

Scaling Agentic Reinforcement Learning for Tool-Integrated Reasoning in VLMs

Meng Lu^{1*}, Ran Xu^{2*}, Yi Fang¹, Wenxuan Zhang³, Yue Yu⁴, Gaurav Srivastava¹, Yuchen Zhuang⁴,
Mohamed Elhoseiny³, Charles Fleming⁵, Carl Yang², Zhengzhong Tu⁶, Yang Xie⁷,
Guanghua Xiao⁷, Di Jin^{8†}, Wenqi Shi^{7†}, Xuan Wang^{1†}

¹Virginia Tech ²Emory University ³KAUST ⁴Georgia Tech ⁵Cisco

⁶TAMU ⁷UT Southwestern Medical Center ⁸Eigen AI

Abstract

While recent vision-language models (VLMs) demonstrate strong image understanding, their ability to “think with images,” i.e., to reason through multi-step visual interactions, remains limited. We introduce VISTA-Gym, a scalable training environment for incentivizing tool-integrated visual reasoning capabilities in VLMs. VISTA-Gym unifies diverse real-world multimodal reasoning tasks (7 tasks from 13 datasets in total) with a standardized interface for visual tools (e.g., grounding, parsing), executable interaction loops, verifiable feedback signals, and efficient trajectory logging, enabling visual agentic reinforcement learning at scale. While recent VLMs exhibit strong text-only reasoning, both proprietary and open-source models still struggle with tool selection, invocation, and coordination. With VISTA-Gym, we train VISTA-R1 to interleave tool-use with agentic reasoning via multi-turn trajectory sampling and end-to-end reinforcement learning. Extensive experiments across 11 public reasoning-intensive VQA benchmarks show that VISTA-R1-8B outperforms state-of-the-art baselines with similar sizes by 9.51%-18.72%, demonstrating VISTA-Gym as an effective training ground to unlock the tool-integrated reasoning capabilities for VLMs. Code is available at <https://github.com/Lucanyc/VISTA-Gym>.

1. Introduction

Recent progress in VLMs has demonstrated strong performance across tasks such as visual question answering, multimodal reasoning, and grounded mathematical problem solving [3, 14, 20, 66]. Many of these advancements stem from incorporating text-based reasoning paradigms, particularly the Chain-of-Thought (CoT) [71] and reinforcement learning (RL) [18], which decomposes complex tasks into intermediate text reasoning steps for problem solving, and

leverages outcome-based rewards to refine reasoning quality [22, 35, 37, 53, 61, 86].

Most of the existing VLM reasoning processes still rely on static visual embeddings and shallow cross-modal alignment. As a result, their text-only reasoning struggles to capture the fine-grained visual structures, spatial relationships, and quantitative dependencies present in real-world scenes [16, 87]. These limitations underscore the need for thinking-with-image paradigms [65], where reasoning is tightly coupled with visual perception, enabling richer cross-modal interaction and step-by-step visual reasoning.

To enhance visual-centric reasoning and encourage *thinking-with-image* behaviors in VLMs, tool-integrated reasoning (TIR) [19, 87] has been introduced recently. TIR equips models with external tools, such as *grounding*, *zoom-in*, and *search*, to facilitate fine-grained perception and reasoning over object interactions. Despite its promise, current open-sourced VLMs still remain inadequate at leveraging these tools for effective reasoning, as shown in Figure 1. This limitation highlights the urgent need for an effective training environment and strategies to *select*, *invoke*, and *coordinate* visual tools dynamically. While there are several studies dedicated to improving tool usage capabilities for VLMs [16, 64, 72, 87], those works are often narrow in scope, often confined to specific tasks (e.g., using image search for knowledge or zoom-in for object grounding). In parallel, recent efforts in gaming and robotics [9, 68] have introduced unified environments for training VLM agents. These platforms provide controllable dynamics, grounded feedback, and structured task spaces, offering clear advantages for robot learning and conceptually aligning with TIR. However, the simulation of tool-integrated thinking in VLMs—especially for open-domain visual reasoning—remains largely underexplored.

Motivated by these challenges, we introduce VISTA-Gym (**V**isual-centric **T**ool-integrated **A**gentic training environment), a scalable, agentic training environment designed to systematically enhance the reasoning and decision-making capabilities of VLM agents across complex, real-

*Equal contribution first authors: Meng Lu, Ran Xu. †Equal correspondence to: Di Jin, Wenqi Shi, Xuan Wang.

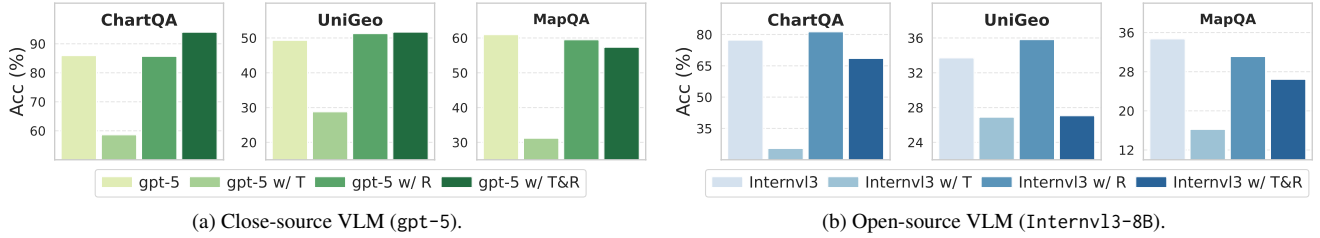


Figure 1. Directly augmenting VLMs with tools significantly degrades accuracy (*w/ T*), yet intrinsic reasoning offers limited gains on complex VQA (*w/ R*). Supplying *tool-selection prior knowledge* and interleaving reasoning with tool execution improve performance (*w/ T&R*); gains are task-dependent for commercial VLMs, while small open-source VLMs remain particularly struggling.

world scenarios. Specifically, VISTA-Gym wraps up visual tool operations and provides textual feedback on receiving tool call commands by VLMs, and features:

- **Diverse multi-modal reasoning tasks.** VISTA-Gym provides a comprehensive VQA suite spanning 7 reasoning tasks across 13 public datasets, supporting training and evaluation of agents with strong generalization and tool-integration skills.
- **Unified, extensible tool interface.** VISTA-Gym exposes a standardized API over 26 pre-defined tools that support perception, symbolic manipulation, chart and document interpretation, grounding high-level reasoning in structured intermediate results.
- **Scalable interactive infrastructure.** VISTA-Gym accelerates agent training via multithreading, parallel execution, and sequential sampling, enabling efficient trajectory collection and large-scale automated evaluation compatible with diverse agent scaffolds.

Building on VISTA-Gym, we further develop VISTA-R1, a VLM-based agent trained for robust, tool-augmented reasoning. Across five in-domain and six out-of-domain reasoning-intensive VQA benchmarks, VISTA-R1-8B surpasses state-of-the-art open-source baselines of comparable size by 9.51%–18.72%. VISTA-Gym proves to be an effective and scalable solution for developing VLM agents that can robustly interleave reasoning and tool use to solve complex, multi-step visual problems.

2. Related Works

RL for Tool-Integrated Reasoning in VLMs. To better characterize the visual information, recent works explore the “thinking with images” paradigm, which goes beyond standard reasoning steps by incorporating additional tools in the reasoning process [21, 29]. DeepMMSearch-R1 [47] and MMSearch-R1 [72] leverage image search tools to augment the context with external knowledge. Other approaches [16, 64, 86, 87] utilize *zoom-in* operations to focus on fine-grained visual regions, thereby improving grounding and multi-step perception reasoning. ReLook [30] utilizes a VLM as an auxiliary tool to enable agentic training via cross-model interaction. Despite these advances, most existing methods rely on a single specialized tool and are re-

stricted to narrow task domains, thereby limiting their generalization to broader multimodal reasoning scenarios.

RL Training Environment for Agentic Reasoning. To facilitate agentic model training, several frameworks have been proposed, including GEM [36], AgentGym-RL [74], SkyRL-Gym [5], Collabrative GYM [58] and RAGEN [70] for training general agents. Beyond general frameworks, many efforts target domain-specific applications, such as text-based reasoning [63], software engineering [15, 23, 51, 80], machine learning engineering [48, 54, 55], search and web browsing [10, 75, 77], and scientific reasoning [69, 78]. Moreover, the vast majority of these environments operate purely in text-only settings, providing limited support for multimodal grounding or visual reasoning. For VLMs, to the best of our knowledge, only two environments, namely VLM-Gym [9] and VAGEN [68], are designed to support VLM training, focusing respectively on compositional visual games and embodiment tasks that require intermediate state information. VISTA-Gym fills this gap by providing a scalable environment for tool-integrated RL, enabling systematic development of agentic visual reasoning for VLMs.

3. Preliminaries

Problem Setup. We study visually grounded question answering requiring complex reasoning over both image and text to derive a final prediction \hat{y} . The model is equipped with a predefined, finite tool set (*i.e.*, action space) \mathcal{A} . Formally, the objective is to synthesize a trajectory U with interleaved reasoning steps g_t and external tool invocations $a_t \in \mathcal{A}$. The resulting solution path is defined as $U = (g_0, a_0, \dots, g_T, \hat{y})$, where T denotes the number of tool-interaction turns required to reach the final answer \hat{y} .

Tool-Integrated Reasoning in VLMs. Formally, we frame TIR in VLMs as a Partially Observable Markov Decision Process (POMDP) defined by the tuple $\langle \mathcal{S}, \mathcal{O}, \mathcal{A}, \mathcal{I}, \mathcal{P}, \mathcal{R} \rangle$, where $\mathcal{I}, \mathcal{S}, \mathcal{O}, \mathcal{A}$, and \mathcal{R} represent the spaces for instructions, environment states, agent observations, actions, and rewards, respectively. Let $\mathcal{P} : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$ represent the deterministic state transition function. Following ReAct [82], we structure the agent outputs to prioritize reasoning thought before action. Given a problem description $x \in \mathcal{I}$, the history, and the current feedback,

Table 1. Error pattern identification and distribution from 500 error samples. Note that one case may contain multiple error types.

ID	Error Type	GPT-5	InternVL3-8B
E1	Invocation schema violation (wrong function call structure)	12.8%	3.8%
E2	Invalid argument name (wrong argument name)	0.0%	44.8%
E3	Invalid argument value (wrong argument value format)	18.2%	18.0%
E4	Incorrect argument value (wrong argument value content)	39.6%	57.4%
E5	Invalid output from tool execution (wrong answer format)	25.6%	0.0%
E6	Incorrect reasoning from tool execution	28.1%	64.8%

at each turn t , the agent generates the thought $g_{t+1} \sim \pi_\theta(\cdot|x, g_1, a_1, o_1, \dots, g_t, a_t, o_t)$ first and the subsequent action $a_{t+1} \sim \pi_\theta(\cdot|x, g_1, a_1, o_1, \dots, g_t, a_t, o_t, g_{t+1})$. Following a_{t+1} , the environment transitions to a new state s_{t+1} via $\mathcal{P}(s_{t+1}|s_t, a_t)$, and provides a new partial observation $o_{t+1} \sim \mathcal{O}(\cdot|s_{t+1})$. Thus, the entire trajectory is:

$$\tau = (g_0, a_0, o_0, \dots, o_{T-1}, g_T, \hat{y}) \sim \pi_\theta(\tau|x),$$

where \hat{y} is the final answer obtained from the model generations, and π_θ can be decomposed:

$$\begin{aligned} \pi_\theta(\tau|x) &= \pi_\theta(g_T|x, c_{T-1}) \cdot \prod_{t=0}^{T-1} \pi_\theta(g_t, a_t|x, c_{t-1}) \\ &= \pi_\theta(g_T|x, c_{T-1}) \cdot \prod_{t=0}^{T-1} \pi_\theta(a_t|x, c_{t-1}, g_t) \cdot \pi_\theta(g_t|x, c_{t-1}), \end{aligned}$$

where $c_{t-1} = (g_0, a_0, o_0, \dots, g_{t-1}, a_{t-1}, o_{t-1})$ represents the interactive history up to $t-1$.

