

Waking Up Blind: Cold-Start Optimization of Supervision-Free Agentic Trajectories for Grounded Visual Perception

Ashutosh Bajpai^{1,*}, Tamal Majumder^{1,*}, Akshay Nambi³, Tanmoy Chakraborty^{1,2}

¹Indian Institute of Technology Delhi, India

²Indian Institute of Technology Abu Dhabi, UAE

³Microsoft Research, India

{eez228482,tanchak}@ee.iitd.ac.in, 987tamal@gmail.com,
akshayn@microsoft.com

Abstract

Small Vision-Language Models (SVLMs) are efficient task controllers but often suffer from visual brittleness and poor tool orchestration. They typically require expensive supervised trajectory tuning to mitigate these deficits. In this work, we propose **Self-supervised Perception Enabled by Cascaded Tool Roll-out Alignment (SPECTRA)**, a *supervision-free framework* that bootstraps agentic capabilities via Coldstart Reinforcement Learning for SVLMs. SPECTRA enforces *Soft Structured Multi-turn Rollouts*, a topological constraint that directs agents to explicitly sequence tool-derived evidence before synthesis, effectively grounding reasoning in visual observations. We employ a multi-objective reward signal that simultaneously maximizes task correctness, roll-out structure, and tool utility, enabling agent to self-discover robust behaviors without human preference labels. We further introduce **Tool Instrumental Utility (TIU)**, a novel metric to quantify tool efficacy in the absence of ground truth. Extensive evaluations across composite and out-of-distribution (MMMU-Pro) benchmarks demonstrate that SPECTRA boosts agentic trajectories, improving task accuracy by up to 5% and tool efficiency by 9%, enabling more efficient multimodal agents that learn effectively from environmental interaction alone.

1 Introduction

SVLMs As Autonomous Controllers. Foundation models are rapidly transitioning from passive reasoning to agentic systems that perceive, reason, and act across multimodal, multi-step tasks (Bommasani et al., 2022; Yang et al., 2023). Within this shift, small vision-language models (SVLMs) such as Qwen2.5-VL (Bai et al., 2025) are attractive controllers due to favorable latency and deployment cost, yet they lag larger VLM controllers

in long-horizon reasoning and suffer from weak fine-grained visual perception and inefficient tool orchestration (Yao et al., 2024; Wang et al., 2024b; Tong et al., 2024; Guan et al., 2024). Prior work shows that stepwise reasoning with tool calls can be scaffolded by prompting frameworks like ReAct (Yao et al., 2023) and that VLM-driven agents can outperform LLM-only controllers when visual context determines which tools to call (Gao et al., 2025; Zheng et al., 2024a). Despite these advances, SVLMs still struggle to decompose multimodal problems into reliable stepwise trajectories that balance perception, reasoning, and tool usage under data scarcity (Li et al., 2025).

Recent advancements in agentic trajectories optimization. Two approaches work to close the aforementioned gap. First, trajectory tuning with verifiable synthetic tool-use data improves VLM controllers’ stepwise tool reasoning (LLaVA-Plus (Liu et al., 2023), MLLM-Tool (Wang et al., 2025)), with T3-Agent introducing MM-Traj, demonstrating 20% gains on GTA (Wang et al., 2024a) via ReAct-style trajectory tuning of MiniCPM-V-8.5B and Qwen2-VL-7B (Gao et al., 2025). Larger-scale resources extend this direction, with evaluators increasingly assessing trajectory quality rather than focusing solely on final answers (Anonymous, 2025). Second, Reinforcement Learning (RL) for tool-augmented agents emphasizes sample efficiency and adaptive perception (e.g., Tool-R1 (Zhang et al., 2025a)), while tool integration and memory mechanisms (e.g., AutoTool (Zou et al., 2025), ToolMem (Xiao et al., 2025)) reduce redundant calls and better align tools with task demands.

Limitations of trajectory tuning. Despite aforementioned advances, most systems either (i) tune for tool-use reasoning without directly refining visual perception via structured rollouts, or (ii) apply supervised data-driven mechanisms to align tool efficiency, visual evidence acquisition, and step-

*Equal Contribution.

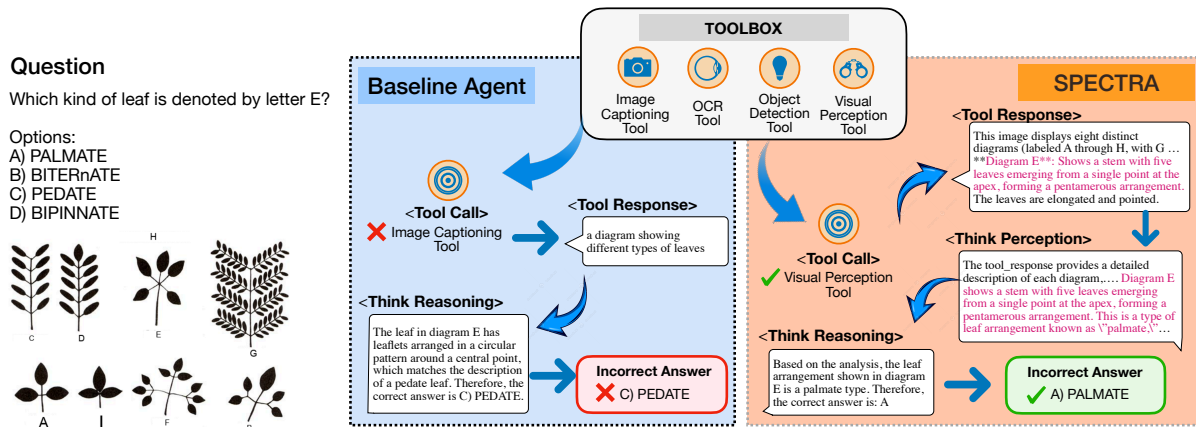


Figure 1: Comparison of the Qwen-2.5-VL-7B based SPECTRA with a baseline agent, showcasing its efficient tool utilization and visually-guided structured approach, resulting in accurate outcomes.

wise reasoning, especially relying on expensive synthetically generated data.

We investigate how tool and structured rollouts-mediated visual perception refinement mechanisms can enhance SVLMs’ agentic capabilities, enabling better task decomposition into stepwise trajectories and bridging task-specific tool alignments in absence of supervised tool trajectories. We pose two research questions:

- **RQ1:** How can SVLMs’ agentic capabilities be augmented by improving tool-calling efficiency via coldstart RL, i.e., learning from environment signals without human preference models?
- **RQ2:** How can improving visual perception through soft structured rollouts address SVLMs’ weak visual perception?

To address these questions, we introduce the **Self-supervised Perception Enabled by Cascaded Tool Rollout Alignment (SPECTRA)**. Unlike traditional approaches dependent on supervised demonstrations, SPECTRA optimizes an SVLM policy directly from environmental interactions using a multi-objective reward signal that simultaneously maximizes task correctness, rollout structure, and instrumental tool utility via Group Relative Policy Optimization (GRPO) strategy. We enforce a hierarchical reasoning topology, *Soft Structured Multi-turn Rollouts*, which compels the agent to explicitly sequence tool selection, observation integration, and perceptual synthesis before answering. By employing a multi-objective reward signal that balances task correctness with structural integrity and tool diversity, SPECTRA enables the model to self-discover robust agentic behaviors and ground

its visual perception in tool-derived evidence, effectively bypassing the need for human preferred tool trajectories or reasoning traces often distilled from large models. Moreover, Tool Accuracy (Gao et al., 2025) is commonly used to evaluate tool trajectories. However, limited attention has been given to evaluating tool correctness or efficiency in absence of supervised trajectories.

To address this limitation, we propose a novel metric, **Tool Instrumental Utility (TIU)**, which combines tool reliability, alignment, and selectivity to offer a comprehensive measure. We release SPECTRA in 7B and 3B variants, both utilizing the Qwen-2.5-VL. SPECTRA boosts agentic trajectories, improving task accuracy by up to 5% and tool efficiency by 9% over closest baseline across four multimodal benchmarks. An illustration of SPECTRA is presented in Figure 1.

The primary contributions of this work are summarized as follows¹:

- We introduce soft structured multi-turn topological (rollout) constraints to enforce focused and grounded stepwise visual reasoning.
- We propose SPECTRA, a novel supervision-free agentic policy optimization for tool trajectories and visual reasoning traces.
- We propose TIU, a novel metric to measure tool efficiency in the absence of supervised tool trajectory preferences.
- Empirical studies with qualitative analysis demonstrate that SPECTRA outperform strong baselines in enhancing both task accuracy and agent trajectories.

¹Source code and dataset are available at <https://github.com/ab-iitd/spectra>

2 Related Works

Multimodal agents and controllers. Early multimodal agents utilized large language models (LLMs) to plan and execute tool calls via textual interfaces, often employing ReAct-style reasoning or prompting frameworks (Surís et al., 2023; Gupta and Kembhavi, 2022; Shen et al., 2023; Yao et al., 2023; Yang et al., 2023). However, relying solely on textual queries can hinder tool selection when visual context is critical (Fan et al., 2024; Wu et al., 2023; Trivedi et al., 2024). To mitigate this, VLM-driven agents integrate the controller within a multimodal model, allowing direct conditioning on visual inputs to improve efficiency in vision-centric tasks (Trivedi et al., 2024; Zheng et al., 2024b; Wang et al., 2024c; Zheng et al., 2024c). Addressing the weaker reasoning capabilities of smaller VLMs, recent approaches finetune models on tool-usage trajectories (Liu et al., 2023; Wang et al., 2025). Notably, T3-Agent (Gao et al., 2025) leverages automatically generated trajectories, demonstrating substantial improvements on benchmarks like GTA and GAIA (Mialon et al., 2023).

RL for decision-making VLM agents. Interactive RL offers a pathway to teach agents multi-step decision-making using environmental feedback. While RLHF aligns LLMs with human preferences (Ouyang et al., 2022; Stiennon et al., 2020), task-specific RL has proven effective in text-only domains (Ramamurthy et al., 2023; Zhou et al., 2024). Extending this to VLMs, RL4VLM (Zhai et al., 2024) fine-tunes a 7B model using environmental rewards on tasks requiring visual and linguistic understanding. Zhang et al. (2025b) expanded on this to introduce sample-efficient RL for tool use. Despite training instability challenges in GRPO (Deng et al., 2025), these studies confirm that RL can effectively equip SVLM controllers (Schulman et al., 2017; Haarnoja et al., 2018).

