Explainable Chain-of-Thought Object Counting in Vision-Language Models using Reinforcement Learning

E. Zhixuan Zeng, Saeejith Nair, Junfeng Lei
University of Waterloo
{ezzeng, smnair, junf137}@uwaterloo.ca

Abstract

Counting objects in images remains a challenging task for vision-language models, particularly when multiple instances are densely packed or partially occluded. This report proposes a novel framework for explainable object counting, leveraging Qwen-2.5-VL and fine-tuning it via Low-Rank Adaptation (LoRA) under a Group Relative Policy Optimization (GRPO) reinforcement learning scheme. Rather than providing a single numeric output, our approach produces a structured chain-of-thought, explicitly pointing to each counted object via centroid coordinates. We construct an augmented version of the TallyQA dataset focusing on simple counting questions, enriched with object centroids. Notably, we apply GRPO directly to the base model without supervised pretraining, demonstrating the effectiveness of a cold-start approach. Our multi-component reward system balances format adherence, numeric accuracy, and spatial precision, achieving 67.94% counting accuracy and 92.62% pointing accuracy—significantly outperforming both the baseline (34.84%/2.73%) and supervised fine-tuning (59.93%/86.89%). Through ablation studies, we demonstrate that single-reward configurations often lead to reward exploitation, while combined rewards produce balanced, interpretable outputs. Our findings highlight the critical role of prompt engineering and reward design in developing transparent, verifiable counting systems for vision-language models.

1. Introduction

Vision-language models (VLMs) have demonstrated impressive progress in tasks such as image captioning and question answering. However, object counting remains a surprisingly persistent challenge for these models, which often underperform when the number of instances grows beyond training distributions or becomes visually complex [12, 13]. Worse, standard counting approaches yield only a numeric prediction, making it difficult to trace *why* a particular count was chosen. This lack of transparency can obscure model mistakes and hinder user trust.

Recent works emphasize the importance of chain-of-thought reasoning in large language models [3, 4], showing that step-by-step rationales can improve both interpretability and generalization. In the *vision* domain, chain-of-thought translates naturally to pointing-based explanations, where the model enumerates each counted object with a visual pointer [6]. Yet, purely supervised methods for pointing (or bounding-box supervision) are expensive to develop and sometimes fail to generalize beyond the annotated categories.

To address these challenges, we propose an **RL-driven, chain-of-thought counting framework** built on top of Qwen-2.5-VL [2], a large multimodal foundation model. We aim to:

- Provide a *transparent* reasoning trace, indicating each counted object via centroid-based pointing.
- Improve both counting accuracy and the *faithfulness* of the intermediate enumerations through reward shaping under *Group Relative Policy Optimization (GRPO)*.
- Use *Low-Rank Adaptation (LoRA)* [8] to ensure parameter efficiency in our fine-tuning process.

Our experiments focus on a new *augmented* version of TallyQA [1], containing centroid annotations derived from bounding boxes in COCO and Visual Genome images. We demonstrate that reinforcement learning (with carefully designed reward functions) can effectively refine chain-of-thought generation for counting, leading to more interpretable outputs and improved performance.

2. Related Work

2.1. Object Counting in Vision-Language Models

Counting remains difficult for modern VLMs, especially under distribution shift and visual clutter. Paiss et al. [12] mitigate CLIP-based counting failures via specialized contrastive objectives, while Qharabagh et al. [13] propose a hierarchical divide-and-conquer pipeline. These methods improve numeric accuracy but still output a single scalar without an explicit reasoning trace.

2.2. Chain-of-Thought and Visual Pointing

Chain-of-thought (CoT) has been shown to improve interpretability and generalization in language models [3, 4]. In vision, Deitke et al. [6] extends this idea to pointing-based explanations, where the model marks object locations as part of its rationale. However, such methods typically rely on large supervised datasets with bounding boxes or cen-

troids. We instead use reinforcement learning to encourage both correct counts and faithful visual pointing, reducing reliance on dense supervised labels.