Exploratory Experiments. We conduct exploratory experiments on three VQA tasks [6, 8, 43] with both proprietary and open-source VLMs. As shown in Fig. 1, naively enabling tools for base models induces a sharp accuracy drop: *without instruction or reasoning priors, external tools act as distractors rather than aids*. To diagnose failure modes, we annotate 500 tool-enabled errors across GPT-5 and InternVL3-8B (Table 1). The majority of failures arise from the *if/when/which/how of tool calls*, schema/argument selection, and correctness (E1–E5); followed by *incorrect post-tool reasoning* (E6). These observations motivate us to train VLMs for TIR; To facilitate easy scale-up, we introduce VISTA-Gym (Figure 2): we curate verifiable, visual-centric tasks with tools (Sec. 4.1), instantiate an interactive, executable environment (Sec. 4.2) with scalable training facilities (Sec. 4.3), and systematically improve open-source VLM agent within VISTA-Gym that *interleaves* reasoning with tool execution (Sec. 5).

4. VISTA-Gym: A Scalable Tool-Integrated Agentic Training Environment for VLMs

4.1. Diverse Task and Tool Collection

Reasoning-Intensive VQA Tasks. VISTA-Gym comprises a unified training and evaluation environment that couples diverse multimodal tasks with executable tools and verifiers, emphasizing not only final answers but also *au-*

ditable sequences of tool calls that ground those answers. We curate *verifiable* instances from 13 established benchmarks to ensure broad coverage of difficulty and reasoning types, spanning perception (vision), symbolic manipulation (math/geometry), and language understanding (document/chart interpretation). Specifically, training tasks span seven complementary axes (Appendix A): (1) *Chart Understanding* (FigureQA [26], ChartQA [43]), (2) *Geometric Reasoning* (Geometry3K [38], GeoQA [7], UniGeo [8]), (3) *Geospatial Reasoning* (MapQA [6], InfographicVQA [45]), (4) *Scientific Reasoning* (ScienceQA [40], VizWiz [4]), (5) *Document Understanding* (DocVQA [44]), (6) *Spatial/Compositional Reasoning* (CLEVR [25]), and (7) *Others* (ThinkVL [11], A-OKVQA [57]).

Visual-Centric Tool Sets. Small open-source VLMs often lack the fine-grained perception and domain routines (*e.g.*, precise localization, symbolic formalization, dense chart parsing) required by the above tasks. VISTA-Gym therefore exposes a standardized, extensible tool interface that lets agents offload these subproblems to reliable modules, *grounding* high-level reasoning in structured intermediate results (*e.g.*, tables, text). Twenty-six tools are organized into four families (Appendix B): (1) *Perception*, such as GroundingDINO [33], SAM [28], EasyOCR; (2) *Chart Understanding*, such as ChartMoE [79]; (3) *Diagram Formalization*, such as CDL, Inter-GPS; and (4) *Math Solvers*, such as G-LLaVA [17], MultiMath [52].

4.2. Executable Interactive Environment

Interface. VISTA-Gym exposes a Gymnasium-style API with `reset()` and `step()`. Given an instruction $x \in \mathcal{I}$, `reset` returns the initial partial observation o_0 (question text and supporting image[s]) and initializes the interaction history c_0 . Each episode is a POMDP mentioned in Sec. 3.

Action Space. The action space \mathcal{A} is strictly constrained by the available toolset. We formalize an action as a typed tuple $a_t \in \mathcal{A}$ consisting of the unique identifier and the corresponding arguments passed to the interface for execution.

Observation Space. The observation space \mathcal{O} comprises environmental feedback after tool execution. Each $o_t \in \mathcal{O}$ encapsulates either successful execution results or runtime error messages (if any), serving as external signals for verifying intermediate hypotheses and adjusting the subsequent reasoning trajectory.

4.3. Scalable Training Facility

VLMs as Tools. To enhance visual perception, we include compute-intensive VLMs like G-LLaVA [17] and ChartMoE [79] as interactive services. To integrate them efficiently into distributed RL, we deploy a highly concurrent microservice architecture that encapsulates each VLM as an independent HTTP service with three layers: (i) a **FastAPI** front end that exposes RESTful endpoints with

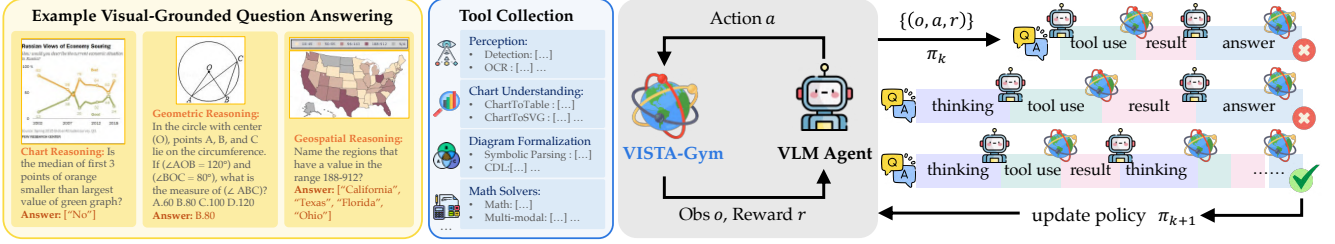


Figure 2. Overview of VISTA-Gym. VISTA-Gym contains a comprehensive suite of reasoning-intensive VQA tasks and tools in an interactive execution environment, scaling visual-centric tool-integrated agentic training for VLM agents.

asynchronous batched requests; (ii) an intermediate **Tool** layer that parses instruction actions, retrieves image data from trajectory metadata, and formats observations; and (iii) a **Ray Actor** layer that keeps model weights resident in GPU memory after initialization, eliminating the prohibitive latency of repeated model (re)loading under high-frequency tool calls.

Asynchronous Training. To sustain high throughput during RL rollouts, we use Ray to orchestrate concurrency. At each step, the policy emits a `<think>` segment followed by a `<tool_call>`; on generation of the sentinel token `</tool_call>`, decoding halts and the framework assembles batched HTTP requests that include trajectory identifiers and image paths. Ray manages request queues and load balancing across tool servers. For resource efficiency, compute-heavy VLM tools are pinned to dedicated GPUs, while lightweight utilities are multiplexed on shared CPUs.

Extensible Tool Set. To support rapid expansion beyond tools used in our experiments, we provide a generic BaseTool interface that enables *plug-and-play* integration of new tools with minimal boilerplate. Operational robustness is ensured via health and metric endpoints (`/health`, `/metrics`) and Ray’s automated failure recovery, which restarts crashed actors without disrupting ongoing training.

5. VISTA-R1: Bootstrapping Tool-Integrated Reasoning via RL for VLMs

5.1. Two-Stage Training Framework

With the developed VISTA-Gym, we enable a two-stage training framework. First, we bootstrap an interactive agent with behavioral cloning (BC), instilling instruction-following and basic tool-use. We then refine it with multi-turn, online RL, which encourages deeper reasoning and disciplined, reasoning-guided tool orchestration.

Stage I: Warmup w. Imitation Learning. We initialize the policy by BC on synthesized expert trajectories that *explicitly* interleave thoughts and actions. (i) *Candidate generation and filtering.* We first generate tool-executing trajectories with a proprietary model (GPT-5) and retain only those whose final answers exactly match ground truth (outcome-based filtering). (ii) *Rationale densifica-*

tion. To strengthen supervision on reasoning, we replace concise rationales with extended traces produced by an open-weights expert (Qwen3-VL-235B-A22B-Thinking). Let \mathcal{D} denote the resulting set of tuples (x, τ) where $\tau = (g_1, a_1, \dots, g_T, \hat{y})$ and c_{t-1} is the interaction history up to step $t-1$. The BC objective maximizes the likelihood of interleaved thought-action tokens:

$$\begin{aligned} \mathcal{L}_{\text{BC}}(\theta) &= \mathbb{E}_{(x, \tau) \sim \mathcal{D}} [\log \pi_{\theta}(\tau|x)] \\ &= \mathbb{E} \left[\sum_{t=0}^{T-1} \log \pi_{\theta}(a_t|x, c_{t-1}, g_t) + \sum_{t=0}^T \log \pi_{\theta}(g_t|x, c_{t-1}) \right]. \end{aligned}$$

The synthesized corpus covers a diverse tool mix (Figure 3), providing a robust prior for tool syntax and selection.

Stage II: Online RL. After SFT, we train the agent in the executable environment with multi-turn rollouts, where each step emits a reasoning segment followed by a function call; VISTA-Gym executes the call and appends the (structured) feedback to the context for continued reasoning. For policy improvement, we adopt Group Relative Policy Optimization (GRPO) [59] with group-normalized advantages over G rollouts:

$$\begin{aligned} \mathcal{L}_{\text{GRPO}}(\theta) &= \frac{1}{G} \sum_{i=1}^G \frac{1}{|\mathcal{T}_i|} \sum_{k=1}^{|\mathcal{T}_i|} \min[r_{i,k}(\theta) \cdot \hat{A}_{i,k}, \\ &\quad \text{clip}(r_{i,k}(\theta), 1 - \epsilon, 1 + \epsilon) \cdot \hat{A}_{i,k}], \end{aligned}$$

where $r_{i,k}(\theta)$ denotes the token-level importance ratio at the k -th token.

$$r_{i,k}(\theta) = \frac{\pi_{\theta}(\tau_{i,k}|\tau_{i,<k})}{\pi_{\text{old}}(\tau_{i,k}|\tau_{i,<k})},$$

and $\hat{A}_{i,k}$ is the normalized advantage across all tokens:

$$\hat{A}_{i,k} = \frac{R(\tau_i) - \text{mean}(\{R(\tau_1), \dots, R(\tau_G)\})}{\text{std}(\{R(\tau_1), \dots, R(\tau_G)\})}.$$

5.2. Reward Design

We employ a multi-round interaction protocol, $U = (u_0, u_1, \dots, u_T)$, for agent training, and design the following rewards R for policy update. This protocol mandates the structural format for each turn u_i , enforcing an explicit *think*→*tool_call* loop prior to a *think*→*answer* termination:

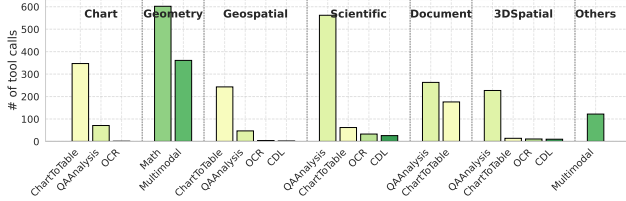


Figure 3. Top tool call distribution of different tasks in SFT data.