Tool-usage datasets and trajectory tuning. Agents depend on high-quality tool-usage data capturing queries, multimodal files, thoughts, actions, and observations. Early tool datasets, such as APIBank (Li et al., 2023b); ToolAlpaca (Tang et al., 2023) ToolBench (Qin et al., 2024); AnyTool (Du et al., 2024); AgentOhana (Zhang et al., 2024); APIGen (Liu et al., 2024); AgentInstruct (Mittra et al., 2024), focus primarily on text tools, with limited multimodal coverage. Benchmarks such as OS-World (Xie et al., 2024), MMInA (Tian et al., 2025), AgentBench (Liu et al., 2025), and AgentGym (Xi

et al., 2024) incorporate multimodality, but datasets for training tool-augmented VLMs at scale are scarce. MLLM-Tool (Wang et al., 2025) and T3-Agent (Gao et al., 2025) overcome the aforementioned challenge by advancing tool-trajectories data generation methods. Current methods for improving perception and tool efficiency in multimodal agents predominantly rely on supervised signals from curated tool trajectories and structured reasoning traces. However, acquiring such high-quality supervision at scale is costly and limits generalization. To overcome these challenges, we propose SPECTRA, a novel method that leverages structured rollout and unsupervised cold-start RL to enhance agentic trajectories without reliance on expensive ground-truth annotations.

3 Dataset and Benchmarks

To foster robust agentic reasoning in SVLMs, we consider four public distinct domains datasets: AI2D (Kembhavi et al., 2016), TQA (Kim et al., 2019), ScienceQA (Lu et al., 2022), and OK-VQA (Marino et al., 2019). These benchmarks are selected to ensure broad coverage of multi-modal capabilities, ranging from diagrammatic parsing and textbook-length comprehension to external knowledge retrieval and chain-of-thought reasoning. We use 1,000 training and 200 test samples per dataset (4,000 and 800 total) to ensure sample-efficient learning. For rigorous out-of-distribution evaluation, we employ single-image MMMU-Pro (Yue et al., 2025), a challenging 30-subject university-level benchmark for testing integrated visual-textual reasoning (Appendix A).

4 Methodology

We introduce **Self-supervised Perception Enabled by Cascaded Tool Rollout Alignment (SPECTRA)** (Figure 2), to augment SVLMs tool-use and visual grounding by using domain-specialized GRPO to learn directly from environmental signals, bypassing the need for supervised demonstrations.

4.1 Problem Formulation

We formulate the agentic reasoning task similar to a Partial Observation Markov Decision Process (POMDP) (Åström, 1965). Let the multimodal input space be defined by tuples $(I, q) \sim \mathcal{D}$, where I is the visual context (image) and q is the natural language query.

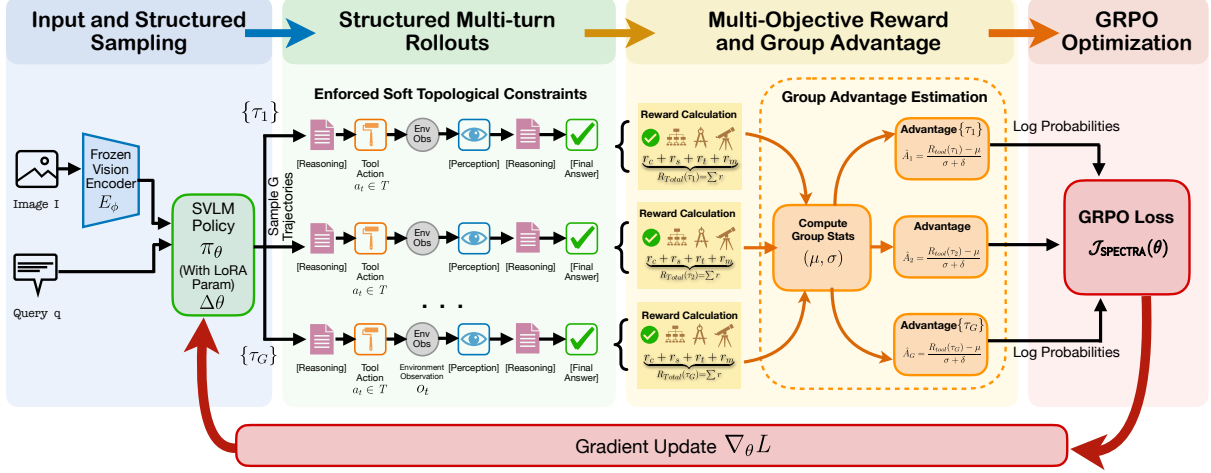


Figure 2: Overview of SPECTRA. The pipeline samples structured multi-turn trajectories from the SVLM policy, aggregates multi-objective rewards to compute group-relative advantages, and iteratively optimizes the policy parameters via GRPO-based objective, where $r_c : R_{corr}$, $r_s : R_{struct}$, $r_t : R_{tool}$, $r_m : R_{term}$ are reward components for task accuracy, structural integrity, tool efficiency, and terminal delimitation, respectively.

Policy Architecture. The agent is parameterized by an SVLM policy π_θ . The architecture utilizes a frozen vision encoder E_ϕ and a trainable LLM decoder adapted via Low-Rank Adaptation (LoRA). We optimize only the adapter parameters $\Delta\theta$, keeping the base vision-language alignment fixed to preserve general knowledge. The policy output at time step t is given by:

$$\pi_\theta(a_t|s_t) = \text{Softmax}(W_{\text{frozen}}h_t + BAh_t) \quad (1)$$

where W_{frozen} represents the fixed pre-trained weights, $B, A \in \mathbb{R}^{d \times r}, \mathbb{R}^{r \times d}$ are the low-rank learnable matrices, h_t is latent representation of input, and a_t denotes action at time t given state s_t .

Action Space. The action space \mathcal{A} is a union of natural language tokens and a discrete set of tool primitives $T = \{T_{\text{cap}}, T_{\text{det}}, T_{\text{ocr}}, T_{\text{vp}}\}$, corresponding to Image Captioning, Object Detection, OCR, and Visual Perception respectively. The detailed description of tools are provided in Table 4.

4.2 Soft Structured Multi-turn Rollouts

Drawing inspiration from Xue et al. (2025), we address the “blindness” often observed in SVLMs by enforcing a hierarchical reasoning structure. An optimal trajectory τ must adhere to the following topological sequence:

$$\tau = \langle \text{reason} \rightarrow \text{tool} \rightarrow \text{obs} \rightarrow \text{percep} \rightarrow \text{reason} \rightarrow \text{ans} \rangle \quad (2)$$

This structure forces the model to explicitly reason about *which* tool to use, integrate the tool’s raw

output (Observation), synthesize that output with visual features (Perception) and re-reason before committing to a final answer (Figure 3).

4.3 Cold-Start Agentic Optimization

We introduce a specialized optimization objective that enables the SVLM to self-explore valid tool-usage patterns without human reference trajectories. We employ a GRPO style policy gradient with group-normalized advantages (Shao et al., 2024). Unlike standard RLHF, which constrains the policy to a semantic reference, our “Cold-Start” setting requires the model to significantly deviate from its initial distribution to discover agentic behaviors. Therefore, we rely on group-relative normalization to stabilize the learning of diverse reasoning paths.

For each multimodal input (I, q) , we sample a group of G distinct cognitive rollouts $\{\tau_1, \tau_2, \dots, \tau_G\}$ from the current policy $\pi_{\theta_{\text{old}}}$. These trajectories represent competing reasoning strategies for the same visual stimulus. The GRPO objective for SPECTRA is formulated as:

$$\mathcal{J}_{\text{SPECTRA}}(\theta) = \mathbb{E}_{(I, q) \sim \mathcal{D}, \{\tau_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}} \left[\frac{1}{G} \sum_{i=1}^G \frac{1}{|\tau_i|} \sum_{t=1}^{|\tau_i|} \min(\rho_{i,t} \hat{A}_{i,t}, \text{clip}(\rho_{i,t}, 1 - \epsilon_l, 1 + \epsilon_h) \hat{A}_{i,t}) \right] - \psi D_{\text{KL}}(\pi_\theta \parallel \pi_{\theta_{\text{ref}}}) \quad (3)$$

where $\rho_i = \frac{\pi_\theta(\tau_i|I, q, \tau_{i, < t})}{\pi_{\theta_{\text{old}}}(\tau_i|I, q, \tau_{i, < t})}$ represents the probability ratio. \hat{A}_i is the **Trajectory Advantage**, which

Definition: Multi-Turn Structured Rollout (τ)

```
1. Reasoning Phase:
<think_reasoning> Analyzes query complexity and plans the next
step. </think_reasoning>
2. Tool Execution:
<tool_call> Invokes module  $T_i \in T$  with arguments
</tool_call>
3. Observation Integration:
<tool_response> Ingests the raw external tool output.
</tool_response>
4. Perceptual Re-alignment:
<think_perception> Synthesizes tool output with visual features.
</think_perception>
... [Iterative Process] ...
5. Conclusion:
<answer> Generates the final prediction. </answer>
```

Figure 3: Multi-turn structured rollout template.

standardizes the dense multi-objective reward R_{total} against the group’s baseline performance:

$$\hat{A}_i = \frac{R_{\text{total}}(\tau_i) - \mu(\{R_{\text{total}}(\tau_j)\}_{j=1}^G)}{\sigma(\{R_{\text{total}}(\tau_j)\}_{j=1}^G) + \delta} \quad (4)$$

Here, μ and σ act as a dynamic, input-specific baseline computed over sampled group, effectively filtering out the inherent difficulty of the visual query I and isolating the quality of the tool-usage strategy.

4.4 Multi-Objective Agentic Reward

The optimization is driven by $R_{\text{total}}(\tau)$, a scalarized signal that decomposes the agent’s behavior into correctness, structural integrity, and tool efficiency:

$$R_{\text{total}}(\tau) = \lambda_1 R_{\text{corr}} + \lambda_2 R_{\text{struct}} + \lambda_3 R_{\text{tool}} + \lambda_4 R_{\text{term}} \quad (5)$$

Where λ^* denotes the reward component weight.