2.3. Reinforcement Learning for Vision-Language Tasks

Reinforcement learning has recently emerged as an alternative to supervised annotation-heavy approaches for visual reasoning [3, 7]. Group Relative Policy Optimization (GRPO) [15, 17] improves upon PPO [14] by ranking samples within a group and optimizing relative advantages, avoiding a separate value network and lowering memory costs.

2.4. Parameter-Efficient Fine-Tuning via LoRA

Scaling RL on large VLMs is constrained by memory and compute. Low-Rank Adaptation (LoRA) [8] reduces the number of trainable parameters by inserting small rank-decomposed matrices while freezing base weights, and has been shown to be effective for vision-language specialization [11].

3. Methodology

We use Qwen-2.5-VL, a 3B vision-language model with an image encoder and text decoder [16]. We fine-tune it on our augmented TallyQA dataset with Group Relative Policy Optimization (GRPO) to learn chain-of-thought object counting with centroid-based pointing [5, 15]. To keep training efficient, we apply Low-Rank Adaptation (LoRA) and update only small rank-limited adapter matrices while freezing the base model weights [8].

3.1. Augmented TallyQA for Point-Based Counting

We use the simple split of TallyQA [1] and augment each image-question pair with centroid coordinates extracted from COCO and Visual Genome bounding boxes. Prompts are formatted to require chain-of-thought reasoning in `<think>` tags with explicit coordinate listings, followed by a final count in `<answer>` tags. We train on 1,000 samples and evaluate on 100.

3.2. Group Relative Policy Optimization (GRPO)

We fine-tune using GRPO [5, 15], where G chain-of-thought completions are sampled per prompt, each scored with a scalar reward r_i. The relative advantage is computed within the group:

$$A_i = r_i - \frac{1}{G}\sum_{j=1}^{G} r_j.$$

The policy is updated by maximizing

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{a_i \sim \pi_{\theta_{\text{old}}}}\left[\frac{1}{G}\sum_{i=1}^{G}\min\Big(\rho_i A_i, \text{clip}(\rho_i, 1-\epsilon, 1+\epsilon)\, A_i\Big)\right] - \beta D_{\text{KL}}\big(\pi_\theta \,\|\, \pi_{\text{ref}}\big) \quad (1)$$

where ρ_i is the importance ratio. Unlike PPO, GRPO avoids a learned critic and relies only on relative ranking, reducing memory cost while improving training stability.

3.3. Reward Design

We combine six reward terms, each scaled to $[0, 1]$:

$$\begin{aligned} R_{\text{final}} = & R_{\text{format}} + R_{\text{num\_points}} \\ & + R_{\text{points\_accuracy}} + R_{\text{count\_consistency}} \\ & + R_{\text{count\_accuracy}} + R_{\text{on\_segmentation}} \end{aligned} \quad (2)$$

R_{format} checks proper use of `<think>` and `<answer>` tags and structured centroid listings.

$R_{\text{num\_points}}$ encourages predicting the correct number of centroids:

$$R_{\text{num\_points}} = 1 - \min\left(1,\ \frac{|\hat{N} - N|}{N}\right),$$

where $\hat{N}$ is the number of predicted centroids.

$R_{\text{points\_accuracy}}$ promotes spatial alignment via optimal one-to-one matching (Hungarian algorithm). Matched pairs (p_i, g_i) contribute

$$\max\left(0,\ 1 - \frac{d(p_i, g_i)}{800}\right),$$

and the reward is their average; if either set is missing, it is set to zero.

$R_{\text{count\_consistency}}$ checks whether the final numeric answer matches the number of listed centroids.

$R_{\text{count\_accuracy}}$ gives credit only if the final answer equals the ground-truth count.

$R_{\text{on\_segmentation}}$ rewards predicted points that fall inside ground-truth segmentation masks.

Combined, these terms encourage both accurate counting and faithful, interpretable object localization.