- **Turn ($< T$) (Tool Call):** the first turns $u_{i < T}$ should generate reasoning with function calls:

$$u_{i < T} = \langle \text{think} \rangle \dots \langle \text{think} \rangle \langle \text{tool_call} \rangle \dots \langle \text{tool_call} \rangle.$$

- **Turn T (Final Answer):** the last turn u_T should generate reasoning with final answer:

$$u_T = \langle \text{think} \rangle \dots \langle \text{think} \rangle \langle \text{answer} \rangle \dots \langle \text{answer} \rangle.$$

Repetition Penalty. We first apply a high-priority repetition detector $R_{\text{rep}}(U) \in \{-3.0, -2.0, -1.5, 0\}$ that scans for contiguous token/phrase/character repeats and assigns a severity-dependent negative reward (extreme, severe, moderate, none). This term dominates all subsequent logic.

Format Reward. Conditional on *no* repetition, $R_{\text{rep}}(U)=0$, we validate structural well-formedness (*e.g.*, correct tags, order, and non-nested closure) at every turn. We define:

$$R_{\text{format}}(U) = (\mathbb{I}\{\text{all } u_i \text{ conforms the format}\} - 0.5) \times 2.$$

Correctness Reward. We extract the final prediction \hat{y} from $\langle \text{answer} \rangle \dots \langle \text{answer} \rangle$ for rule-based checking. To keep the signal low-noise and format-aware, correctness is credited only for repetition-free, well-formed outputs:

$$R_{\text{correct}}(U) = \mathbb{I}\{\hat{y} = y\}.$$

Final Reward. The rollout-level reward is the sum of the three components:

$$R(U) = R_{\text{rep}}(U) + R_{\text{format}}(U) + R_{\text{correct}}(U).$$

This sparse, format-aware design allocates positive reward only to generations that are repetition-free, structurally valid, and correct, thereby encouraging the policy to internalize the intended *think*→*tool_call*→*answer* protocol rather than exploiting intermediate heuristics.

6. Experiments

6.1. Experiment Setups

Evaluation Datasets. Following several prior works [21, 22, 31, 34], we focus on *reasoning-intensive VQA tasks*, evaluating the effectiveness of VISTA-Gym in improving the TIR of open-source VLMs. For *in-distribution* evaluation, we select five out of the thirteen datasets in VISTA-Gym training pool with official held-out test

splits: (1) ChartQA [43], (2) Geometry3K [38], (3) GeoQA [7], (4) UniGeo [8], and (5) MapQA [6]. To assess *out-of-distribution* generalization, we evaluate on six additional benchmarks not used for training: (6) TABMWP [41], (7) AI2D [27], (8) PlotQA [46], (9) CLEVR-Math [32], (10) IconQA [39], and (11) Math-Vista [42]. Detailed dataset information is in Appendix A.

Baselines. We consider the following baselines on VISTA-Gym: (i) *API-based proprietary VLMs*, including GPT-5 [49], GPT-5-mini [49], GPT-o4-mini [50], GPT-o3 [50], Gemini-2.5-Pro [12], Gemini-2.5-Flash [12], Claude-4.5-Sonnet [2]; (ii) *Open-source VLMs*, including InternVL3 [88], Qwen2.5-VL [3], LLaVA-OneVision-1.5 [1]; (iii) *Tool/Reasoning-integrated VLMs*, including VTool-R1 [73], R1-VL [85], R1-Onevision [81], Perception-R1 [83]. All baselines are evaluated *without tool access*, as naive tool exposure significantly degrades performance without extensive agentic training (see Table 5 in Appendix I); except for tool-integrated baselines following their original tool configurations. Additional baseline details are available in Appendix C.

Implementation Details. We consider four different backbones for VISTA-R1 with varying sizes, including InternVL3-2B/8B/14B and Qwen2.5-VL-7B, and implement training with VerI-Tool [24]. We adopt *accuracy* (ACC) as the primary evaluation metric. All experiments are conducted with 8 NVIDIA H200 GPUs, each equipped with 141GB of memory. See Appendix D and H for prompt templates and other implementation details, respectively.

6.2. Main Experiment Results

Table 2 shows the main results of VISTA-R1 trained in VISTA-Gym against baselines on eleven visual-reasoning datasets. From the results, we have the following key observations: (i) **VISTA-R1 achieves strong performance over baselines.** Notably, VISTA-R1-8B outperforms baselines with similar sizes by 9.51%-18.72% with tools and 2.03%-11.24% even *without* tools, respectively. (ii) **RL is critical for boosting TIR in VLMs.** Simply augmenting VLMs with tools without explicit reasoning supervision degrades accuracy. By contrast, RL delivers substantial gains, indicating that base checkpoints do not exhibit robust TIR ability and that RL is crucial for unlocking tool-use capabilities in visual reasoning. Moreover, RL confers strong generalization: for example, VISTA-R1-8B attains accuracy on out-of-distribution VQA benchmarks comparable to substantially larger proprietary models (*e.g.*, GPT-o3 and Claude-4.5-Sonnet). (iii) **VISTA-R1 has a strong parameter efficiency.** VISTA-R1-2B achieves competitive or even better performance with 8B baselines, while VISTA-R1-8B performs comparably to baselines with 38B parameters. The superiority of VISTA-R1 demonstrates that **VISTA-Gym provides a scalable training ground for**

Table 2. Main results (acc%) on five in-distribution and six out-of-distribution VQA benchmarks. † indicates results reported from the original papers. *w/o Tools* excludes tool access from both training and inference stage; *w/o Reasoning* removes RL training stage.

Baselines (↓)	In-Distribution						Out-of-Distribution						All	
	ChartQA	Geometry3K	GeoQA	UniGeo	MapQA	avg.	TABMWP	AI2D	PlotQA	CLEVR-Math	IconQA	MathVista		avg.
<i>Commercial VLMs (For reference)</i>														
GPT-5	85.92	94.84	90.91	49.34	60.95	76.39	99.90	86.10	72.90	26.00	87.20	80.20	75.38	75.84
GPT-5-mini	82.80	91.01	88.01	46.02	61.35	73.84	98.57	85.85	70.95	27.15	87.73	79.30	74.93	74.43
GPT-o4-mini	85.24	90.02	89.17	40.72	62.25	73.48	96.73	84.38	73.35	27.35	86.53	78.00	74.39	73.98
GPT-o3	85.63	92.35	90.14	50.27	54.25	74.53	99.18	79.58	71.85	25.55	81.87	75.40	72.24	73.28
Gemini-2.5-Pro	83.64	94.34	86.85	60.34	64.20	77.87	99.90	90.28	77.40	25.70	96.93	82.00	78.70	78.33
Gemini-2.5-Flash	85.85	92.85	88.78	46.15	53.35	73.40	97.96	84.50	68.75	23.45	84.73	77.90	72.88	73.12
Claude-4.5-Sonnet	85.03	85.19	92.26	55.57	91.85	81.98	99.49	80.44	82.95	23.95	94.50	75.10	76.07	78.76
<i>Base Size: < 7B parameters</i>														
InternVL3-2B	24.08	40.26	40.21	21.88	16.35	28.56	82.41	63.59	38.94	17.90	60.13	32.70	49.28	39.86
VISTA-R1 (InternVL3-2B)	88.55	56.24	68.54	42.31	33.35	57.80	85.35	73.80	63.85	39.81	84.13	60.00	67.82	63.27
w/o Tools	72.88	43.93	66.30	39.52	39.45	52.42	74.23	64.70	35.90	15.93	63.53	49.60	50.65	51.45
w/o Reasoning	43.76	43.43	54.93	23.47	25.05	38.13	85.27	51.29	30.85	21.24	68.53	28.30	47.58	43.28
Qwen2.5-VL-3B	40.08	38.09	34.40	19.00	18.00	29.91	82.05	64.69	40.02	23.07	41.00	34.60	47.57	39.51
LLaVA-OneVision-1.5-4B	61.60	50.92	29.79	14.06	27.25	36.72	93.15	73.68	43.90	24.00	84.80	44.50	60.67	49.79
<i>Large Size: 7 - 13B parameters</i>														
Qwen2.5-VL-7B	76.40	39.43	41.39	30.64	26.30	42.83	80.64	70.44	65.53	25.18	61.45	59.20	60.41	52.42
VISTA-R1 (Qwen2.5-VL-7B)	90.08	53.27	71.32	44.43	66.85	65.19	86.40	72.68	74.82	33.37	72.51	63.00	67.13	66.25
w/o Tools	83.32	48.92	68.11	33.42	51.35	57.02	64.13	56.09	76.45	26.92	68.58	56.10	58.05	57.58
w/o Reasoning	79.68	47.92	58.61	34.88	26.70	49.56	82.11	72.09	66.90	17.93	82.87	42.50	60.73	55.65
VTool-R1-7B	80.70 [†]	62.06	65.18	33.29	19.65	52.18	64.21	79.34	48.85	19.70	75.67	36.60	54.06	53.20
R1-VL-7B	83.90 [†]	58.90	62.09	37.93	23.87	57.34	92.64	76.51	48.50	18.00	87.27	63.50 [†]	65.20	61.63
R1-Onevision-7B	59.92	55.79	57.47	28.30	33.20	46.94	84.19	61.75	35.40	19.30	79.30	64.10	57.34	52.61
Perception-R1-7B	76.41	48.59	51.23	21.22	40.52	47.59	84.05	67.77	44.30	18.00	81.87	74.20 [†]	61.70	55.29
InternVL3-8B	77.32	47.09	44.87	33.71	34.70	47.54	95.32	74.54	63.05	28.95	73.27	59.60	65.79	57.49
VISTA-R1 (InternVL3-8B)	91.92	61.27	76.41	49.67	68.45	69.54	96.52	79.86	66.38	38.20	91.09	62.80	72.48	71.14
w/o Tool	84.24	51.47	73.10	34.70	53.50	59.40	94.68	75.37	60.25	29.24	80.93	62.80	67.21	63.66
w/o Reasoning	68.56	35.64	55.90	27.06	26.45	42.72	92.34	50.45	38.70	28.18	80.07	29.10	53.14	48.40
LLaVA-OneVision-1.5-8B	64.96	54.08	32.69	16.05	32.40	40.04	94.27	77.37	46.60	25.00	89.67	47.70	63.44	52.80
<i>XL Size: > 13B parameters</i>														
InternVL3-14B	87.30	68.89	75.05	41.78	50.77	64.76	97.71	75.40	74.12	32.83	81.47	71.67	72.20	68.82
VISTA-R1 (InternVL3-14B)	93.60	83.87	78.85	52.94	70.50	75.95	98.02	87.60	76.00	40.86	92.13	68.00	77.10	76.58
Qwen2.5-VL-32B	87.30	65.22	50.29	42.94	89.20	66.99	94.79	82.16	71.17	35.14	92.60	68.20	74.01	70.82
InternVL3-38B	89.20	70.72	79.11	45.14	55.13	67.86	98.35	83.03	78.48	33.88	83.73	74.03	75.25	71.89
InternVL3-78B	89.70	78.76	85.43	53.40	58.15	73.09	99.39	85.85	82.35	25.15	86.33	78.80	76.31	74.85

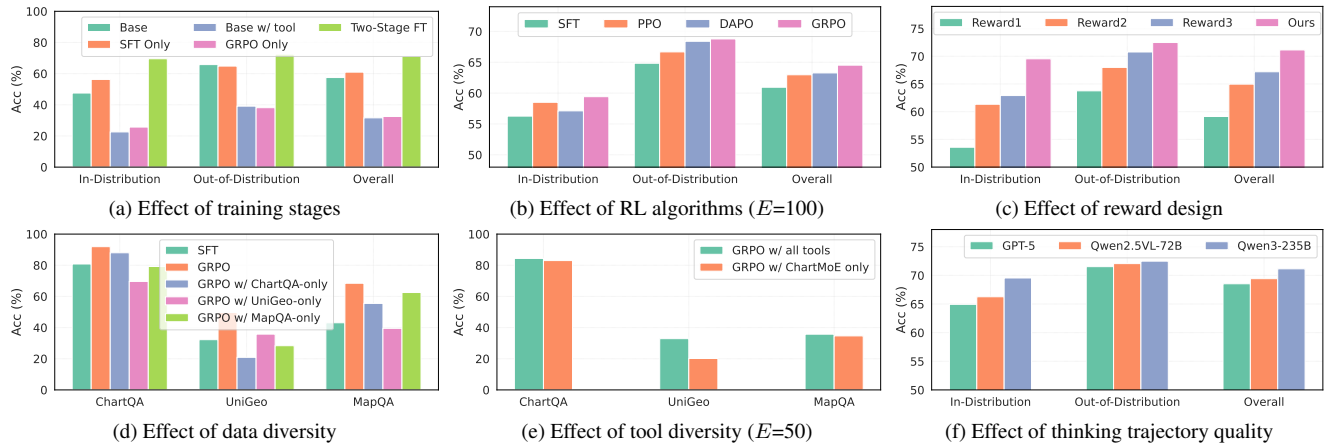


Figure 4. Ablation studies and diversity analysis with InternVL3-8B as backbone VLM.

tool-integrated agentic RL, enabling robust visual reasoning in open-source VLMs.