Task Correctness (R_{corr}). A sparse signal indicating if the final answer y_{pred} matches the ground truth y_{gt} :

$$R_{\text{corr}}(\tau) = C1 \cdot \mathcal{K}(y_{\text{pred}} = y_{\text{gt}}) \quad (6)$$

where $C1 \in \mathbb{Z}$ represents the reward magnitude and \mathcal{K} is the binary indicator function.

Hierarchical Structural Integrity (R_{struct}). To enforce the *Soft Structured Multi-turn Rollout* (Eq. 2), we gradually penalize deviations from the optimal reasoning topology to ensure other valid paths are not marginalized. We define a hierarchy mapping $\phi(\tau)$, where $\phi = 0$ represents a complete agentic loop. The reward decays as the structure degrades from optima:

$$R_{\text{struct}}(\tau) = \alpha \cdot \gamma^{\phi(\tau)} \quad (7)$$

where $\alpha = 2.0$, $\gamma = 0.75$, and $\phi(\tau)$ map partial structures (e.g., missing perception) to higher integers. Please refer Appendix C.1 for more details.

Tool Utility (R_{tool}). This component ensures the agent generates executable tool calls and avoids hallucinating non-existent tools. The total tool utility is the sum of a syntactic validity reward, an execution success reward, and a diversity bonus:

$$R_{\text{tool}}(\tau) = \mathcal{K}_{\text{syntax}}(\tau) + \mathcal{K}_{\text{success}}(\tau) + R_{\text{div}}(\tau) \quad (8)$$

where $\mathcal{K}_{\text{syntax}}$ and $\mathcal{K}_{\text{success}}$ are binary indicators ensuring the tool call follows the correct format and returns a successful response (non-error), respectively. To prevent mode collapse (e.g., relying solely on OCR) and to penalize infinite loops of tool invocations, the diversity reward R_{div} incentivizes the use of unique tools subject to two constraints: a per-tool saturation cap κ and a global diversity cap η :

$$R_{\text{div}}(\tau) = \min \left(\eta, \sum_{T_k \in \mathcal{T}} \beta \cdot \min(\text{count}(T_k, \tau), \kappa) \right) \quad (9)$$

Here, β is the per-tool reward and η limits the total diversity reward to prevent reward hacking via excessive tool calls.

Terminal Delimitation (R_{term}). A structural anchor ensuring the reasoning process converges to a definitive answer token (<answer>), preventing infinite reasoning loops:

$$R_{\text{term}}(\tau) = C2 \cdot \mathcal{K}(\langle \text{answer} \rangle \in \tau) \quad (10)$$

where $C2 \in \mathbb{Z}$ represents the reward magnitude.

Finally, to achieve stabilized advantage updates, we normalize and scale the total reward function. As a result, the final reward function is:

$$R_{\text{Total}}(\tau) = S \times \frac{R_{\text{total}}(\tau)}{N_{\text{norm}}} \quad (11)$$

where $N_{\text{norm}} = |\max(R_{\text{total}})|$ is the normalization factor, which is the maximum possible reward allowed, and S is the scaling factor. The nomenclature of all notations is detailed in Appendix J.

5 Evaluation Measures

5.1 Task Accuracy

We evaluate and report the specific task accuracies within the context of MCQ-style Q/A setup.

Model	AI2D	TQA	OK-VQA	SCIENCE-QA	Avg. (In Dist.)	MMMU-Pro
Closed-Source Models						
GPT-4o	76.5	77.0	88.5	86.0	82.0	61.8
GPT-4o Mini	64.0	73.5	78.5	83.5	74.9	51.9
Base Models (Non-Agentic)						
Qwen2.5-VL [3B]	42.2	42.2	44.5	42.0	42.7	32.7
Gemma-3-IT [4B]	68.0	73.0	74.0	73.0	72.0	42.0
Phi-4-Multimodal [6B]	50.0	52.0	65.0	73.5	60.1	41.2
Qwen2.5-VL [7B]	63.8	74.6	71.5	73.5	70.9	40.5
Baseline (Agentic)						
VERL: Qwen2.5-VL [3B]	56.5	56.0	70.5	58.0	60.3	33.8
Custom: Gemma-3-IT [4B]	59.5	68.0	74.0	72.0	68.4	30.2
Custom: Phi-4-Multimodal [6B]	19.5	20.0	19.0	26.5	20.1	16.9
HF SmolAgent: Qwen2.5-VL [7B]	66.0	72.5	76.0	79.0	73.4	36.9
VERL: Qwen2.5-VL [7B] \uparrow	67.5	73.3	74.6	78.3	73.4	44.3
SPECTRA (Ours)						
VERL: Qwen2.5-VL [3B]	60.5	60.0	66.5	68.5	63.9	36.3
VERL: Qwen2.5-VL [7B]	71.1	77.5	79.6	83.1	77.8	46.7

Table 1: Benchmark comparison of **SPECTRA** over various baselines (Agentic and Non-Agentic Zero-shot), including out-of-distribution MMMU-Pro for both the variants: Qwen2.5-VL (3B and 7B). Where, \uparrow denotes closest baseline.

5.2 Tool Instrumental Utility (TIU)

To rigorously evaluate the efficacy of tool use in agentic models under self-supervision, we define an evaluation set $\mathcal{D}_{\text{eval}}$ and utilize the library of available tool primitives T (as defined in Section 4). We propose *Tool Instrumental Utility* (TIU) as a composite metric to evaluate the overall effectiveness of tool use in the absence of human preferences. TIU integrates reliability, alignment, and selectivity into a normalized scalar $[0, 1]$. It is defined as:

$$\text{TIU} = \text{TER} \times \underbrace{\left(\frac{1 + \text{TTAC}}{2} \right)}_{\text{Normalized Relevance}} \times \underbrace{\tanh(\text{TSS})}_{\text{Bounded Intentionality}} \quad (12)$$

The term $\frac{1 + \text{TTAC}}{2}$ normalizes the alignment coefficient to $[0, 1]$ for the metric Task-Tool Alignment Coefficient (TTAC), penalizing detrimental tool use. The hyperbolic tangent $\tanh(\text{TSS})$ bounds the infinite range of the KL-divergence based Tool Selectivity Score (TSS), rewarding selectivity with diminishing returns. A TIU score of 1.0 represents an agent that is perfectly reliable, strategically selective, and uses tools that guarantee task success. We describe the components of this metric below:

Tool Execution Reliability (TER). TER measures the mechanical robustness of the agent, specifically the rate of syntactically valid and error-free tool invocations. Let N be the total number of tool calls attempted across $\mathcal{D}_{\text{eval}}$. It is defined as:

$$\text{TER} = \frac{1}{N} \sum_{i=1}^N \mathbb{1}(o_i = \text{success}) \quad (13)$$

where $\mathbb{1}$ is the indicator function and o_i represents the execution outcome of the i -th tool call.

Task-Tool Alignment Coefficient (TTAC). To assess instrumental relevance, we quantify the relationship between tool usage and task success. TTAC is calculated as the average point-biserial correlation between the binary usage vector \mathbf{u}_T (indicating if tool T was used) and the task success vector \mathbf{y}_{succ} (derived from R_{corr}):

$$\text{TTAC} = \frac{1}{|T|} \sum_{T_k \in T} r(\mathbf{u}_{T_k}, \mathbf{y}_{\text{succ}}) \quad (14)$$

Here, positive values of Pearson correlation coefficient $r \in [-1, 1]$ indicate that tool T usage correlates with successful task resolution.

Tool Selectivity Score (TSS). TSS measures the agent’s intentionality by quantifying the deviation of its empirical tool usage distribution P from a random uniform distribution U . We utilize the Kullback-Leibler (KL) divergence:

$$\text{TSS} = D_{KL}(P||U) = \sum_{T_k \in T} P(T_k) \log \left(\frac{P(T_k)}{U(T_k)} \right) \quad (15)$$

Higher values indicate strong preferences for strategic selection of tools within the action space \mathcal{A} , while values approaching 0 indicate random guessing. Please refer to Appendix C.2 for more details.

6 Experimental Results

Experimental Setting. We implement SPECTRA by fine-tuning Qwen2.5-VL (3B and 7B) (Bai

Model	Dataset	TER (%)	TTAC	TSS	TIU (%)
Baseline Agent	AI2D	77.08	-0.051	1.32	31.77
	TQA	78.92	0.078	2.96	42.34
	OK-VQA	80.13	0.009	2.78	40.12
	SCIENCE-QA	73.08	-0.049	1.14	28.29
	Mean	77.30	-0.003	2.05	35.63
SPECTRA	AI2D	85.71	-0.025	3.08	41.59
	TQA	90.07	0.081	3.04	48.47
	OK-VQA	85.16	-0.100	2.76	37.99
	SCIENCE-QA	93.84	0.082	3.05	50.58
	Mean	88.69	0.009	2.98	44.66

Table 2: Tool efficiency enhancement in SPECTRA 7B over baseline agent across datasets. (TER: Tool Execution Reliability, TTAC: Tool-Task Alignment Coefficient, TSS: Tool Selectivity Score, and TIU: Tool Instrumental Utility; a composite metric)

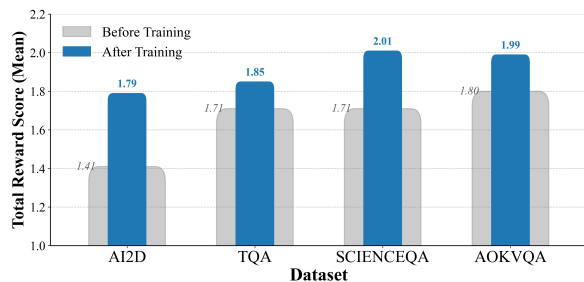


Figure 4: Average total reward progression across datasets after applying SPECTRA’s (7B) optimization.

et al., 2025) using the VERL framework (Sheng et al., 2024) with the vLLM engine (Kwon et al., 2023). We apply LoRA adapters (Hu et al., 2021) to the language model while freezing the vision encoder. We evaluate performance against strong baselines on both in-distribution and out-of-distribution (MMMU-Pro) test data to demonstrate scalability. Prompts, hyperparameters, and computational efficiency details are provided in Appendices F, I, and H, respectively.