3.4. Training Procedure

We train the model directly with GRPO, without any supervised warmup, updating only LoRA adapters. GRPO is run for two epochs on 1,000 samples with group size $G = 2$. The model generates two completions per prompt, receives R_{final}, and updates accordingly. Outputs follow a structured chain-of-thought format, e.g.:

```
<think>I spot a cow at <232,414>...
 Therefore, I see 3 cows.</think>
      <answer>3</answer>
```

4. Experiments and Results

We evaluate on the augmented TallyQA test split using two metrics: (1) counting accuracy, based on the final value in `<answer>`, and (2) point accuracy, measuring spatial alignment between predicted and ground-truth centroids.

4.1. Baseline

The unmodified Qwen-2.5-VL achieves **34.84%** counting accuracy and **2.73%** point accuracy. It often produces plain-text answers but rarely follows the required structured `<think>` and `<answer>` format, preventing reliable extraction of both counts and locations.

4.2. Single-Reward vs. Multi-Reward Ablations

We compare single-reward and combined-reward training. Single rewards lead to specialization and reward exploitation: counting-only rewards produce correct totals but no centroids, while segmentation or spatial rewards yield high point accuracy but near-zero counting performance by placing multiple points on the same instance. In contrast, combining complementary rewards promotes both correct counting and faithful centroid enumeration. The best configuration (format + consistency + count + spatial) reaches **68.64%** counting accuracy and **92.15%** point accuracy, significantly outperforming all single-reward setups (Table 1).

Table 1. Reward ablations using symbolic reward notation. Results reported as **counting / point accuracy (%)**.

Reward Setup	Count	Point
$R_{\text{count\_accuracy}}$ only	65.51	0.00
$R_{\text{count\_consistency}}$ only	65.51	85.51
$R_{\text{on\_segmentation}}$ only	4.18	52.09
$R_{\text{points\_accuracy}}$ only	0.69	93.83
$R_{\text{num\_points}}$ only	67.24	80.87
R_{format} only	50.52	76.92
$R_{\text{format}} + R_{\text{on\_segmentation}}$	1.05	63.39
$R_{\text{format}} + R_{\text{num\_points}} + R_{\text{points\_accuracy}} + R_{\text{count\_consistency}} + R_{\text{count\_accuracy}}$	**68.64**	92.15
$R_{\text{format}} + R_{\text{num\_points}} + R_{\text{points\_accuracy}} + R_{\text{count\_consistency}} + R_{\text{count\_accuracy}} + R_{\text{on\_segmentation}}$	67.94	**92.62**

4.3. Supervised Fine-Tuning vs. GRPO

Supervised fine-tuning (SFT) achieves 59.93% counting accuracy and 86.89% point accuracy. GRPO with the combined reward improves both to 67.94% and 92.62%, respectively, demonstrating that reinforcement learning refines both numeric correctness and spatial reasoning beyond what supervised training provides.

5. Discussion

Our results highlight that explainable counting requires more than numeric accuracy. The model must reason about visual instances, express that reasoning in a structured format, and align with imperfect human annotations. We discuss how prompt design, reward interactions, and dataset ambiguities shape this behavior.

5.1. Prompt Engineering for Chain-of-Thought Counting

Prompt structure strongly affects both counting accuracy and category focus. When the question appears before the reasoning instructions, the model often latches onto salient but irrelevant objects (e.g., people instead of baseball bats):

```
How many baseball bats are there? First
output the thinking process in <think>
</think> tags and then output the final
answer in <answer> </answer> tags...

Response:
<think>
I see a person at <603, 223>.
I spot a person at <284, 150>.
Therefore, I see 2 people.
</think>
<answer>2</answer>
```

Placing the question *after* formatting guidance and examples reduces this confusion, encouraging correct category-focused enumeration with consistent coordinate formatting. This mirrors recent findings on instruction ordering in vision-language prompting [4, 10].

```
First, count each instance you spot and
their <x, y> coordinates as part of the
thinking process in <think> </think> tags.
Then output the final answer as a single
number in <answer> </answer> tags.
Below are some examples:

<think>I spot a person at <401, 105>.
I spot a person at <48, 19>.
Therefore, I see 2 people.</think>
<answer>2</answer>

...more examples

{Question}.
```

5.2. Impact of Reward Functions

Formatting rewards act as structural scaffolding, accelerating convergence when combined with segmentation or spatial rewards (Figures 1 and 2). Without formatting, models often optimize for segmentation or spatial precision but neglect counting or category focus.