6.3. Ablation Studies of VISTA-R1

Effect of Tool-Integrated Reasoning. Table 2 also disentangles the roles of *reasoning* and *tool use* by reporting three variants: (1) *w/o Tools* (63.66%) that enables reasoning but disables tool access; (2) *w/o Reasoning* (48.40%) that exposes tools (with prior tool selection knowledge) without reinforcing reasoning; (3) and the *full* VISTA-R1

(71.14%) that interleaves both. While strengthening reasoning or tool-use alone sometimes yields measurable gains, it remains suboptimal compared to VISTA-R1, which synergistically couples reasoning with tool execution. Two key findings follow: (i) simply exposing tools can be *detrimental without sufficient reasoning* to navigate the action space, and (ii) optimal problem solving arises when reasoning *guides (if/when/which)* tool invocation in an interleaved loop, rather than relying on either capability in isolation.

Effect of Training Stage. Figure 4a shows that SFT serves

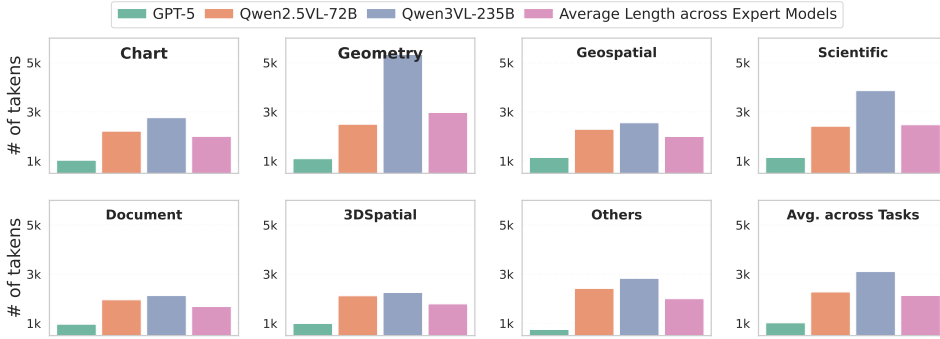


Figure 5. Quality of expert thinking trajectories by length.

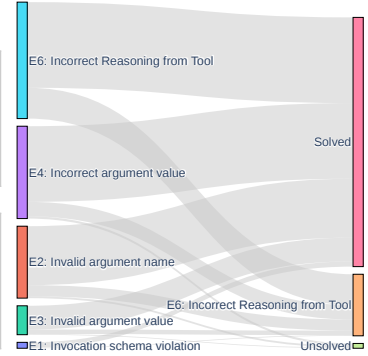


Figure 6. Error analysis.

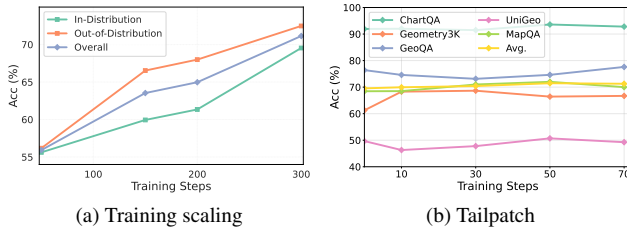


Figure 7. Training scaling from easy to hard.

as a critical *warm-up*, aligning the model with tool-use formats and syntax (+3.46%); RL then unlocks substantially larger gains (+10.19% over SFT). *Both stages contribute*: SFT establishes the necessary behavioral priors for reliable tool interaction, while RL triggers deeper, interleaved reasoning required to navigate complex tasks effectively.

Effect of RL Algorithm. In addition to GRPO, we evaluate PPO [56] and DAPO [84] as preference-optimization algorithms; yet we observe that GRPO delivers the most robust performance, as shown in Figure 4b. The DAPO deficit may be due to the elimination of *uniform-reward* groups in its preference objective: in early stage, removing all-incorrect collapses the effective batch and starves the policy of learning signal; in later stage, removing all-correct erases useful supervision for reasoning–tool coordination. In contrast, GRPO retains all rollouts and uses group-normalized advantages to deliver low-variance, difficulty-adaptive credit assignment, yielding more robust gains.

Effect of Reward Design. Figure 4c compares different types of reward design, including (1) dense reward $R_{\text{dense}} = -1 + 0.5R_{\text{format}} + 0.5R_{\text{correct}} + R_{\text{format}} \cdot R_{\text{correct}}$, (2) sparse reward $R_{\text{sparse}} = R_{\text{format}} \cdot R_{\text{correct}}$, (3) difficulty-adaptive reward $R_{\text{diff}} = R \cdot w$, where $w = \text{clamp}(2 - D, 0, 1) \times 0.5 + 0.5$ and D represents the difficulty measured by the mean base reward across all rollouts, $D = \mathbb{E}[R]$, and (4) ours. In comparison, our reward design couples a *verifiable, low-noise binary end signal with difficulty-adaptive scaling* via group-normalized advantages, yielding superior accuracy while avoiding overfitting to easy cases or exploitation of intermediate signals.

6.4. Additional Studies of VISTA-Gym

Data Diversity: Effect of Data Mixture. Figure 4d examines how task composition during RL shapes TIR. Single-task training exhibits weak cross-task transfer, especially across distinct domains such as ChartQA (chart understanding) and UniGeo (geometric reasoning), likely due to shifts in tool-use schemas and argument distributions that narrow the learned policy’s coverage of the action space and impede general TIR. By contrast, multi-task training consistently improves generalization, indicating that exposure to a diverse task mixture during training is key to learning transferable TIR policies.

Tool Diversity: Effect of Tool Mixture. Figure 4e analyzes how tool composition during RL shapes TIR using ChartMoE as an example. Training *exclusively* with chart-understanding tools leaves chart-centric tasks (ChartQA; to a lesser extent, MapQA) largely unchanged, but transfers poorly to tasks with different tool affordances (UniGeo). Similar to data mixture, exposure to heterogeneous tool schemas and argument formats also mitigates over-specialization and broadens the learned action policy.

Reasoning Trajectory Quality. We assess the *quality* of expert trajectories (GPT-5, Qwen2.5VL-72B, Qwen3VL-235B) using token length as a simple proxy in Figure 5. As expected, longer trajectories during the SFT warm-up establish better behavior prior, thus contribute to higher performance after RL (Figure 4f). Note that commercial APIs such as GPT-5 often do not expose full intermediate reasoning, yielding significantly shorter trajectories and weaker gains from agentic RL under the same training recipe.

Tail-Patch Curriculum. To overcome the performance plateau, we adopt an easy-to-hard curriculum that *tail-patch* training with increasingly challenging samples. We use rollout pass rates from the previous checkpoint as a dynamic difficulty signal and isolate *hard-but-learnable* instances with pass rates in $[0.125, 0.375]$. Continuing RL exclusively on this high-difficulty slice pushes performance beyond the initial convergence point (69.54% \rightarrow 71.27%) (Figure 7). This suggests a general principle: focusing the optimiza-

Table 3. Case studies on geometric reasoning tasks with VISTA-R1-8B trained in VISTA-Gym.