Baselines. Our evaluation utilizes Phi-4-Multimodal-Instruct [6B] (Microsoft et al., 2025), Gemma-3-IT [4B] (Team et al., 2025), and a closed-source GPT-4o (Hurst et al., 2024). We benchmark SPECTRA against both standard (non-agentic) base models and strong agentic baselines including smolagents (Roucher et al., 2025). For rigorous qualitative assessment, we compare SPECTRA with the closest VERL-based agentic baseline using Qwen2.5-VL [7B].

6.1 Benchmark Comparison

As shown in Table 1, SPECTRA 3B and 7B consistently outperform the closest strong VERL-based agentic baseline across all benchmarks, demonstrat-

Setting	AI2D	TQA	OKVQA	ScienceQA	Avg.
R_{total}	71.17	77.5	79.67	83.17	77.8
w/o R_{corr}	68.5	78.5	80.5	77.5	76.2
w/o R_{struct}	66.0	77.5	82.5	77.0	75.7
w/o R_{tool}	74.5	74.0	79.5	78.0	76.5
w/o R_{term}	72.0	75.5	77.5	78.0	75.7

Table 3: Leave-one-out ablation study of reward components in SPECTRA 7B. Performance changes are visualized via background color: gray indicates improvement, while red indicates a drop compared to total reward.

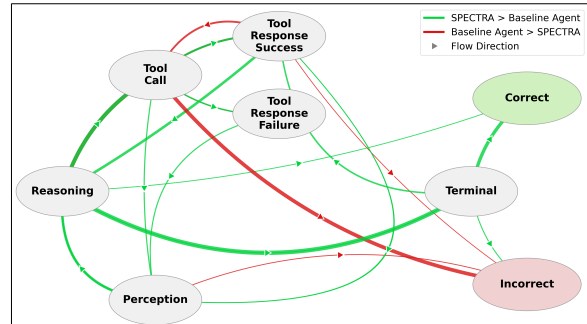


Figure 5: Differential State Transition Analysis: Baseline agent vs. SPECTRA 7B (in-distribution test set). The impactful transitions shifts are highlighted by the magnitude of color change. SPECTRA incentivizes green paths and down-scaling red paths w.r.t. baseline agent.

ing strong generalization on both in-distribution and the out-of-distribution (MMMU-Pro) datasets. Notably, SPECTRA achieves an increment of 3 to 5 percentage points across benchmarks and a substantial gain on reasoning-intensive tasks, surpassing the baseline by margins of 4.8 and 5.0 percentage points on SCIENCE-QA and OK-VQA, respectively for 7B variant. To validate the robustness of these findings, we report the mean average over three independent runs for SPECTRA 7B and the baseline; an independent two-tailed T-test, average across datasets, p -value of 0.0019 confirms the statistical significance of the finding.

6.2 Analysis of Tool Utility and Efficiency

Quantitative analysis presented in Table 2 reveals that SPECTRA significantly enhances agentic proficiency, achieving a 25.3% relative improvement in mean TIU compared to the baseline (35.63% \rightarrow 44.66%). This gain is predominantly driven by a robust 11.4% increase in TER, confirming that our topological constraints effectively minimize syntax errors and invalid calls, particularly in ScienceQA (93.84%). Furthermore, the elevated TSS of 2.99 (vs. 2.06) indicates that the policy successfully avoids mode collapse, converging on a focused subset of high-utility tools rather than ran-

Comparative Analysis: Robustness to Execution Errors

Question: In the food web, what role does grass play?

Options: A) Producer/Energy Source B) Decomposer C) Predator D) Consumer



Baseline Agent

Reasoning: Identifies grass as producer.

» Tool Call: None()

« Err: Exec tool 'None'

Crash: The agent enters a loop calling 'None' and terminates abruptly.

Final Output:

assistant\n <think ... [End of context]

Prediction: None (Failed)

SPECTRA (Ours)

Reasoning: Identifies grass as producer.

» Tool Call:{"captioning_tool"...}

(Undefined <Captioning_Tool> Tag Hallucination)

addCriterion: The image is...

[Repetitive Loop x3]

Recovery: Despite the hallucinated tool and syntax errors, it recovers to output the answer.

Prediction: A ✓

Figure 6: Robustness comparison. The **Baseline Agent** (left) crashes after a tool error, ending abruptly without an answer. **SPECTRA** (right) hallucinates an undefined tool (highlighted in red) and enters a generation loop, yet successfully recovers to provide the correct answer (A).

dom guessing. While TTAC remains a challenging objective under cold-start conditions, SPECTRA notably shifts the mean correlation from negative to positive (0.010), suggesting that self-discovered policies effectively ground visual reasoning in actionable evidence without supervised preferences.

6.3 Agentic Trajectory Analysis

We perform a differential state transition analysis in multi-turn rollout to evaluate SPECTRA against the baseline agent, highlighting shifts in error propagation and resolution efficiency. Results presented in Figure 5 demonstrate that SPECTRA significantly optimizes the trajectory toward successful outcomes, evidenced by a marked increase in Reasoning \rightarrow Terminal transitions (+48) and Reasoning \rightarrow Tool_Call accuracy (+44). Notably, the model consolidates the iterative loop Reasoning \rightarrow Tool_Call \rightarrow Tool_Response_Success \rightarrow Perception \rightarrow Reasoning as the primary success path, verifying its enhanced stability in multi-step execution. Crucially, the model actively curtails pathological behaviors observed in the baseline, specifically suppressing the recursive, error-prone Tool_Call \rightarrow Tool_Call loops (-103) and reducing direct Tool_Call \rightarrow Incorrect failures (-41). This structural shift indicates that SPECTRA dampens cyclic dependencies in the incorrect category while incentivizing linear, reasoning-driven paths to correct terminal states. The detailed quantitative results are provided in Appendix E.2.

6.4 Reward Investigation

Leave-One-Out Reward Ablation. The leave-one-out ablation study (Table 3) highlights the

effectiveness of SPECTRA's reward formulation, especially on ScienceQA, where removing any component causes a performance drop exceeding 5%. The impact of R_{tool} varies across datasets, reflecting differing levels of tool reliance in tasks such as AI2D and TQA. The structural integrity reward (R_{struct}) remains critical for preventing reasoning collapse in complex visual question answering. Overall, robust generalization requires the full multi-objective framework despite domain-sensitive sub-objectives.

Reward Convergence Findings. In Figure 4, SPECTRA demonstrates substantial and consistent performance gains across all evaluated datasets, with the most notable improvement in AI2D, where the average reward increases from 1.41 to 1.79 (+27%). This uniform uplift validates that the newly formulated multi-objective agentic reward effectively stabilizes the optimization process, allowing the model to internalize complex multimodal reasoning constraints without the high variance typical of traditional reinforcement learning across diverse tasks ranging from diagram interpretation (AI2D) to knowledge-intensive QA (TQA).

6.5 Qualitative Error Analysis

SPECTRA significantly improves over the baseline by optimizing tool selection, enforcing structured multi-turn rollouts, and ensuring convergence to terminal conditions, effectively preventing infinite loops or abrupt endings. However, as illustrated in Figure 6, the formulation is not without limitations; it occasionally exhibits intermediate instability, such as hallucinating tool tags or generating repetitive text, before arriving at the correct solu-

tion. Appendix G provides comprehensive qualitative examples of error recoveries and failures, while Appendix E.1 contains the tool-level analysis.

7 Conclusion

We propose SPECTRA, a novel supervision-free framework that uses Coldstart RL to optimize SVLM tool use and visual reasoning. By introducing multi-turn topological constraints with a multi-objective reward, SPECTRA improves grounded reasoning and mitigates perceptual limitations. We further introduce Tool Instrumental Utility, a metric for quantifying tool efficiency without supervised preferences. Experiments show that environment-driven signals combined with structured constraints significantly enhance SVLM agentic performance, pointing toward autonomous, perceptually aware agents without costly human supervision.

8 Limitations

While SPECTRA demonstrates significant advancements in visual-intensive tasks, there are two primary limitations to our current approach. First, the model is explicitly designed with a focus on vision-specific tools. Consequently, it lacks access to broader utilities such as code execution environments or search engines, which can be critical for solving complex, multi-modal vision tasks requiring external knowledge or precise calculation. Future iterations of SPECTRA could integrate these general-purpose tools to expand its applicability beyond strictly vision-focused analysis.

Second, despite substantial improvements over strong baselines, SPECTRA occasionally exhibits hallucinations within intermediate reasoning steps, even when the final prediction is correct. This phenomenon indicates that while the outcome is robust, the interpretability and factual grounding of the reasoning chain can be further refined. Future work will explore the integration of specific hallucination-removal objectives and consistency constraints to mitigate these errors and ensure high fidelity across the entire reasoning process.

9 Ethical Considerations

This study relies exclusively on publicly available datasets for the training and evaluation of SPECTRA. These resources were utilized solely for scientific research and academic purposes. We explicitly state that we do not claim ownership or licensing rights over any of the datasets employed in this

work. All data remains subject to the original licenses, terms of use, and copyright protections established by the respective source providers. We have made every effort to ensure our usage complies with these original distribution terms. In accordance with AI usage guidelines, generative AI tools were used solely for the purposes of paraphrasing, grammar correction, and spell checking.

Acknowledgments

Tanmoy Chakraborty acknowledges the support of Microsoft Research Grant, Azure AI Credits Grant from Microsoft’s Accelerating Foundation Models Research (AFMR) initiative, Google GCP Grant, and Rajiv Khemani Young Faculty Chair Professorship in Artificial Intelligence.

