	60.0	221	-	-	-	-	-	-	-	-
OpenVLThinker	7B	SFT+RL	72.3	242	25.9	**326**	63.3	200	87.5	177	82.2	171	83.2	160	69.1	213
MM-Eureka	7B	RL	73.0	252	26.9	612	64.1	246	87.3	159	83.5	202	83.5	207	69.7	280
ThinkLite	7B	RL	75.1	247	28.5	599	65.0	175	**88.7**	113	-	-	83.6	168	-	-
VL-Rethinker	7B	RL	74.9	268	**32.3**	566	64.9	231	87.6	201	85.5	205	82.4	226	71.3	283
DualMindVLM	7B	RL	**75.6**	**184**	30.2	446	**65.3**	**121**	88.3	**69**	**87.2**	**98**	**83.8**	**104**	**71.7**	**170**



(a) Qwen2.5-VL-7B (b) Qwen2.5-VL-3B (c) InternVL3-8B

Figure 4. **DualMindVLM vs. GRPO across different backbones.** We compare DualMindVLM with GRPO on three backbones of different architectures and scales. Each subfigure reports the accuracy improvement over the corresponding base model together with the token savings ratio relative to GRPO. Across all three backbones, DualMindVLM consistently achieves a better balance between accuracy and token efficiency, demonstrating that the proposed dual-mode training strategy generalizes beyond a single model configuration.

Table 2. Ablation study on key components of DualMindVLM.

Model	MathVista		MathVision	
	acc ↑	len ↓	acc ↑	len ↓
Qwen2.5-VL	68.2	205	25.1	511
DualMindVLM	75.6	184	30.2	446
w/o anchoring	72.6	120	28.5	332
w/o dual-mode RL	75.0	271	28.9	584

Table 3. **Sensitivity to length thresholds.** Comparison of performance and response length averaged over six benchmarks. "Fast" and "Slow" denote the average response length in fast and slow thinking, while "Mean" denotes the overall mean response length.

τ_{fast}	τ_{slow}	Accuracy	Length		
			Fast	Slow	Mean
None	None	68.9	72	131	98
100	200	70.1	23	275	164
50	200	69.8	23	302	197
100	250	69.8	29	290	173

balance between accuracy and token efficiency.

Threshold sensitivity. To investigate how the length thresholds (τ_{fast} and τ_{slow}) influence the model's behavior, we sample 5k fast-thinking and 5k slow-thinking examples from the training set according to different thresholds, and train a separate model for each configuration. The "None" configuration serves as a baseline, where fast- and slow-thinking samples are randomly selected.

As shown in Table 3, two key phenomena emerge. First, randomly assigning thinking modes without considering length-based priors significantly blurs the boundaries between the two thinking modes: the fast-thinking mode becomes notably longer (72 vs. 23 tokens) while the slow-thinking mode shrinks (131 vs. 275 tokens), resulting in a 1.2% drop in accuracy. Second, our length-based anchoring is robust to the specific choice of thresholds: varying τ_{fast} and τ_{slow} across rows 2–4 results in only marginal performance fluctuations ($\leq 0.3\%$). This indicates that anchoring thinking modes with length-based priors helps maintain a clear separation between the two behaviors.

References

[1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 3

[2] 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? *Advances in Neural Information Processing Systems*, 37:27056–27087, 2024. 1, 3

[3] Yihe Deng, Hritik Bansal, Fan Yin, Nanyun Peng, Wei Wang, and Kai-Wei Chang. Openvlthinker: An early exploration to complex vision-language reasoning via iterative self-improvement. *arXiv preprint arXiv:2503.17352*, 2025. 1, 3

[4] Yuhao Dong, Zuyan Liu, Hai-Long Sun, Jingkang Yang, Winston Hu, Yongming Rao, and Ziwei Liu. Insight-v: Exploring long-chain visual reasoning with multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9062–9072, 2025. 1

[5] Jonathan St BT Evans. Dual process theory: Perspectives and problems. *Dual process theory 2.0*, pages 137–155, 2017. 1

[6] Jonathan St BT Evans and Keith E Stanovich. Dual-process theories of higher cognition: Advancing the debate. *Perspectives on psychological science*, 8(3):223–241, 2013. 1

[7] Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models. *arXiv preprint arXiv:2503.06749*, 2025. 1