	<p>Case Study 1: Find $m \angle R$. Answer: A: 58</p> <p>InternVL3-8B-Tool: <code><tool_call>{"tool": "multimath", "task": "solve_the_problem", "arguments": {"image_url": <image_url>, "data": {"image/png": base64, iVBORw0KGgoAAAANSUHEUgAAOAAAAAHCAIAAADYXJpAAAAA3NCSV ... (including long base64 image data)}}</code><code></tool_call></code></p> <p>VISTA-R1: <code><tool_call>{"tool": "gllava", "task": "solve", "arguments": "Find $m \angle R$ from the diagram: a diamond-shaped quadrilateral with vertices R (left), Q (top), T (right), S (bottom). Angles labeled: at $R = x^\circ$, at $T = x^\circ$, at $Q = (2x+5)^\circ$, at $S = (2x+7)^\circ$."}</code> Tool Call: Correct (✓)</p>
	<p>Case Study 2: As shown in the figure, in $\odot O$, AB is a diameter and CD is a chord. $AB \perp CD$, with the foot of the perpendicular at point E. Connect CO and AD. If $\angle BOC = 30^\circ$, then the degree measure of $\angle BAD$ is (). Answer: 15°</p> <p>InternVL3-8B-Tool: ... Step 12 (Property of isosceles triangle): $\angle BAD = \angle OAD$. Answer: $\angle BAD = 15^\circ$. <code><think></code> Alright, let's tackle this geometry problem step by step. ... But according to the tool output, the answer is 15 degrees, which is option D. But my computation also says the answer is 52.5 degrees. So the answer is closer to C. <code></think></code> ... Correct Tool, Wrong Reasoning: Incorrect reasoning to answer. (✗)</p> <p>VISTA-R1: ... Step 12 (Property of isosceles triangle): $\angle BAD = \angle OAD$. Answer: $\angle BAD = 15^\circ$. <code><think></code> ... Based on the previous verification and tool output, I can analyze and provide the answer of 15°. <code></think></code> <code><answer>15^\circ</answer></code>. Reasoning: Correct (✓)</p>
	<p>Case Study 3: $AB \parallel CD$. Line EF intersects AB at point E and intersects CD at point F. EG bisects $\angle BEF$ and intersects CD at point G. If $\angle 1 = 50^\circ$, then $\angle 2$ equals (). Choices: A: 50° B: 60° C: 65° D: 90°. Answer: C: 90°.</p> <p>VISTA-R1: <code><think></code> ... ChartMoE is appropriate because it is designed to analyze structured geometric diagrams, accurately reading angle labels, identifying parallel-line configurations, and aligning textual annotations with diagram elements. I will have it parse the diagram to confirm $AB \parallel CD$, locate the transversal EF and the points E, F, G, and verify that EG bisects $\angle BEF$ while intersecting CD at G. By interpreting the angles formed at E relative to BE and FE and using the parallel-line angle relations (corresponding, alternate interior, and consecutive interior angles), ChartMoE will relate $\angle 1$ to the angle between BE and EG, propagate the bisector relation to $\angle BEF$, and then deduce the measure of $\angle 2$ at the intersection of EF and CD from the resulting angle measures. Finally, it will align that computed value with the provided choices. Therefore, I will call chartmoe now. <code></think></code> <code><tool_call>{"tool": "chartmoe", "task": "analyze"}</code> ... New Tool Selection: Expected G-LLaVA but got ChartMoE. (✓)</p>
	<p>Case Study 4: Find XZ. Answer: D: 34.</p> <p>InternVL3-8B-Tool: <code><tool_call>{"tool": "multimath", "task": "Solve the problem", "arguments": {"image": "<image>", "question": "Find X Z ... "}}</code> <code></tool_call></code> ... Step 5 (Simplify the equation): $XZ^2 = 29y^2 + 12y + 360$. Answer: <code>\boxed{29y^2 + 12y + 360}</code> <code><think></code> ... But as we saw earlier, this leads to a quadratic with no real solutions. Wait, maybe I made a mistake in the tool output. Let me double-check the steps. In Step 3: $XZ^2 = (5y - 6)^2 + (18 + 2y)^2 = 25y^2 - 60y + 36 + 324 + 72y + 4y^2 = 29y^2 + 12y + 360$. That seems correct. In Step 5: Simplify the equation: $XZ^2 = 29y^2 + 12y + 360$. <code></think></code> <code><answer>\boxed{C}</answer></code> Tool Output Error: Incorrect reasoning to answer. (✗)</p> <p>InternVL3-8B-Reasoning: <code><think></code> Alright, let's tackle this problem step by step. First, I notice that triangle WYZ is a right triangle because it has a right angle at W. ... Using the Pythagorean theorem again: $XZ^2 = WZ^2 + XY^2$. $XZ^2 = 34^2 + 34^2$, $XZ^2 = 2312$, $XZ = \sqrt{2312} = 34 * \sqrt{2}$. Therefore, the length of XZ is $34\sqrt{2}$. <code></think></code> <code><answer>34\sqrt{2}</answer></code> No Tool-use: Insufficient capability to answer. (✗)</p> <p>VISTA-R1: <code><think></code> ... Multimath is appropriate because ... I will read from the image that $WX \perp ZX$. ... Multimath will translate these visual relations into algebraic constraints, ... <code></think></code> <code><tool_call>{"tool": "multimath", "task": "solve"}</code> <code></tool_call></code> ... Step 10 (Calculate the value of XZ): $XZ = 18 + 2y = 18 + 2(\frac{6}{5}) = 18 + \frac{12}{5} = \frac{90}{5} + \frac{12}{5} = \frac{102}{5} = 20.4$. <code><think></code> Looking at the tool's solution, it set up the equation $25y^2 - 60y + 36 = 0$ and solved to get $y = 6/5$. Let me verify this step by step. ... Wait, let me check if this makes sense geometrically. ... Actually, if the tick marks indicate equal segments, shouldn't we have $5y - 6 = 18 + 2y$? Let me solve this: $5y - 6 = 18 + 2y \implies 3y = 24 \implies y = 8$. With $y = 8$, let's calculate XZ. If $XZ = 18 + 2y$, then: $XZ = 18 + 2(8) = 18 + 16 = 34$. Hmm, that's different from the tool's answer of 20.4. Let me reconsider. ... The tool calculated 20.4, but based on the equal segment marks, the answer is 34. <code></think></code> <code><answer>34</answer></code> TIR: correct (✓)</p>

tion budget on the *frontier of competence*—cases where the model is uncertain yet occasionally succeeds—sharpens long-horizon reasoning and delivers fine-grained gains.

6.5. Error Analysis and Case Study

Error Analysis. We re-evaluate error patterns on the same 500 samples from Table 1 after InternVL3-8B (left in Figure 6) training with VISTA-Gym (right). Agentic training within VISTA-Gym markedly resolves the majority of tool-call and reasoning failures observed in base models.

Case Studies. Table 3 presents four case studies illustrating the capabilities acquired through agentic training in VISTA-Gym: (i) schema- and argument-correct *tool invocation*; (ii) *post-tool reasoning* that grounds the final answer in tool outputs; (iii) *cross-tool generalization* that repurposes atypical yet useful tools (e.g., applying ChartMoE on geometry problems); and (iv) *robustness to imperfect tool outputs*, where the agent identifies and exploits partial signals over multiple turns to complete multi-step reasoning.

7. Conclusion

In this work, we introduce VISTA-Gym, a scalable training ground for tool-integrated, visual-centric agentic RL in VLMs, and VISTA-R1, an agent trained to interleave multi-turn reasoning with structured tool use. Practitioners leveraging VISTA-Gym for boosting TIR in VLMs may consider: (i) train policies that *interleave* reasoning with tools; (ii) warm up with SFT for tool call syntax/format, then apply online RL for in-depth tool-integrated visual reasoning; (iii) diversify tasks and tools to expand action-space coverage; and (iv) allocate additional RL budget to hard-but-learnable slices via tail-patching. Our verifiers emphasize terminal correctness and structural validity; richer stepwise semantics and broader tool ecosystems may further benefit long-horizon TIR. VISTA-Gym provides a scalable agentic training environment—unified interfaces, executable feedback, and efficient logging—for systematic progress in *thinking with images*.

Acknowledgments

This work was supported by NSF 2442253, NSF NAIRR Pilot with PSC Neocortex and NCSA Delta, Cisco Research, NVIDIA, Amazon, the Commonwealth Cyber Initiative, the Amazon–Virginia Tech Center for Efficient and Robust Machine Learning, the Sanghani Center for AI and Data Analytics at Virginia Tech, and the Virginia Tech Innovation Campus to XW. RX and CY were partially supported by internal funds and GPU servers provided by the Computer Science Department of Emory University, Emory Global Diabetes Center of the Woodruff Sciences Center, the US National Science Foundation under Award Numbers 2442172, 2312502, 2319449, and the US National Institutes of Health under Award Numbers K25DK135913, RF1NS139325, R01DK143456 and U18DP006922. GX was partially supported by the US National Institutes of Health under Award Numbers R01GM140012, R01GM115473, U01CA249245, and CPRIT Awards under RP230330 and RP250561. YX was partially supported by the US National Institutes of Health under Award Numbers U01AI169298, R35GM136375, and CPRIT RP240521. WS was partially supported by internal funds and GPU servers provided by the Texas Advanced Computing Center (TACC), NVIDIA Academic Grant Program, Google Gemini Academic Program Award, and Thinking Machines Lab Tinker Research Grant.

References

- [1] Xiang An, Yin Xie, Kaicheng Yang, Wenkang Zhang, Xiuwei Zhao, Zheng Cheng, Yirui Wang, Songcen Xu, Changrui Chen, Chunsheng Wu, et al. Llava-onevision-1.5: Fully open framework for democratized multimodal training. *arXiv preprint arXiv:2509.23661*, 2025. 5, 3
- [2] Team Anthropic. Introducing claude sonnet 4.5. <https://www.anthropic.com/news/claude-sonnet-4-5>, 2025. 5
- [3] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 1, 5, 4
- [4] Jeffrey P Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samuel White, et al. Vizwiz: nearly real-time answers to visual questions. In *Proceedings of the 23rd annual ACM symposium on User interface software and technology*, pages 333–342, 2010. 3, 1
- [5] Shiyi Cao, Sumanth Hegde, Dacheng Li, Tyler Griggs, Shu Liu, Eric Tang, Jiayi Pan, Xingyao Wang, Akshay Malik, Graham Neubig, Kourosh Hakhmaneshi, Richard Liaw, Philipp Moritz, Matei Zaharia, Joseph E. Gonzalez, and Ion Stoica. Skylr-v0: Train real-world long-horizon agents via reinforcement learning, 2025. 2
- [6] Shuaichen Chang, David Palzer, Jialin Li, Eric Fosler-Lussier, and Ningchuan Xiao. MapQA: A dataset for question answering on choropleth maps. In *NeurIPS 2022 First Table Representation Workshop*, 2022. 3, 5, 1
- [7] Jiaqi Chen, Jianheng Tang, Jinghui Qin, Xiaodan Liang, Lingbo Liu, Eric Xing, and Liang Lin. Geoqa: A geometric question answering benchmark towards multimodal numerical reasoning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 513–523, 2021. 3, 5, 1
- [8] Jiaqi Chen, Tong Li, Jinghui Qin, Pan Lu, Liang Lin, Chongyu Chen, and Xiaodan Liang. Unigeo: Unifying geometry logical reasoning via reformulating mathematical expression. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3313–3323, 2022. 3, 5, 1
- [9] Liang Chen, Hongcheng Gao, Tianyu Liu, Zhiqi Huang, Flood Sung, Xinyu Zhou, Yuxin Wu, and Baobao Chang. G1: Bootstrapping perception and reasoning abilities of vision-language model via reinforcement learning. *arXiv preprint arXiv:2505.13426*, 2025. 1, 2
- [10] De Chezelles, Thibault Le Sellier, Sahar Omid Shayegan, Lawrence Keunho Jang, Xing Han Lù, Ori Yoran, Dehan Kong, Frank F Xu, Siva Reddy, Quentin Cappart, et al. The browsergym ecosystem for web agent research. *arXiv preprint arXiv:2412.05467*, 2024. 2
- [11] Krishna Teja Chitty-Venkata and Murali Emani. Imagenet-think-250k: A large-scale synthetic dataset for multimodal reasoning for vision language models. *arXiv preprint arXiv:2510.01582*, 2025. 3
- [12] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasapat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blstein, Ori Ram, Dan Zhang, Evan Rosen, 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. 5
- [13] Yihe Deng, Hritik Bansal, Fan Yin, Nanyun Peng, Wei Wang, and Kai-Wei Chang. Opencilthinker: Complex vision-language reasoning via iterative sft-rl cycles, 2025. 1
- [14] 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
- [15] Weihua Du, Hailei Gong, Zhan Ling, Kang Liu, Lingfeng Shen, Xuesong Yao, Yufei Xu, Dingyuan Shi, Yiming Yang, and Jiecao Chen. Generalizable end-to-end tool-use rl with synthetic codegym. *arXiv preprint arXiv:2509.17325*, 2025. 2
- [16] Yue Fan, Xuehai He, Diji Yang, Kaizhi Zheng, Ching-Chen Kuo, Yuting Zheng, Sravana Jyothi Narayanaraju, Xinze Guan, and Xin Eric Wang. Grit: Teaching mllms to think with images. *arXiv preprint arXiv:2505.15879*, 2025. 1, 2
- [17] Jiahui Gao, Renjie Pi, Jipeng Zhang, Jiacheng Ye, Wanjun Zhong, Yufei Wang, Lanqing HONG, Jianhua Han, Hang Xu, Zhenguo Li, and Lingpeng Kong. G-LLaVA: Solving geometric problem with multi-modal large language model. In *The Thirteenth International Conference on Learning Representations*, 2025. 3