References

- Anonymous. 2025. [Beyond the final answer: Evaluating the reasoning trajectories of tool-augmented agents](#). In *Submitted to The Fourteenth International Conference on Learning Representations*. Under review.
- Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao 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, and 8 others. 2025. [Qwen2.5-vl technical report](#). *Preprint*, arXiv:2502.13923.
- Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, and 95 others. 2022. [On the opportunities and risks of foundation models](#). *Preprint*, arXiv:2108.07258.
- Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. [End-to-end object detection with transformers](#). *Preprint*, arXiv:2005.12872.
- Wenlong Deng, Yushu Li, Boying Gong, Yi Ren, Christos Thrampoulidis, and Xiaoxiao Li. 2025. [On grp collapse in search-r1: The lazy likelihood-displacement death spiral](#). *Preprint*, arXiv:2512.04220.
- Yu Du, Fangyun Wei, and Hongyang Zhang. 2024. [Any-tool: Self-reflective, hierarchical agents for large-scale api calls](#). *Preprint*, arXiv:2402.04253.
- Yue Fan, Xiaoqian Ma, Rujie Wu, Yuntao Du, Jiaqi Li, Zhi Gao, and Qing Li. 2024. [Videoagent: A](#)

- memory-augmented multimodal agent for video understanding. *ArXiv*, abs/2403.11481.
- Zhi Gao, Bofei Zhang, Pengxiang Li, Xiaojuan Ma, Tao Yuan, Yue Fan, Yuwei Wu, Yunde Jia, Song-Chun Zhu, and Qing Li. 2025. **Multi-modal agent tuning: Building a VLM-driven agent for efficient tool usage**. In *The Thirteenth International Conference on Learning Representations*.
- Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. 2024. **Hallusionbench: An advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models**. *Preprint*, arXiv:2310.14566.
- Tanmay Gupta and Aniruddha Kembhavi. 2022. **Visual programming: Compositional visual reasoning without training**. *Preprint*, arXiv:2211.11559.
- Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. **Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor**. *Preprint*, arXiv:1801.01290.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. **Lora: Low-rank adaptation of large language models**. *Preprint*, arXiv:2106.09685.
- Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, and 399 others. 2024. **Gpt-4o system card**. *Preprint*, arXiv:2410.21276.
- Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. 2016. **A diagram is worth a dozen images**. *Preprint*, arXiv:1603.07396.
- Daesik Kim, Seonhoon Kim, and Nojun Kwak. 2019. **Textbook question answering with multi-modal context graph understanding and self-supervised open-set comprehension**. *Preprint*, arXiv:1811.00232.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. **Efficient memory management for large language model serving with pagedattention**. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.
- Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023a. **Blip-2: bootstrapping language-image pre-training with frozen image encoders and large language models**. In *Proceedings of the 40th International Conference on Machine Learning, ICML'23*. JMLR.org.
- Minghao Li, Yingxiu Zhao, Bowen Yu, Feifan Song, Hangyu Li, Haiyang Yu, Zhoujun Li, Fei Huang, and Yongbin Li. 2023b. **Api-bank: A comprehensive benchmark for tool-augmented llms**. *Preprint*, arXiv:2304.08244.
- Zhecheng Li, Guoxian Song, Yujun Cai, Zhen Xiong, Junsong Yuan, and Yiwei Wang. 2025. **Texture or semantics? vision-language models get lost in font recognition**. *ArXiv*, abs/2503.23768.
- Shilong Liu, Hao Cheng, Haotian Liu, Hao Zhang, Feng Li, Tianhe Ren, Xueyan Zou, Jianwei Yang, Hang Su, Jun Zhu, Lei Zhang, Jianfeng Gao, and Chunyuan Li. 2023. **Llava-plus: Learning to use tools for creating multimodal agents**. *Preprint*, arXiv:2311.05437.
- Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, and 3 others. 2025. **Agentbench: Evaluating llms as agents**. *Preprint*, arXiv:2308.03688.
- Zuxin Liu, Thai Quoc Hoang, Jianguo Zhang, Ming Zhu, Tian Lan, Shirley Kokane, Juntao Tan, Weiran Yao, Zhiwei Liu, Yihao Feng, Rithesh R N, Liangwei Yang, Silvio Savarese, Juan Carlos Niebles, Huan Wang, Shelby Heinecke, and Caiming Xiong. 2024. **APIGen: Automated Pipeline for generating verifiable and diverse function-calling datasets**. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
- Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kaiwei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. **Learn to explain: Multimodal reasoning via thought chains for science question answering**. *Preprint*, arXiv:2209.09513.
- Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. 2019. **Ok-vqa: A visual question answering benchmark requiring external knowledge**. *Preprint*, arXiv:1906.00067.
- Grégoire Mialon, Clémentine Fourier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. 2023. **Gaia: a benchmark for general ai assistants**. *Preprint*, arXiv:2311.12983.
- Microsoft, :, Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, Dong Chen, Dongdong Chen, Junkun Chen, Weizhu Chen, Yen-Chun Chen, Yi ling Chen, Qi Dai, and 57 others. 2025. **Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras**. *Preprint*, arXiv:2503.01743.
- Arindam Mitra, Luciano Del Corro, Guoqing Zheng, Shweti Mahajan, Dany Rouhana, Andres Codas, Yadong Lu, Wei ge Chen, Olga Vrousos, Corby Rosset, Fillipe Silva, Hamed Khanpour, Yash Lara, and Ahmed Awadallah. 2024. **Agentinstruct: Toward**

- generative teaching with agentic flows. *Preprint*, arXiv:2407.03502.
- Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. [Training language models to follow instructions with human feedback](#). *Preprint*, arXiv:2203.02155.
- Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, Sihan Zhao, Lauren Hong, Runchu Tian, Ruobing Xie, Jie Zhou, Mark Gerstein, dahai li, Zhiyuan Liu, and Maosong Sun. 2024. [ToolLLM: Facilitating large language models to master 16000+ real-world APIs](#). In *The Twelfth International Conference on Learning Representations*.
- Rajkumar Ramamurthy, Prithviraj Ammanabrolu, Kianté Brantley, Jack Hessel, Rafet Sifa, Christian Bauckhage, Hannaneh Hajishirzi, and Yejin Choi. 2023. [Is reinforcement learning \(not\) for natural language processing: Benchmarks, baselines, and building blocks for natural language policy optimization](#). *Preprint*, arXiv:2210.01241.
- Aymeric Roucher, Albert Villanova del Moral, Thomas Wolf, Leandro von Werra, and Erik Kaunismäki. 2025. ‘smolagents’: a smol library to build great agentic systems. <https://github.com/huggingface/smolagents>.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. [Proximal policy optimization algorithms](#). *Preprint*, arXiv:1707.06347.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. [Deepseekmath: Pushing the limits of mathematical reasoning in open language models](#). *Preprint*, arXiv:2402.03300.
- Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. [Hugging-GPT: Solving AI tasks with chatGPT and its friends in hugging face](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. [Hybridflow: A flexible and efficient rlhf framework](#). *arXiv preprint arXiv:2409.19256*.
- R. Smith. 2007. An overview of the tesseract ocr engine. In *Proceedings of the Ninth International Conference on Document Analysis and Recognition - Volume 02, ICDAR '07*, page 629–633, USA. IEEE Computer Society.
- Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan J. Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. 2020. [Learning to summarize from human feedback](#). *ArXiv*, abs/2009.01325.
- Dídac Surís, Sachit Menon, and Carl Vondrick. 2023. [Vipergpt: Visual inference via python execution for reasoning](#). *Preprint*, arXiv:2303.08128.
- Qiaoyu Tang, Ziliang Deng, Hongyu Lin, Xianpei Han, Qiao Liang, Boxi Cao, and Le Sun. 2023. [Toolalpaca: Generalized tool learning for language models with 3000 simulated cases](#). *Preprint*, arXiv:2306.05301.
- Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. [Gemma 3 technical report](#). *Preprint*, arXiv:2503.19786.
- Shulin Tian, Ziniu Zhang, Liangyu Chen, and Ziwei Liu. 2025. [Mmina: Benchmarking multihop multimodal internet agents](#). *Preprint*, arXiv:2404.09992.
- Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. 2024. [Eyes wide shut? exploring the visual shortcomings of multimodal llms](#). *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9568–9578.
- Harsh Trivedi, Tushar Khot, Mareike Hartmann, Ruskin Manku, Vinty Dong, Edward Li, Shashank Gupta, Ashish Sabharwal, and Niranjan Balasubramanian. 2024. [Appworld: A controllable world of apps and people for benchmarking interactive coding agents](#). *Preprint*, arXiv:2407.18901.
- Chenyu Wang, Weixin Luo, Sixun Dong, Xiaohua Xuan, Zhengxin Li, Lin Ma, and Shenghua Gao. 2025. [Mllm-tool: A multimodal large language model for tool agent learning](#). *Preprint*, arXiv:2401.10727.
- Jize Wang, Ma Zerun, Yining Li, Songyang Zhang, Cailian Chen, Kai Chen, and Xinyi Le. 2024a. [GTA: A benchmark for general tool agents](#). In *The Thirty-eighth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
- Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024b. [Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution](#). *Preprint*, arXiv:2409.12191.
- Zhenyu Wang, Aoxue Li, Zhenguo Li, and Xihui Liu. 2024c. [Genartist: Multimodal LLM as an agent for unified image generation and editing](#). In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

- Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. 2023. [Visual chatgpt: Talking, drawing and editing with visual foundation models](#). *Preprint*, arXiv:2303.04671.
- Zhiheng Xi, Yiwen Ding, Wenxiang Chen, Boyang Hong, Honglin Guo, Junzhe Wang, Dingwen Yang, Chenyang Liao, Xin Guo, Wei He, Songyang Gao, Lu Chen, Rui Zheng, Yicheng Zou, Tao Gui, Qi Zhang, Xipeng Qiu, Xuanjing Huang, Zuxuan Wu, and Yu-Gang Jiang. 2024. [Agentgym: Evolving large language model-based agents across diverse environments](#). *Preprint*, arXiv:2406.04151.
- Yunzhong Xiao, Yangmin Li, Hewei Wang, Yunlong Tang, and Zora Zhiruo Wang. 2025. [Toolmem: Enhancing multimodal agents with learnable tool capability memory](#). *Preprint*, arXiv:2510.06664.
- Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. 2024. [OSWorld: Benchmarking multimodal agents for open-ended tasks in real computer environments](#). In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
- Zhenghai Xue, Longtao Zheng, Qian Liu, Yingru Li, Xiaosen Zheng, Zejun Ma, and Bo An. 2025. [Simpletir: End-to-end reinforcement learning for multi-turn tool-integrated reasoning](#). *Preprint*, arXiv:2509.02479.
- Sherry Yang, Ofir Nachum, Yilun Du, Jason Wei, Pieter Abbeel, and Dale Schuurmans. 2023. [Foundation models for decision making: Problems, methods, and opportunities](#). *Preprint*, arXiv:2303.04129.
- Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. [React: Synergizing reasoning and acting in language models](#). *Preprint*, arXiv:2210.03629.
- Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, Qianyu Chen, Huarong Zhou, Zhensheng Zou, Haoye Zhang, Shengding Hu, Zhi Zheng, Jie Zhou, Jie Cai, Xu Han, and 4 others. 2024. [Minicpm-v: A gpt-4v level mllm on your phone](#). *Preprint*, arXiv:2408.01800.
- Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, Yu Su, Wenhui Chen, and Graham Neubig. 2025. [Mmmu-pro: A more robust multi-discipline multimodal understanding benchmark](#). *Preprint*, arXiv:2409.02813.
- Yuxiang Zhai, Hao Bai, Zipeng Lin, Jiayi Pan, Shengbang Tong, Yifei Zhou, Alane Suhr, Saining Xie, Yann LeCun, Yi Ma, and Sergey Levine. 2024. [Fine-tuning large vision-language models as decision-making agents via reinforcement learning](#). *Preprint*, arXiv:2405.10292.
- Bofei Zhang, Zirui Shang, Zhi Gao, Wang Zhang, Rui Xie, Xiaojian Ma, Tao Yuan, Xinxiao Wu, Song-Chun Zhu, and Qing Li. 2025a. [Tongui: Internet-scale trajectories from multimodal web tutorials for generalized gui agents](#). *Preprint*, arXiv:2504.12679.
- Jianguo Zhang, Tian Lan, Rithesh Murthy, Zhiwei Liu, Weiran Yao, Ming Zhu, Juntao Tan, Thai Hoang, Zuxin Liu, Liangwei Yang, Yihao Feng, Shirley Kokane, Tulika Awalgaonkar, Juan Carlos Niebles, Silvio Savarese, Shelby Heinecke, Huan Wang, and Caiming Xiong. 2024. [Agentohana: Design unified data and training pipeline for effective agent learning](#). *Preprint*, arXiv:2402.15506.
- Yabo Zhang, Yihan Zeng, Qingyun Li, Zhen Hu, Kavin Han, and Wangmeng Zuo. 2025b. [Tool-rl: Sample-efficient reinforcement learning for agentic tool use](#). *Preprint*, arXiv:2509.12867.
- Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. 2024a. [Gpt-4v\(ision\) is a generalist web agent, if grounded](#). *ArXiv*, abs/2401.01614.
- Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. 2024b. [Gpt-4v\(ision\) is a generalist web agent, if grounded](#). *Preprint*, arXiv:2401.01614.
- Sipeng Zheng, jiazheng liu, Yicheng Feng, and Zongqing Lu. 2024c. [Steve-eye: Equipping LLM-based embodied agents with visual perception in open worlds](#). In *The Twelfth International Conference on Learning Representations*.
- Yifei Zhou, Andrea Zanette, Jiayi Pan, Sergey Levine, and Aviral Kumar. 2024. [Archer: Training language model agents via hierarchical multi-turn rl](#). *Preprint*, arXiv:2402.19446.
- Jiaru Zou, Ling Yang, Yunzhe Qi, Sirui Chen, Mengting Ai, Ke Shen, Jingrui He, and Mengdi Wang. 2025. [Autotool: Dynamic tool selection and integration for agentic reasoning](#). *Preprint*, arXiv:2512.13278.
- K.J Åström. 1965. [Optimal control of markov processes with incomplete state information](#). *Journal of Mathematical Analysis and Applications*, 10(1):174–205.

A Benchmark Details

This section details the public datasets utilized for the training and in- and out-of-distribution evaluation of SPECTRA. A sample from each dataset is illustrated in Figure 7. Here is the details of datasets involved in SPECTRA’s training and in-distribution evaluations:

- **Allen Institute for Artificial Intelligence Diagrams (AI2D):** AI2D (Kembhavi et al., 2016) dataset contains grade school level textbook style scientific diagrams paired with multiple choice questions that evaluate diagram understanding and visual reasoning. Contains

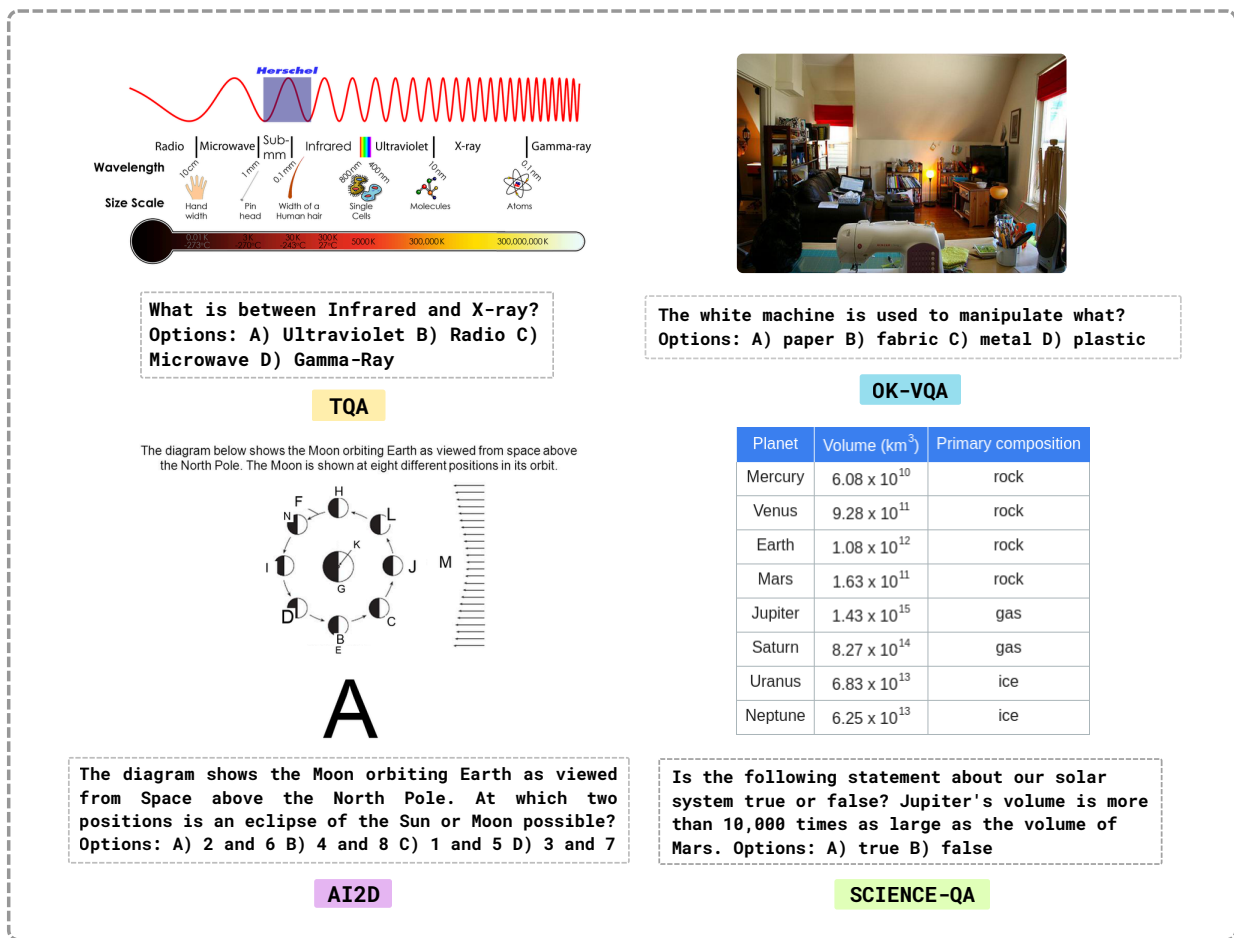


Figure 7: A sample from each of the four datasets (AI2D, TQA, OK-VQA, and Science QA) used in SPECTRA's training and in-distribution evaluation.

questions from physics, biology and Earth and environmental science.

- **Textbook Question Answering (TQA):** TQA (Kim et al., 2019) Textbook Question Answering (TQA) dataset is a multimodal benchmark constructed from middle school science textbooks that combines text passages, diagrams, and multiple-choice questions. Questions are from physics, biology and geography.
- **Outside Knowledge Visual Question Answering (OK-VQA):** OK-VQA (Marino et al., 2019) is a visual question answering dataset where answering questions about images requires external world knowledge beyond what is visible in the image. The questions cover broad domains such as common-sense, culture, geography, history, and objects, encouraging models to combine vision with knowledge-based reasoning.
- **Science Question Answering (ScienceQA):**

ScienceQA (Lu et al., 2022) is a large multimodal question answering dataset built from grade-school science curricula, containing questions with images, diagrams, text explanations, and multiple-choice answers. The dataset spans subjects such as natural science, social science, and language arts, and emphasizes reasoning, explanation, and multimodal understanding.

Out-Of-Distribution Evaluation. To assess SPECTRA's out-of-distribution robustness, we evaluate it against Massive Multi-discipline Multimodal Understanding and Reasoning Pro version (MMMUPro) (Yue et al., 2025), a contemporary multimodal benchmark requiring complex multi-discipline reasoning. The dataset is an enhanced iteration of the original MMMU dataset spanning 30 college-level subjects. This benchmark is designed to filter out text-only biases and prioritize tasks that require deep integration of visual and textual information, reflecting real-world academic challenges.

For our evaluation, we focused on the standard setting (four-option multiple choice) and specifically isolated single-image questions to maintain consistency with our training environment. This refined subset results in a final testing dataset of 1,592 (out of 1730) data points, ensuring a focused and robust assessment of the model’s multimodal capabilities.

B Tools

To enable robust multimodal processing, SPECTRA integrates a specialized set of tools targeting distinct visual capabilities. High-level semantic context is derived using the Image Captioning Tool for scene summarization, while fine-grained entity localization is achieved via the Object Detection Tool. Textual information is extracted by the OCR Tool, and complex spatial reasoning is handled by the Visual Perception Tool, powered by a VLM, ensuring comprehensive scene interpretation and detail analysis. Respective implementation details for each tool is presented at Table 4.