Single rewards cause specialization (e.g., correct totals without pointing, or high spatial accuracy but incorrect counts), while combined rewards yield complementary behavior, producing both accurate counts and faithful centroid annotations. This supports prior findings that multimodal reasoning benefits from balancing structural, process-level, and final-answer rewards [15, 16].

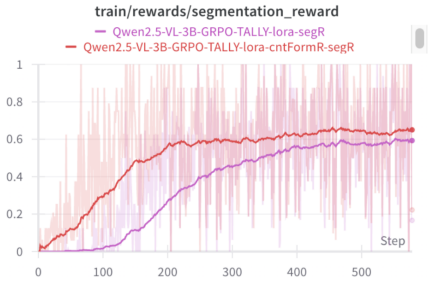

Figure 1. Segmentation reward optimization with (red) vs. without (purple) formatting reward. Formatting accelerates convergence.

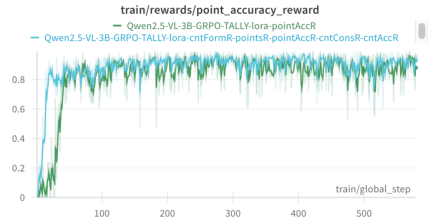

Figure 2. Point accuracy improvement with (blue) vs. without (dark green) formatting reward. Structured outputs speed up learning.

5.3. Dataset Annotation Challenges

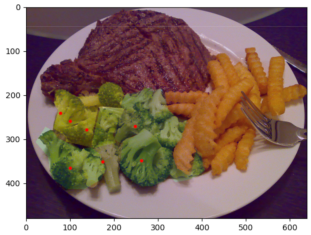

Figure 3. COCO merges multiple broccoli florets into one instance, causing undercounting.

Instance-level annotations in COCO and Visual Genome are often inconsistent for clustered or overlapping objects, as shown in Figure 3. A single mask may represent multiple countable items, making counting dependent on annotation granularity rather than visual distinctness. This affects both training and evaluation: models may be penalized even when correctly identifying separate objects. Similar ambiguity appears in grapes, crowds, or small repetitive objects, suggesting the need for clearer granularity standards in counting datasets [9].

5.4. Failure Cases and Model Limitations

In dense scenes, the model frequently produces linear centroid placements rather than identifying individual instances

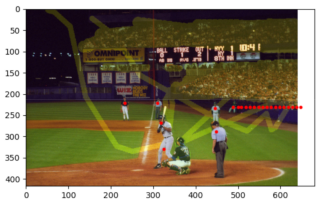

Figure 4. In crowded scenes, the model often places points in a line instead of detecting individuals.

(Figure 4). This behavior likely stems from:

- **Sparse high-count training samples**: Most training examples contain fewer than 10 objects.
- **Reward saturation**: Adding more points yields diminishing reward benefits, leading to collapsed predictions.
- **Annotation ambiguity**: Crowds are often annotated as coarse regions rather than distinct individuals.

Small or partially occluded objects are also commonly omitted. These limitations indicate that explainable chain-of-thought improves transparency but does not replace the need for robust instance localization, especially under occlusion, density, or segmentation ambiguity.

5.5. Future Directions

Future work includes analyzing performance across training steps by saving intermediate GRPO checkpoints, and studying whether warm-starting from SFT improves convergence stability over our cold-start approach. Extending training to more high-count and occluded scenarios could further improve robustness.