- [18] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025. 1, 3
- [19] Xingang Guo, Utkarsh Tyagi, Advait Gosai, Paula Vergara, Ernesto Gabriel Hernández Montoya, Chen Bo Calvin Zhang, Bin Hu, Yunzhong He, Bing Liu, and Rakshith Sharma Srinivasa. Beyond seeing: Evaluating multimodal llms on tool-enabled image perception, transformation, and reasoning. *arXiv preprint arXiv:2510.12712*, 2025. 1
- [20] Zixian Guo, Ming Liu, Qilong Wang, Zhilong Ji, Jinfeng Bai, Lei Zhang, and Wangmeng Zuo. Integrating visual interpretation and linguistic reasoning for geometric problem solving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3988–3998, 2025. 1
- [21] Yi Han, Cheng Chi, Enshen Zhou, Shanyu Rong, Jingkun An, Pengwei Wang, Zhongyuan Wang, Lu Sheng, and Shanghang Zhang. Tiger: Tool-integrated geometric reasoning in vision-language models for robotics. *arXiv preprint arXiv:2510.07181*, 2025. 2, 5
- [22] 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, 5, 3
- [23] Naman Jain, Jaskirat Singh, Manish Shetty, Tianjun Zhang, Liang Zheng, Koushik Sen, and Ion Stoica. R2e-gym: Procedural environment generation and hybrid verifiers for scaling open-weights SWE agents. In *Second Conference on Language Modeling*, 2025. 2
- [24] Dongfu Jiang, Yi Lu, Zhuofeng Li, Zhiheng Lyu, Ping Nie, Haozhe Wang, Alex Su, Hui Chen, Kai Zou, Chao Du, et al. Verltool: Towards holistic agentic reinforcement learning with tool use. *arXiv preprint arXiv:2509.01055*, 2025. 5
- [25] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2901–2910, 2017. 3, 1
- [26] Samira Ebrahimi Kahou, Adam Atkinson, Vincent Michalski, Ákos Kádár, Adam Trischler, and Yoshua Bengio. FigureQA: An annotated figure dataset for visual reasoning, 2018. 3, 1
- [27] 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. 5, 1
- [28] 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. 3
- [29] Ming Li, Jake Zhong, Shitian Zhao, Haoquan Zhang, Shaoheng Lin, Yuxiang Lai, Wei Chen, Konstantinos Psounis, and Kaipeng Zhang. Tir-bench: A comprehensive benchmark for agentic thinking-with-images reasoning, 2025. 2
- [30] Yuhang Li, Chenchen Zhang, Ruilin Lv, Ao Liu, Ken Deng, Yuanxing Zhang, Jiaheng Liu, Wiggim Zhou, and Bo Zhou. Relook: Vision-grounded rl with a multimodal llm critic for agentic web coding. *arXiv preprint arXiv:2510.11498*, 2025. 2
- [31] Zongxia Li, Wenhao Yu, Chengsong Huang, Rui Liu, Zhenwen Liang, Fuxiao Liu, Jingxi Che, Dian Yu, Jordan Boyd-Graber, Haitao Mi, et al. Self-rewarding vision-language model via reasoning decomposition. *arXiv preprint arXiv:2508.19652*, 2025. 5, 3
- [32] Adam Dahlgren Lindström and Savitha Sam Abraham. Clevr-math: A dataset for compositional language, visual and mathematical reasoning. *arXiv preprint arXiv:2208.05358*, 2022. 5, 1
- [33] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 3
- [34] Yuqi Liu, Tianyuan Qu, Zhisheng Zhong, Bohao Peng, Shu Liu, Bei Yu, and Jiaya Jia. Visionreasoner: Unified visual perception and reasoning via reinforcement learning. *arXiv preprint arXiv:2505.12081*, 2025. 5, 3
- [35] Zuyan Liu, Yuhao Dong, Yongming Rao, Jie Zhou, and Jiwen Lu. Chain-of-spot: Interactive reasoning improves large vision-language models. *arXiv preprint arXiv:2403.12966*, 2024. 1
- [36] Zichen Liu, Anya Sims, Keyu Duan, Changyu Chen, Simon Yu, Xiangxin Zhou, Haotian Xu, Shaopan Xiong, Bo Liu, Chenmian Tan, et al. Gem: A gym for agentic llms. *arXiv preprint arXiv:2510.01051*, 2025. 2
- [37] Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi Wang. Visual-rl: Visual reinforcement fine-tuning. *arXiv preprint arXiv:2503.01785*, 2025. 1, 3
- [38] Pan Lu, Ran Gong, Shibiao Jiang, Liang Qiu, Siyuan Huang, Xiaodan Liang, and Song-chun Zhu. 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)*, pages 6774–6786, 2021. 3, 5, 1
- [39] 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. 5, 1
- [40] 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, 1
- [41] 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. 5, 1
- [42] 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. 5, 1
- [43] 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, 5, 1
- [44] 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, 1
- [45] Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and CV Jawahar. Infographicvqa. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1697–1706, 2022. 3, 1
- [46] Nitesh Methani, Pritha Ganguly, Mitesh M Khapra, and Pratyush Kumar. Plotqa: Reasoning over scientific plots. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1527–1536, 2020. 5, 1
- [47] Kartik Narayan, Yang Xu, Tian Cao, Kavya Nerella, Vishal M Patel, Navid Shiee, Peter Grasch, Chao Jia, Yinfei Yang, and Zhe Gan. Deepmmsearch-r1: Empowering multimodal llms in multimodal web search. *arXiv preprint arXiv:2510.12801*, 2025. 2
- [48] Deepak Nathani, Lovish Madaan, Nicholas Roberts, Nikolay Bashlykov, Ajay Menon, Vincent Moens, Amar Budhiraja, Despoina Magka, Vladislav Vorotilov, Gaurav Chaurasia, et al. Mlgym: A new framework and benchmark for advancing ai research agents. *arXiv preprint arXiv:2502.14499*, 2025. 2
- [49] Team OpenAI. Introducing gpt-5. <https://openai.com/index/introducing-gpt-5/>, 2025. 5
- [50] Team OpenAI. Introducing openai o3 and o4-mini. <https://openai.com/index/introducing-o3-and-o4-mini/>, 2025. 5
- [51] Jiayi Pan, Xingyao Wang, Graham Neubig, Navdeep Jaitly, Heng Ji, Alane Suhr, and Yizhe Zhang. Training software engineering agents and verifiers with SWE-gym. In *Forty-second International Conference on Machine Learning*, 2025. 2
- [52] Shuai Peng, Di Fu, Liangcai Gao, Xiuqin Zhong, Hongguang Fu, and Zhi Tang. Multimath: Bridging visual and mathematical reasoning for large language models. *arXiv preprint arXiv:2409.00147*, 2024. 3
- [53] Ji Qi, Ming Ding, Weihang Wang, Yushi Bai, Qingsong Lv, Wenyi Hong, Bin Xu, Lei Hou, Juanzi Li, Yuxiao Dong, et al. Cogcom: A visual language model with chain-of-manipulations reasoning. *arXiv preprint arXiv:2402.04236*, 2024. 1
- [54] Rushi Qiang, Yuchen Zhuang, Yinghao Li, Rongzhi Zhang, Changhao Li, Ian Shu-Hei Wong, Sherry Yang, Percy Liang, Chao Zhang, Bo Dai, et al. Mle-dojo: Interactive environments for empowering llm agents in machine learning engineering. *arXiv preprint arXiv:2505.07782*, 2025. 2
- [55] Rushi Qiang, Yuchen Zhuang, Anikait Singh, Percy Liang, Chao Zhang, Sherry Yang, and Bo Dai. Mle-smith: Scaling mle tasks with automated multi-agent pipeline. *arXiv preprint arXiv:2510.07307*, 2025. 2
- [56] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 7
- [57] 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, 1
- [58] Yijia Shao, Vinay Samuel, Yucheng Jiang, John Yang, and Diyi Yang. Collaborative gym: A framework for enabling and evaluating human-agent collaboration. *arXiv preprint arXiv:2412.15701*, 2024. 2
- [59] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. 4
- [60] 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. 3
- [61] Haozhan Shen, Kangjia Zhao, Tiancheng Zhao, Ruochen Xu, Zilun Zhang, Mingwei Zhu, and Jianwei Yin. Zoom-eye: Enhancing multimodal llms with human-like zooming capabilities through tree-based image exploration. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 6613–6629, 2025. 1
- [62] Wenhao Shi, Zhiqiang Hu, Yi Bin, Junhua Liu, Yang Yang, See-Kiong Ng, Lidong Bing, and Roy Ka-Wei Lee. Mathllava: Bootstrapping mathematical reasoning for multimodal large language models, 2024. 1
- [63] Zafir Stojanovski, Oliver Stanley, Joe Sharratt, Richard Jones, Abdulhakeem Adefioye, Jean Kaddour, and Andreas Köpf. Reasoning gym: Reasoning environments for reinforcement learning with verifiable rewards. *arXiv preprint arXiv:2505.24760*, 2025. 2
- [64] Zhaochen Su, Linjie Li, Mingyang Song, Yunzhuo Hao, Zhengyuan Yang, Jun Zhang, Guanjie Chen, Jiawei Gu, Juntao Li, Xiaoye Qu, et al. Openthinking: Learning to think with images via visual tool reinforcement learning. *arXiv preprint arXiv:2505.08617*, 2025. 1, 2
- [65] Zhaochen Su, Peng Xia, Hangyu Guo, Zhenhua Liu, Yan Ma, Xiaoye Qu, Jiaqi Liu, Yanshu Li, Kaide Zeng, Zhengyuan Yang, et al. Thinking with images for multimodal reasoning: Foundations, methods, and future frontiers. *arXiv preprint arXiv:2506.23918*, 2025. 1
- [66] Omkar Thawakar, Dinura Dissanayake, Ketan More, Ritesh Thawkar, Ahmed Heakl, Noor Ahsan, Yuhao Li, Mohammed Zumri, Jean Lahoud, Rao Muhammad Anwer, et al. Llamav-