C Methodology And Evaluation Measures Cont.

In this section we provide additional details that need derivation and disclosure for reproducibility.

C.1 Hierarchical Soft Structural Integrity Reward Details

The hierarchical soft structural integrity reward term is given by:

$$R_{\text{struct}}(\tau) = \alpha \cdot \gamma^{\phi(\tau)} \quad (16)$$

where $\alpha = 2.0$, $\gamma = 0.75$, and $\phi(\tau)$ map partial structures (e.g., missing perception) to higher integers.

Our motivation is to maximize rewards for the optimal multi-turn rollout structure without penalizing other valid paths that reach the correct solution. This approach ensures the model captures learning signals from various successful topological constraints, encouraging it to discern when tool usage is unnecessary.

We define three rollout structure templates as follows:

Template: Optimal (Z_1)

```

<think_reasoning> ... </think_reasoning>
  <tool_call> ... </tool_call>
  <tool_response> ... </tool_response>
<think_perception> ... </think_perception>
<think_reasoning> ... </think_reasoning>
.....
<answer> ... </answer>
```

Template: Valid (Z_2)

```

<think_reasoning> ... </think_reasoning>
<think_perception> ... </think_perception>
<think_reasoning> ... </think_reasoning>
<answer> ... </answer>
```

Template: Alternative (Z_3)

```

<think_reasoning> ... </think_reasoning>
<answer> ... </answer>
```

We define the reward signals for the three template structures based on the parameterization shown in Table 5. The reward contribution is zero in case model follows any deviations other than the defined ones.

C.2 Tool Selectivity Score (TSS) Details

While maximum entropy promotes exploration and randomness (Xue et al., 2025), we minimize entropy to demonstrate tool selectivity. To calculate the TSS, we begin by assuming that the tool selection policy is initially represented as a uniform distribution across a set of K tools. Let $T = \{T_1, T_2, T_3, \dots, T_K\}$ denote the set of $|K|$ independent available tools. The initial distribution of tool selection can be modeled as follows:

$$U(T) = \text{Uniform}(T) \quad (17)$$

This implies each tool is chosen with equal probability. The probability of each tool being chosen is simply as:

$$U(T_k) = \frac{1}{K} \quad (18)$$

Subsequently, we define $P(T_k)$, which represents an empirical tool usage distribution for an agent. This distribution encapsulates the observed likelihood of selecting a given tool T_k based on the agent’s reasoning. It is defined as:

$$P(T_k) = \frac{n_k}{\sum_{j=1}^K n_j} \quad (19)$$

Here, n represents the total number of times a specific tool is utilized within a given dataset.

Tool Name	Method	Description
Image_Captioning_Tool	BLIP2 (Li et al., 2023a)	Generates a natural language summary describing the visual content of an image. <i>Response:</i> plain-text
Object_Detection_Tool	DETR (Carion et al., 2020)	Identifies and localizes specific objects within an image using bounding boxes. <i>Response:</i> box: [x1, y1, x2, y2], label: string, score: float ...
OCR_Tool	Tesseract (Smith, 2007)	Extracts and digitizes printed or handwritten text characters from visual inputs. <i>Response:</i> plain-text
Visual_Perception_Tool	Qwen2.5-VL[7B] instruct (Bai et al., 2025)	Analyzes images to generate spatial relationships or details. <i>Response:</i> plain-text

Table 4: Description of Tools.

Template	α	γ	$\phi(\tau)$	R_{struct}
Z_1 (Optimal)	2.0	0.00	0	2.000
Z_2 (Valid)	2.0	0.75	1	1.500
Z_3 (Alternative)	2.0	0.75	2	1.125

Table 5: Reward Signal Parameterization for Rollout Templates

We choose Kullback-Leibler (KL) divergence to model the distribution shift as it effectively represents the agent’s specificity in strategically selecting tools, moving away from a random uniform distribution. It is defined as follows:

$$TSS = D_{KL}(P(T) || U(T)) = \sum_{T_k \in T} P(T_k) \log \left(\frac{P(T_k)}{U(T_k)} \right) \quad (20)$$

The range for the TSS is $[0, \infty]$. To constrain the TSS parameter within the interval $[0, 1]$, we apply the hyperbolic tangent (tanh) function to the D_{KL} term. The intuition behind the parameter is that the further the distribution deviates from a random uniform distribution, the better. In other words, higher values of $\tanh(D_{KL})$, which correspond to greater divergence from randomness, are considered more desirable for an agent to strategically select provided tools. Finally, we also report a mean TSS value over all the datasets under evaluation, defined as:

$$T\bar{S}S = \frac{1}{|D_{eval}|} \sum_{d \in D_{eval}} \sum_{T_k^d \in T^d} P(T_k^d) \log \left(\frac{P(T_k^d)}{U(T_k^d)} \right) \quad (21)$$

In our experiments, we compared the TSS of SPECTRA with a baseline agent to demonstrate that SPECTRA is better equipped to strategically select tools compared to the baseline agent.

D Algorithm

In Algorithm 1, we present the SPECTRA formulation.

Algorithm 1 Self-supervised Perception Enabled by Cascaded Tool Rollout Alignment (SPECTRA)

Require: Dataset \mathcal{D} , SVLM Policy π_θ with frozen E_ϕ , Toolset T

Require: Hyperparameters: Group size G , Clip ϵ , Coefficients $\lambda_{1..4}$

Ensure: Optimized Policy Parameters θ^*

- 1: Initialize LoRA parameters $\Delta\theta$ in π_θ
- 2: **while** not converged **do**
- 3: Sample batch of inputs $\mathcal{B} = \{(I, q)\} \sim \mathcal{D}$
- 4: **for all** $(I, q) \in \mathcal{B}$ **do**
- 5: $\pi_{\theta_{old}} \leftarrow \pi_\theta$ {Freeze current policy for sampling}
- 6: **Step 1: Structured Multi-turn Rollouts**
- 7: Sample G trajectories $\{\tau_1, \dots, \tau_G\} \sim \pi_{\theta_{old}}(\cdot | I, q)$
- 8: {Ensure topology: reason \rightarrow tool \rightarrow obs \rightarrow percep \rightarrow reason \rightarrow ans}
- 9: **Step 2: Multi-Objective Reward Calculation**
- 10: $R_{group} \leftarrow \emptyset$
- 11: $N = \text{abs}(\max(R_{total}))$
- 12: **for** $i = 1$ to G **do**
- 13: Calculate component rewards:
- 14: $(r_c, r_s, r_t, r_m) \in \tau_i$
- 15: Aggregate, Normalize and Scale:
- 16: $R_{total}(\tau_i) \leftarrow \lambda_1 r_c + \lambda_2 r_s + \lambda_3 r_t + \lambda_4 r_m$
- 17: $R_{final}(\tau_i) \leftarrow S * R_{total}(\tau_i) / N$
- 18: Append $R_{total}(\tau_i)$ to R_{group}
- 19: **end for**
- 20: **Step 3: Trajectory Advantage Estimation**
- 21: Compute group stats: $\mu, \sigma \leftarrow \text{Mean}(R_{group}), \text{Std}(R_{group})$
- 22: **for** $i = 1$ to G **do**
- 23: $\hat{A}_i \leftarrow \frac{R_{total}(\tau_i) - \mu}{\sigma + \delta}$
- 24: **end for**
- 25: **end for**
- 26: **Step 4: GRPO Optimization**
- 27: Compute Loss $\mathcal{J}_{CS-ARL}(\theta)$
- 28: Update $\theta \leftarrow \theta + \eta \nabla_\theta \mathcal{L}$
- 29: **end while**
- 30: **return** θ

E Experimental Results Cont.

In this section, we present extended results.

E.1 Tool Utility And Efficiency Cont.

In this section, we present a qualitative analysis of each tool’s performance. Specifically, we evaluate

Tool	TER (%)	
	Baseline Agent	SPECTRA
Image_Captioning_Tool	87.2	87.8
Object_Detection_Tool	65.2	75.0
OCR_Tool	50.0	-
Visual_Perception_Tool	73.3	89.4

Table 6: Tool-level TER statistics for SPECTRA 7B against baseline agent.

the Tool Execution Reliability (TER) of a tool for SPECTRA in comparison to the baseline agent. The tool-level TER results are presented in Table 6. The results are averaged over in-distribution test set.

The qualitative analysis demonstrates that SPECTRA generally outperforms the baseline agent, most notably in Visual Perception (+16.1%) and Object Detection (+9.8%). While both agents show nearly identical performance on the Image Captioning Tool, a distinct difference in tool selection strategy is observed regarding OCR. Specifically, SPECTRA does not utilize the OCR Tool, whereas the baseline agent invoked it only twice with a 50% failure rate, suggesting that SPECTRA’s improved performance may stem from a more selective and reliable tool-calling logic.

Additionally, we conducted a human evaluation of tool efficiency, analyzing 50 random samples from the test set. Out of these, tool calls occurred in 40 samples (80%), with 29 (72.9%) identified as relevant (considering Tool Efficiency and Tool Selectivity Score metrics) based on human judgment. Of the relevant tool calls, 28 (96.5%) resulted in correct outcomes, demonstrating a strong correlation between relevant tool calls and favorable results in human verification too.

E.2 Agentic Trajectory Analysis Cont.

The detailed quantitative results for agentic trajectory analysis are presented in Table 7.

Additionally, we conducted a human-verified qualitative analysis of the <think_perception> tag on 50 samples, finding favorable outcomes for 40/50 cases (80%). Among these, 77.5% followed the <think_perception> based rollout structure, while 90.3% of the responses demonstrated high-quality summarization and accurate understanding of image history based on the human judgment. These results confirm the higher likelihood of achieving favorable outcomes when using optimized, task-specific structured rollouts.

F Prompts And Chat Templates

F.1 Prompts

Figure 8 illustrates the system prompt for the zero-shot setting, while Figure 9 displays the rollout prompt utilized during both training and inference for the agentic setup.