[8] Daniel Kahneman. *Thinking, fast and slow*. macmillan, 2011. 1

[9] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *European conference on computer vision*, pages 235–251. Springer, 2016. 3

[10] Adam Dahlgren Lindström and Savitha Sam Abraham. Clevr-math: A dataset for compositional language, visual and mathematical reasoning. *arXiv preprint arXiv:2208.05358*, 2022. 3

[11] 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. 3

[12] Pan Lu, Liang Qiu, Jiaqi Chen, Tony Xia, Yizhou Zhao, Wei Zhang, Zhou Yu, Xiaodan Liang, and Song-Chun Zhu. Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning. *arXiv preprint arXiv:2110.13214*, 2021. 3

[13] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521, 2022. 3

[14] Pan Lu, Liang Qiu, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, Tanmay Rajpurohit, Peter Clark, and Ashwin Kalyan. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. *arXiv preprint arXiv:2209.14610*, 2022. 3

[15] 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. 3

[16] Ahmed Masry, Xuan Long Do, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the association for computational linguistics: ACL 2022*, pages 2263–2279, 2022. 3

[17] Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2200–2209, 2021. 3

[18] Fanqing Meng, Lingxiao Du, Zongkai Liu, Zhixiang Zhou, Quanfeng Lu, Daocheng Fu, Botian Shi, Wenhai Wang, Junjun He, Kaipeng Zhang, et al. Mm-eureka: Exploring visual aha moment with rule-based large-scale reinforcement learning. *CoRR*, 2025. 2, 3

[19] Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. In *European conference on computer vision*, pages 146–162. Springer, 2022. 3

[20] Haozhan Shen, Peng Liu, Jingcheng Li, Chunxin Fang, Yibo Ma, Jiajia Liao, Qiaoli Shen, Zilun Zhang, Kangjia Zhao, Qianqian Zhang, et al. Vlm-r1: A stable and generalizable r1-style large vision-language model. *arXiv preprint arXiv:2504.07615*, 2025. 1

[21] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8317–8326, 2019. 3

[22] Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. Trl: Transformer reinforcement learning. `https://github.com/huggingface/trl`, 2020. 3

[23] Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhu Chen. Vl-rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. *arXiv preprint arXiv:2504.08837*, 2025. 1, 2, 3

[24] Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. Measuring multimodal mathematical reasoning with math-vision dataset. *Advances in Neural Information Processing Systems*, 37:95095–95169, 2024. 3

[25] Xiyao Wang, Zhengyuan Yang, Chao Feng, Hongjin Lu, Linjie Li, Chung-Ching Lin, Kevin Lin, Furong Huang, and Lijuan Wang. Sota with less: Mcts-guided sample selection for data-efficient visual reasoning self-improvement. *arXiv preprint arXiv:2504.07934*, 2025. 3

[26] Jiaer Xia, Bingkui Tong, Yuhang Zang, Rui Shao, and Kaiyang Zhou. Bootstrapping grounded chain-of-thought in multimodal llms for data-efficient model adaptation. In *ICCV*, 2025. 1

[27] Jiaer Xia, Yuhang Zang, Peng Gao, Yixuan Li, and Kaiyang Zhou. Visionary-r1: Mitigating shortcuts in visual reasoning with reinforcement learning. *arXiv preprint arXiv:2505.14677*, 2025.

[28] Guowei Xu, Peng Jin, Ziang Wu, Hao Li, Yibing Song, Lichao Sun, and Li Yuan. Llava-cot: Let vision language models reason step-by-step. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2087–2098, 2025. 3

[29] Yi Yang, Xiaoxuan He, Hongkun Pan, Xiyan Jiang, Yan Deng, Xingtao Yang, Haoyu Lu, Dacheng Yin, Fengyun Rao, Minfeng Zhu, et al. R1-onevision: Advancing generalized multimodal reasoning through cross-modal formalization. *arXiv preprint arXiv:2503.10615*, 2025. 3

[30] 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.

[31] Jingyi Zhang, Jiaxing Huang, Huanjin Yao, Shunyu Liu, Xikun Zhang, Shijian Lu, and Dacheng Tao. R1-vl: Learning to reason with multimodal large language models via step-wise group relative policy optimization. *arXiv preprint arXiv:2503.12937*, 2025. 1, 3