- ol: Rethinking step-by-step visual reasoning in llms. *arXiv preprint arXiv:2501.06186*, 2025. 1
- [67] Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Mingjie Zhan, and Hongsheng Li. Measuring multimodal mathematical reasoning with math-vision dataset, 2024. 1
- [68] Kangrui Wang, Pingyue Zhang, Zihan Wang, Yaning Gao, Linjie Li, Qineng Wang, Hanyang Chen, Chi Wan, Yiping Lu, Zhengyuan Yang, et al. Vagen: Reinforcing world model reasoning for multi-turn vlm agents. *arXiv preprint arXiv:2510.16907*, 2025. 1, 2
- [69] Ruoyao Wang, Peter Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. Scienceworld: Is your agent smarter than a 5th grader? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11279–11298, 2022. 2
- [70] Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Xing Jin, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, et al. Ragen: Understanding self-evolution in llm agents via multi-turn reinforcement learning. *arXiv preprint arXiv:2504.20073*, 2025. 2
- [71] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, 2022. 1
- [72] Jinming Wu, Zihao Deng, Wei Li, Yiding Liu, Bo You, Bo Li, Zejun Ma, and Ziwei Liu. Mmsearch-rl: Incentivizing llms to search. *arXiv preprint arXiv:2506.20670*, 2025. 1, 2
- [73] Mingyuan Wu, Jingcheng Yang, Jize Jiang, Meitang Li, Kaizhuo Yan, Hanchao Yu, Minjia Zhang, Chengxiang Zhai, and Klara Nahrstedt. Vtool-rl: Vllms learn to think with images via reinforcement learning on multimodal tool use. *arXiv preprint arXiv:2505.19255*, 2025. 5, 3
- [74] Zhiheng Xi, Jixuan Huang, Chenyang Liao, Baodai Huang, Honglin Guo, Jiaqi Liu, Rui Zheng, Junjie Ye, Jiazheng Zhang, Wenxiang Chen, et al. Agentgym-rl: Training llm agents for long-horizon decision making through multi-turn reinforcement learning. *arXiv preprint arXiv:2509.08755*, 2025. 2
- [75] Guangzhi Xiong, Qiao Jin, Xiao Wang, Yin Fang, Haolin Liu, Yifan Yang, Fangyuan Chen, Zhixing Song, Dengyu Wang, Minjia Zhang, et al. Rag-gym: Optimizing reasoning and search agents with process supervision. *arXiv preprint arXiv:2502.13957*, 2025. 2
- [76] 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
- [77] Ran Xu, Yuchen Zhuang, Zihan Dong, Jonathan Wang, Yue Yu, Joyce C Ho, Linjun Zhang, Haoyu Wang, Wenqi Shi, and Carl Yang. Acesearcher: Bootstrapping reasoning and search for llms via reinforced self-play. *arXiv preprint arXiv:2509.24193*, 2025. 2
- [78] Ran Xu, Yuchen Zhuang, Yishan Zhong, Yue Yu, Xiangru Tang, Hang Wu, May D Wang, Peifeng Ruan, Donghan Yang, Tao Wang, et al. Medagentgym: Training llm agents for code-based medical reasoning at scale. *arXiv preprint arXiv:2506.04405*, 2025. 2
- [79] Zhengzhuo Xu, Bowen Qu, Yiyan Qi, SiNan Du, Chengjin Xu, Chun Yuan, and Jian Guo. Chartmoe: Mixture of diversely aligned expert connector for chart understanding. In *The Thirteenth International Conference on Learning Representations*, 2025. 3
- [80] John Yang, Kilian Lieret, Carlos E Jimenez, Alexander Wettig, Kabir Khandpur, Yanzhe Zhang, Binyuan Hui, Ofir Press, Ludwig Schmidt, and Diyi Yang. Swe-smith: Scaling data for software engineering agents. *arXiv preprint arXiv:2504.21798*, 2025. 2
- [81] 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. 5, 3
- [82] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*, 2023. 2
- [83] En Yu, Kangheng Lin, Liang Zhao, Jisheng Yin, Yana Wei, Yang Peng, Haoran Wei, Jianjian Sun, Chunrui Han, Zheng Ge, et al. Perception-rl: Pioneering perception policy with reinforcement learning. *arXiv preprint arXiv:2504.07954*, 2025. 5, 3
- [84] Qiyang Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gao-hong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025. 7
- [85] Jingyi Zhang, Jiaying 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. 5, 3
- [86] Xintong Zhang, Zhi Gao, Bofei Zhang, Pengxiang Li, Xiaowen Zhang, Yang Liu, Tao Yuan, Yuwei Wu, Yunde Jia, Song-Chun Zhu, et al. Chain-of-focus: Adaptive visual search and zooming for multimodal reasoning via rl. *arXiv preprint arXiv:2505.15436*, 2025. 1, 2
- [87] Ziwei Zheng, Michael Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, Chao Shen, and Xing Yu. Deep-eyes: Incentivizing "thinking with images" via reinforcement learning. *arXiv preprint arXiv:2505.14362*, 2025. 1, 2
- [88] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025. 5, 4

Scaling Agentic Reinforcement Learning for Tool-Integrated Reasoning in VLMs

Supplementary Material

A. Task and Data Information

Table 4 presents the statistics of the in-distribution (ID) and out-of-distribution (OOD) datasets; the additional details are as follows:

A.1. Information for In-Distribution Datasets

We curate a diverse suite of training tasks integrated into VISTA-Gym, spanning fifteen datasets across eight distinct reasoning domains:

- **Chart Understanding:** FigureQA [26] and ChartQA [43] demand precise quantitative extraction and logical inference over diverse data visualizations, requiring the agent to align visual features (bars, lines) with numerical values.
- **Geometric Reasoning:** Geometry3K [38], GeoQA [7], and UniGeo [8] necessitate grounding symbolic theorem application in diagrammatic parsing to solve multi-step spatial and quantitative problems.
- **Mathematical Reasoning:** MathVision [67] and MathV360 [62] evaluate the interpretation of complex mathematical diagrams and symbolic expressions, requiring the synthesis of visual perception with algebraic deduction.
- **Geospatial Reasoning:** MapQA [6] and InfographicVQA [45] challenge agents to perform rigorous spatial lookups, legend-symbol alignment, and information retrieval across dense graphical layouts.
- **Visual Scientific Reasoning:** ThinkVL [13], VizWiz [4], and ScienceQA [40] encompass a broad spectrum of tasks ranging from accessibility-focused visual description to domain-specific multimodal scientific inquiry.
- **Document Understanding:** DocVQA [44] tests layout-aware information extraction from unstructured documents, including financial reports, forms, and scanned articles.
- **Spatial Reasoning:** CLEVR [25] provides a diagnostic environment for compositional logic, assessing the agent’s ability to execute complex reasoning chains regarding attribute recognition, spatial relations, and counting in 3D scenes.
- **General Visual Reasoning (Others):** A-OKVQA [57] requires the retrieval of external world knowledge and commonsense reasoning to address open-ended visual questions.

A.2. Information for Out-of-Distribution Datasets

To assess generalization, we evaluate on six additional benchmarks featuring distinct reasoning:

Table 4. Dataset statistics in VISTA-Gym.

Dataset	Type	# of training instances	# of testing instances
FigureQA	ID	2000	–
ChartQA	ID	1500	1200
Geometry3k	ID	2000	550
GeoQA	ID	2000	500
UniGeo	ID	2000	750
MathVision	ID	2000	–
MathV360	ID	1500	–
MapQA	ID	1500	1900
InfographicVQA	ID	1000	–
ThinkVL	ID	2000	–
VizWiz	ID	200	–
ScienceQA	ID	1000	–
DocVQA	ID	970	–
CLEVR	ID	1000	–
A-OKVQA	ID	1500	–
TABMWP	OOD	–	960
AI2D	OOD	–	740
PlotQA	OOD	–	1900
Clever-math	OOD	–	900
IconQA	OOD	–	1500
Mathvista	OOD	–	1000
Total	–	22,170	11,900

- **TABMWP:** TABMWP [41] focuses on table-based math word problems requiring cell selection, row/column aggregation, and arithmetic reasoning grounded in semi-structured text, testing symbolic computation beyond visual chart parsing.
- **AI2D:** AI2D [27] is a diagram-centric science QA benchmark involving annotated figures (labels, arrows, parts); it stresses the joint interpretation of schematic layouts and textual cues via multi-hop reasoning.
- **PlotQA:** PlotQA [46] centers on scientific plots (axes, legends, bars/lines) demanding high-precision value reading and aggregation, serving as a complementary but distributionally distinct test bed to ChartQA.
- **CLEVR-Math:** CLEVR-Math [32] extends compositional reasoning in rendered 3D scenes with arithmetic operations, assessing program-like visual logic coupled with numeric computation.
- **IconQA:** IconQA [39] comprises textbook-style diagram and icon questions (often multiple-choice) that emphasize abstract visual relations and symbolic understanding, diverging from natural image statistics.
- **MathVista:** MathVista [42] is a holistic mathematical reasoning suite spanning real images, charts, and diagrams; it integrates OCR, quantitative reading, geometry, and multi-step deduction for comprehensive evaluation.

B. Toolset Information

In VISTA-Gym, we enable access to 26 tools from four categories detailed as follows:

Perception. The perception layer provides low-level visual signals that supply structural cues for higher-level reasoning, including:

- **Detection:** GroundingDINO is an open-set object detection model that unifies visual and textual modalities, allowing for the detection of arbitrary objects described by natural language. It performs open-set, text-conditioned object detection, localizing arbitrary entities referenced by natural-language queries.
- **Segmentation:** SAM (Segment Anything Model) is a foundational model for promptable image segmentation that can zero-shot segment any object based on flexible inputs like points, boxes, or text. It offers promptable, zero-shot segmentation, isolating regions given points, boxes, or textual prompts.
- **OCR:** EasyOCR is a versatile and ready-to-use optical character recognition tool supporting over 80 languages and various writing scripts for robust text extraction from images. It conducts multilingual optical character recognition, extracting text from images across diverse scripts and layouts.

Chart Understanding. The chart layer specializes in plot- and table-centric inference. Chart understanding tool collections like *ChartMoE* use a Mixture-of-Experts connector to enhance visual-text alignment, supporting both structured data extraction and high-level analytical reasoning for chart question answering, including:

- **Chart To Table.** The *ChartToTable* tool subset converts chart images into structured tabular data suitable for downstream numerical reasoning. *UniChart* and *DePlot* recover underlying data points by detecting axes, legends, and visual marks, while *ChartMoE.to_table* and *ChartMoE.extract_data* leverage the *ChartMoE* backbone to extract series-wise values and metadata (e.g., categories, units) with improved robustness across chart styles.
- **Chart To SVG.** The subset of *ChartToSVG* tools reconstructs vectorized representations of charts. *OpenCV* provides low-level image processing to segment graphical primitives, and *ChartDet* localizes key chart components (axes, legends, bars, lines, markers); *ChartOCR* recognizes textual elements (titles, axis labels, tick labels, legends) and anchors them to detected regions, enabling SVG-style reconstruction.
- **Chart To SCG.** The subset of *ChartToSCG* tools maps charts into structured scene graphs (SCG). *ChartDet* and *ChartOCR* are combined to instantiate nodes for visual and textual elements and edges for relations such as series membership, axis association, and legend-color bindings, yielding a graph representation amenable to symbolic rea-

soning.

- **Captioning Color.** The subset of *CaptioningColor* generates natural-language descriptions of charts with explicit grounding in visual encodings. *ChartAssistant* produces concise summaries of chart type, trends, and salient extrema, whereas *ChartVLM* provides more detailed captions that explicitly reference colors, legends, and value ranges.
- **QA Analysis.** The *QAAnalysis* supports chart question answering and intermediate analysis. *ChartMoE.answer* directly predicts answers to chart-related questions, while *ChartMoE.analyze* performs intermediate computations (e.g., differences, ratios, aggregations) over extracted data, and *ChartMoE.describe* generates explanatory textual analyses that connect the visual structure, quantitative reasoning steps, and final answer.

Diagram Formalization. This layer converts visual diagrams into symbolic structures amenable to automated reasoning. *DiagramFormalizer* parses geometric and schematic diagrams into *ConsCDL* and *ImgCDL*-style formal languages, enabling symbolic representations for downstream deductive geometric reasoning.

- **CDL.** The *CDL* tools convert geometric diagrams and problem statements into formal constraint languages suitable for symbolic solvers. *image_cd1* maps raw diagram images to *ImgCDL*-style primitives (points, lines, circles, incidences), whereas *text_cd1* parses textual problem descriptions into *ConsCDL*-style constraints and goals. *construction_cd1* recovers the underlying construction sequence of a diagram, and *goal_cd1* extracts explicit target predicates (e.g., lengths, angles, congruence relations), enabling theorem-driven reasoning in downstream geometry solvers.
- **Symbolic Parsing.** The *SymbolicParsing* tools build fully symbolic problem representations by combining diagram parsing with formal language construction. *Inter-GPS* parses both the problem text and the associated diagram into a unified formal language and then applies theorem-based symbolic reasoning step by step to solve geometry problems. *DiagramFormalizer* provides complementary parsing endpoints that expose *ImgCDL/ConsCDL* representations directly, supplying structured inputs for interpretable geometry problem solvers such as *Inter-GPS*.