6. Conclusion

Our work introduces a novel RL-based approach for explainable object counting in vision-language models that combines chain-of-thought reasoning with point-based supervision, achieving 67.94% counting accuracy and 92.62% pointing accuracy—significantly outperforming both baseline and supervised fine-tuning approaches. Ablation studies revealed that our multi-component reward system balancing format adherence, numeric accuracy, and spatial precision effectively prevents reward exploitation while promoting transparent reasoning traces. While the model still struggles with densely packed objects and annotation ambiguities, future work could expand training data distribution and develop more sophisticated reward functions for handling challenging scenarios. Beyond object counting, our approach demonstrates that combining structured explanations with multi-objective reinforcement learning creates models that not only make accurate predictions but articulate their decision processes transparently, with potential applications across diverse visual reasoning tasks.

References

[1] Manoj Acharya, Kushal Kafle, and Christopher Kanan. Tallyqa: Answering complex counting questions, 2018. 1, 2

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025. 1

[3] Liang Chen, Lei Li, Haozhe Zhao, Yifan Song, Vinci, Zihao Yue, Lingpeng Kong, Qi Liu, and Baobao Chang. Rlvr in vision language models: Findings, questions and directions, 2025. Accessed: 2025-04-14. 1, 2

[4] Yangyi Chen, Karan Sikka, Michael Cogswell, Heng Ji, and Ajay Divakaran. Measuring and improving chain-of-thought reasoning in vision-language models, 2024. 1, 3

[5] DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. 2

[6] Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, Jiasen Lu, Taira Anderson, Erin Bransom, Kiana Ehsani, Huong Ngo, YenSung Chen, Ajay Patel, Mark Yatskar, Chris Callison-Burch, Andrew Head, Rose Hendrix, Favyen Bastani, Eli VanderBilt, Nathan Lambert, Yvonne Chou, Arnavi Chheda, Jenna Sparks, Sam Skjonsberg, Michael Schmitz, Aaron Sarnat, Byron Bischoff, Pete Walsh, Chris Newell, Piper Wolters, Tanmay Gupta, Kuo-Hao Zeng, Jon Borchardt, Dirk Groeneveld, Crystal Nam, Sophie Lebrecht, Caitlin Wittlif, Carissa Schoenick, Oscar Michel, Ranjay Krishna, Luca Weihs, Noah A. Smith, Hannaneh Hajishirzi, Ross Girshick, Ali Farhadi, and Aniruddha Kembhavi. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models, 2024. 1

[7] Huilin Deng, Ding Zou, Rui Ma, Hongchen Luo, Yang Cao, and Yu Kang. Boosting the generalization and reasoning of vision language models with curriculum reinforcement learning, 2025. 2

[8] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. 1, 2

[9] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023. 4

[10] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 3

[11] Liang Mi, Weijun Wang, Wenming Tu, Qingfeng He, Rui Kong, Xinyu Fang, Yazhu Dong, Yikang Zhang, Yunchun Li, Meng Li, Haipeng Dai, Guihai Chen, and Yunxin Liu. Empower vision applications with lora lmm, 2025. 2

[12] Roni Paiss, Ariel Ephrat, Omer Tov, Shiran Zada, Inbar Mosseri, Michal Irani, and Tali Dekel. Teaching clip to count to ten. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3170–3180, 2023. 1

[13] Muhammad Fetrat Qharabagh, Mohammadreza Ghofrani, and Kimon Fountoulakis. Lvlm-count: Enhancing the counting ability of large vision-language models, 2025. 1

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

[15] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. 2, 3

[16] Haozhan Shen, Peng Liu, Jingcheng Li, Chunxin Fang, Yibo Ma, Jiajia Liao, Qiaoli Shen, Zilun Zhang, Kangjia Zhao,

Qianqian Zhang, Ruochen Xu, and Tiancheng Zhao. Vlm-r1: A stable and generalizable r1-style large vision-language model, 2025. 2, 3

[17] Yuge (Jimmy) Shi. A vision researcher's guide to some rl stuff: Ppo & grpo, 2025. Accessed: 2025-04-14. 2