Math Solvers. The solver layer performs domain-specific high-level reasoning on top of structured perceptual inputs:

- **Multimodal.** *G-LLaVA* is a domain-specific model that integrates a CLIP-based vision encoder with *DeepSeekMath-RL* to bridge the gap between visual perception and complex mathematical reasoning. It executes geometry-aware multimodal reasoning by aligning figure interpretation with textual reasoning through specialized instruction tuning.
- **Math.** *MultiMath* is a specialized multimodal model for

geometric problem solving, optimized through geometric cross-modal alignment and instruction tuning to accurately interpret figures and relationships. It bridges visual perception and symbolic mathematics by combining a CLIP-based vision encoder with DeepSeekMath-RL, tackling complex multimodal math problems end-to-end.

C. Baseline Details

We include additional details of baseline tool-integrated or reasoning-augmented VLMs as follows:

- **VTool-R1** [73] integrates multimodal tool invocation directly into the VLM reasoning loop via reinforcement learning. By optimizing for feedback signals during problem-solving, it surpasses standard prompting and supervised fine-tuning baselines in complex visual reasoning tasks.
- **Perception-R1** [83] utilizes RL to learn active perception policies that optimize how VLMs attend to and interpret visual inputs. This approach moves beyond passive image encoding, fostering decision-oriented visual understanding through reward-driven refinement.
- **R1-VL** [85] employs GRPO to enhance step-by-step visual reasoning. By incentivizing effective intermediate reasoning steps rather than solely focusing on outcomes, it aligns visual perception with logical inference throughout the generated trajectory.
- **R1-OneVision** [81] establishes a framework for converting visual inputs into structured, cross-modal formal representations. By leveraging these symbolic intermediates during reinforcement learning, the model achieves generalized reasoning capabilities across heterogeneous task types.
- **LLaVA-OneVision** [1] proposes a unified training paradigm that consolidates diverse visual tasks under a consistent interface. This design enables a single general-purpose VLM to generalize across wide-ranging visual domains without requiring task-specific adaptation.

D. Prompt Templates

We include the prompt template in Figure 8.

E. Additional Related Works

RL for VLM Reasoning. Improving the reasoning capabilities for VLMs has been an active research front. [76] synthesize reasoning chains via distilling reasoning knowledge from teacher models. Inspired by the success of R1-style training [18], several works attempt to leverage RL [22, 31, 34, 37, 60] to improve VLM reasoning capabilities on visual tasks, including general visual understanding and mathematical reasoning.

System Prompt:

You are an advanced AI agent capable of complex reasoning and tool usage. You must strictly adhere to the following protocol for every interaction:

1. ALWAYS call the appropriate tool first;
2. NEVER provide answers without tool results;
3. Call appropriate tools based on the task; `{tool_descriptions}`
4. Reasoning Before Action: before selecting a tool, you must analyze the user’s request and determine the necessary steps. Output your internal monologue and logic inside `<think>` and `</think>` tags;
5. Tool Execution: If a tool is required, generate the tool call immediately after your reasoning. Use the following standard JSON format wrapped in XML tags: `<tool_call>{"tool": "<tool_name>", "task": "<task_name>"}`
6. Reasoning After Action: Once you receive the output from a tool, you must analyze the results to determine if further actions are needed or if the task is complete. Output this analysis inside `<think>` and `</think>` tags;
7. Final Output: When you have formulated your conclusion, you must wrap your final answer in `<answer>` and `</answer>` tags.

User Prompt:

Please answer the following `{task_type}` question:

Question: `{Question}`

Please provide a complete step-by-step solution to this problem. Your reasoning should:

1. Analyze the problem systematically
2. Check if the tool execution and answer are correct
3. If there are errors, explain what went wrong and provide the correct reasoning
4. Provide the final answer

Use natural expressions like 'let me think' or 'hmm' when helpful, but keep it concise. It’s encouraged to use self-reflection or verification especially in the verifying tool output in the reasoning process.

Provide your detailed reasoning between `<think>` and `</think>` tags, then give your final answer between `<answer>` and `</answer>` tags.

Output format: `{output_format}`

Figure 8. Prompt template.

F. Additional Method Details of VISTA-Gym

Interface Details. VISTA-Gym follows the Gymnasium API with core methods `reset` and `step`. On `reset`, the environment samples a task $x \in \mathcal{I}$ (question + image) and

Table 5. Effect of reasoning and tool-use results without RL training. *w/ Tools* refers to directly augmenting VLMs with tools significantly degrades accuracy; *w/ Reasoning* refers to CoT reasoning without tools. *w/ T&R* refers to a different setting compared to TIR, only supplying *tool-selection prior knowledge* and interleaving reasoning with tool execution improve performance.

Baselines (\downarrow)	In-Distribution						Out-of-Distribution							All
Datasets (\rightarrow)	ChartQA	Geometry3K	GeoQA	UniGeo	MapQA	avg.	TABMWP	AI2D	PlotQA	CLEVR-Math	IconQA	MathVista	avg.	avg.
GPT-5	85.92	94.84	90.91	49.34	60.95	76.39	99.90	86.10	72.90	26.00	87.20	80.20	75.38	75.84
w/ Tools	58.60	74.04	65.00	28.78	31.20	51.52	69.83	65.07	46.75	18.96	60.53	42.60	50.62	51.03
w/ Reasoning	85.68	91.01	86.27	51.25	59.48	74.74	99.81	83.03	84.73	25.50	86.27	80.90	76.71	75.81
w/ T&R	94.00	93.84	91.56	51.69	57.35	77.69	98.92	84.62	79.45	27.79	91.07	57.80	73.28	75.28
InternVL3-8B	77.32	47.09	44.87	33.71	34.70	47.54	95.32	74.54	63.05	28.95	73.27	59.60	65.79	57.49
w/ Tools	25.44	17.06	27.27	26.90	16.22	22.58	58.15	51.74	58.31	18.01	25.86	22.70	39.13	31.61
w/ Reasoning	81.28	47.09	45.07	35.81	31.10	48.07	94.38	76.51	65.85	24.15	75.53	60.70	66.19	57.95
w/ T&R	68.56	35.64	55.90	27.06	26.45	42.72	92.34	50.45	38.70	28.18	80.07	29.10	53.14	48.40

returns the initial observation. At time step t , given the interaction history c_{t-1} , the environment provides the current observation o_t , and the agent produces a thought-action pair (g_t, a_t) . The environment executes a_t , transitions according to the POMDP formalization in Section 3, and yields the next observation o_{t+1} , enabling multi-turn tool-integrated interaction.

G. Additional Method Details of VISTA-R1

Image Additional Design. Integrating the InternVL3 [88] family into our reinforcement learning framework requires a bespoke adaptation of the visual processing pipeline, as its composite architecture differs substantially from monolithic models such as Qwen2.5-VL [3]. Instead of performing early tensor conversion, we preserve raw `<image>` placeholders in the prompt and rely on InternVL’s native processor to expand them into 256 `<IMG_CONTEXT>` tokens and inject the corresponding visual embeddings during the forward pass. To ensure stability in distributed training, we customize Ray and Fully Sharded Data Parallel (FSDP) to wrap only the language decoder layers, leaving the vision encoder unsharded for memory efficiency, and we adjust attention masks and position IDs in VLLM to accommodate the extended visual token sequence. This minimally invasive interface adaptation preserves InternVL’s native visual processing while enabling robust, efficient multimodal GRPO training.

H. Additional Implementation Details

H.1. Detailed Error Types for Error Analysis

We define the detailed error types as follows:

- **E1: Invocation schema violation (wrong function-call structure).** The model produces an invalid tool call that violates the prescribed schema, such as missing required fields, extraneous keys, incorrect nesting, or non-JSON-conformant structures that prevent execution.
- **E2: Invalid argument name (wrong argument key).** The tool call structure is syntactically valid, but one or more argument *names* do not match the tool specification

(*e.g.*, using `"x_axis"` instead of `"x_label"`), causing the call to be rejected by schema validation.

- **E3: Invalid argument value (wrong argument format).** Argument names are correct, but the *value types or formats* are invalid, such as providing strings where numbers are required, out-of-range values, or malformed lists that violate the tool’s input constraints.
- **E4: Incorrect argument value (wrong argument content).** The tool call is syntactically valid and passes type checks, but the *semantic content* of one or more arguments is wrong (*e.g.*, selecting the wrong region, axis, or series for analysis), leading the tool to operate on an incorrect target.
- **E5: Invalid output from tool execution (wrong answer format).** The tool executes but returns an output that does not conform to the expected format for downstream use, such as missing required fields, malformed JSON, or values that cannot be parsed into the canonical answer representation.
- **E6: Incorrect reasoning from tool execution.** The tool output is valid and informative, but the model fails to map it to the correct final answer, due to faulty logical deductions, misinterpretation of intermediate results, or inconsistent multi-step reasoning.

H.2. Additional Training Setup Details

We configure the training environment with a maximum of three turns per interaction, balancing exploration depth and computational cost. The agent retains full access to the conversation history and tool interaction logs across turns, and up to 24 trajectories are processed concurrently to maximize throughput. Agent-side code execution is implemented in Python 3.10. Training uses 8 NVIDIA GPUs with the FSDP2 (Fully Sharded Data Parallel) strategy for efficient distributed optimization. We set parameters to temperature 0.7. The pipeline integrates a tool-router service for real-time tool execution and feedback, and employs asynchronous rollouts with a batch size of 128 to maintain high GPU utilization. We consider four different backbones for VISTA-R1 with varying sizes, including InternVL3-2B/8B/14B and Qwen2.5-VL-7B, in the main experiment.

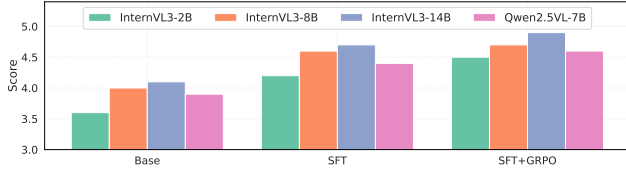


Figure 9. Human study on trajectory quality.

We perform ablation experiments and additional analysis with InternVL3-8B as the VLM. For SFT, we train for one epoch with a batch size of 128, a learning rate of 2×10^{-6} , and a context length of 16,384 tokens. For RL, we use a micro-batch size of 8 per GPU, a mini-batch size of 128, and $G = 8$ rollouts per update. We set the regularization coefficient $\beta = 10^{-3}$, cap the maximum response length at 26,780 tokens, and optimize with a learning rate of 5×10^{-7} for $E = 300$ steps.

I. Additional Experimental Results

I.1. Effect of Tool and Reasoning in Vanilla VLMs

Table 5 summarizes the effect of tools and reasoning on vanilla VLMs for open- and closed-source models. Directly augmenting VLMs with tools significantly degrades accuracy, and intrinsic chain-of-thought alone yields only modest gains on complex VQA. Supplying *tool-selection priors* improves performance; however, gains are strongly task-dependent for commercial VLMs, and smaller open-source VLMs remain particularly challenged. These observations underscore that it is both important and non-trivial to endow VLMs with robust, generalizable tool-use capabilities that adapt across diverse visual reasoning scenarios.

I.2. Human Study for Trajectory Quality

As TIR trajectories are a key component of VISTA-Gym, we evaluate the quality of the generated interleaved tool-use and reasoning steps. Figure 9 reports average human ratings (1–5 scale) over 40 randomly sampled solution trajectory per dataset. We observe that both training stages significantly improve TIR across tasks and model sizes.