F.2 Chat Templates

Qwen-2.5-VL-3B As the Qwen-2.5-VL-3B model has no prior training for tool calling which was evident from our experimentation. We incorporated additional changes in the chat template in addition to the base prompt, a function signature for all the tools. All the hyperparameter and prompts used for training the model are exactly same as the Qwen-2.5-VL-7B-instruct model. The chat template for Qwen-2.5-VL-3B model is provided in Figure 10.

Qwen-2.5-VL-7B Custom chat Template for Qwen-2.5-VL-7B model is provided in Figure 11.

G Qualitative Analysis: Case Studies

We conduct a thorough analysis and present four case studies in this section. Qualitative analysis demonstrates SPECTRA’s superior capability in tool orchestration and error recovery compared to the baseline across three successful cases, while also examining a failure scenario to ensure a comprehensive evaluation. As illustrated in Figure 12, SPECTRA mitigates the baseline’s reliance on brittle heuristics and OCR failures by strategically selecting descriptive perception tools to resolve fine-grained visual distinctions, such as insect life stages. Furthermore, Figure 13 highlights the model’s robustness in handling syntactical errors; while the baseline fails on minimal outputs, SPECTRA successfully validates semantic roles to enforce correct terminal conditions. Crucially, Figure 14 reveals an emergent self-correction mechanism where SPECTRA, unlike the baseline which persists in illogical predictions, leverages detailed tool feedback to refine its initial hypotheses and correctly deduce complex relationships in food web dynamics.

Finally, in Figure 15, we have shown how SPECTRA can succumb to confirmation bias when tool outputs are insufficiently granular. Although the model attempts to verify its reasoning, the generic response from the captioning tool fails to

Source State	Target State	End Outcome	Baseline	SPECTRA	Δ
<i>Top Improvements (Increased Correct Outcomes)</i>					
Reasoning	Terminal	Correct	616	664	+48
Reasoning	Tool_Call	Correct	444	488	+44
Tool_Call	Tool_Call	Correct	20	63	+43
Terminal	Correct	Correct	581	618	+37
Perception	Reasoning	Correct	414	442	+28
Tool_Response_Success	Reasoning	Correct	81	109	+28
Tool_Call	Tool_Response_Success	Correct	414	436	+22
<i>Top Reductions (Decreased Incorrect Outcomes)</i>					
Tool_Call	Tool_Call	Incorrect	140	37	-103
Reasoning	Tool_Call	Incorrect	167	123	-44
Tool_Call	Incorrect	Incorrect	50	9	-41
Tool_Response_Success	Tool_Call	Incorrect	33	14	-19
Tool_Call	Tool_Response_Success	Incorrect	146	127	-19
Perception	Incorrect	Incorrect	12	5	-7
Tool_Call	Tool_Response_Failure	Incorrect	9	3	-6
<i>Other Notable Transitions</i>					
Perception	Reasoning	Incorrect	81	97	+16
Terminal	Tool_Response_Success	Correct	36	51	+15
Tool_Call	Tool_Response_Failure	Correct	13	28	+15
Tool_Call	Perception	Correct	37	48	+11
Reasoning	Terminal	Incorrect	134	142	+8

Table 7: Differential State Transition Analysis: Baseline Agent vs. SPECTRA. The table highlights impactful shifts, sorted by the magnitude of change (Δ). SPECTRA demonstrates a significant reduction in error loops (negative Δ in Incorrect outcomes) and an increase in successful reasoning paths (positive Δ in Correct outcomes). The cutoff value for absolute delta magnitude is 5.

You are a helpful multi-modal assistant. Solve this question step by step and choose the correct option and put the final answer inside `<answer> \boxed{(option)answer} </answer>` tags.

Figure 8: The system prompt used to guide the Zero-shot inference.

contradict the initial spatial misconception, leading the model to reinforce rather than correct its error.

H Computational Statistics

All experiments were conducted on two NVIDIA A100 80GB GPU². Training was completed in approximately 10 hours for 100 steps, utilizing a batch size of 64 across 8 trajectories. Inference on the 800-sample test set required 6 minutes and 10 seconds; detailed parameter settings are provided in the Appendix I. The computational statistics presented here use pre-extracted tool outputs to optimally use the compute for training and inference.

²<https://www.nvidia.com/en-in/data-center/a100>

I Hyperparameter

Table 8 details the specific hyperparameters used for the fine-tuning setup and reward formulation.

J Nomenclature

This section provides comprehensive definitions for the hyperparameters and constants referenced throughout the study. Detailed specifications and their respective definitions are presented in Table 9.

You are a helpful multi-modal reasoning assistant specializing in math, science, and general knowledge questions. You are given a question from a user and you have access to a set of Available_tools. Answer in this sequence:

1. Start by thinking inside `<think_reasoning>... </think_reasoning>` tags about the question and which tool to call if needed.
2. Then you can call tool or tools from Available_tools. Tool call instructions:
For each function call, return a json object with function name and arguments within `<tool_call></tool_call>` XML tags:
`<tool_call>{"name": <function-name>, "arguments": <args-json-object>} </tool_call>` end of response.
List of Available_tools: captioning_tool, ocr_tool, detection_tool and perception_tool. (DO NOT make up any other tools or tool names. Only use the set of tools given to you. Also avoid repetition)
3. After you have used the tools, you will see the tool outputs inside appropriate tags in the same order from the system.
4. After getting tool_response, think Your thoughts on the tool_response inside `<think_perception>... </think_perception>` tags once. (think_perception step only comes after tool_response tags)
5. Then resume your thought process inside `<think_reasoning>... </think_reasoning>` tags again. Try to think clearly, aloud and step-by-step so that you reach to the correct final answer. If needed also reflect on your thought process. All thinking must be inside `<think_reasoning>... </think_reasoning>` tags.
6. At the end you must always choose the most appropriate option and put answer inside: `<answer>\boxed{(option)answer} </answer>` tags, irrespective of the output of the tool being true or false or incorrect.

You MUST provide an answer even if uncertain at the end.

Figure 9: The structured rollout prompt for training and inference used in SPECTRA.

```
You are provided with function signatures within <tools></tools> XML tags:
<tools>
{"type": "function", "function": {"name": "perception_tool",
"description": "A tool for understanding images/visual questions better.",
"parameters": {"type": "object", "properties": {"image_url": {"type": "string",
"description": "The URL or dataset ID of the image."}}, "required": ["image_url"]}}}

{"type": "function", "function": {"name": "captioning_tool",
"description": "A tool for generating captions.", "parameters": {"type": "object",
"properties": {"image_url": {"type": "string",
"description": "The URL or dataset ID of the image to caption."}}, "required": ["image_url"]}}}

{"type": "function", "function": {"name": "detection_tool",
"description": "A tool for detecting objects in an image.", "parameters": {"type": "object",
"properties": {"image_url": {"type": "string", "description": "The URL or dataset ID
of the image to analyze."}, "threshold": {"type": "number", "description":
"The confidence threshold for object detection is 0.9."}}, "required": ["image_url"]}}}

{"type": "function", "function": {"name": "ocr_tool",
"description": "A tool for extracting text from image.", "parameters": {"type": "object",
"properties": {"image_url": {"type": "string", "description": "The URL or dataset ID
of the image to extract text from."}}, "required": ["image_url"]}}}
</tools>

Do not forget to put your answer in <answer> \boxed{(option)answer} </answer> tags at the end.
Now following these instructions solve this question:
```

Figure 10: Tool signature included in the chat template for Qwen-2.5-VL-3B (SPECTRA).

Use Specific tool_call JSON structure:
 <tool_call>
 {"name": "<pick necessary tool_name from Available_tools>", "arguments":{image_url}}
 </tool_call>(End of response).
 Do not forget to put your answer in <answer> \boxed{(option)answer} </answer> tags at the end.
 Now following these instructions solve this question:

Figure 11: Custom chat template text prompt for Qwen-2.5-VL-7B (SPECTRA).

Parameters Type	Sub-Category	Parameter	Value
Training	Fine-tuning	#response per prompt	8
		Batch size	64
		Minibatch size	32
		Microbatch size	8
		LoRA rank	64
		LoRA alpha	64
		n_itter	100
		max_turns	16
		max_prompt_length	16384
		max_response_tokens	2048
		Learning rate	$1e^{-6}$
	top_p	0.95	
	top_k	-1	
	temperature	1.0	
	Engine	VLLM	
Reward	λ_1	1.0	
	λ_2	1.0	
	λ_3	2.0	
	λ_4	3.0	
	C_1	8.0	
	C_2	2.0	
	α	2.0	
	β	0.1	
	γ	0.75	
	η	0.8	
	κ	2	
S	2.5		
N_{norm}	21.6		
GRPO	ϵ_l	0.2	
	ϵ_h	0.4	
	ψ (KL Coeff)	0.001	
Others	Compute	NVIDIA-A100-80GB	2x
		top_p	0.7
	Inference	top_k	-1
		temperature	1.0
		max_response_token	2048

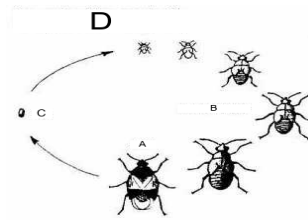
Table 8: Hyperparameters Configuration.

Qualitative Analysis: SPECTRA Improves the Selection of a Right Tool

Question: What stage does letter B represent in the life cycle of the flower bug?

Options:

- A) stages B) adult
- C) egg D) nymphal



ai2d/test_image_91

× Baseline Agent

<think_reasoning>

The image provided shows a life cycle diagram for an insect, likely a flower bug, given the context. The stages are labeled as A, B, C, and D. Typically, the life cycle of an insect includes egg, nymph, adult, and sometimes a pupa stage. In the diagram: - D appears to be the smallest, which could represent eggs. - A and C could represent nymph stages, as they are larger than D and smaller than B. - B appears to be the largest, which could represent the adult stage. Given this reasoning, letter B likely represents the adult stage of the flower bug. I will now call the ocr_tool to confirm the text on the image if necessary.

```
<tool_call> {"name": "ocr_tool", "arguments": {"image_url": "ai2d/test_image_91"}} </tool_call>
<tool_response> Image Perception tool failed: {"success": false, "message": "OCR agent is incapable of detecting text for this image. Please try another image perception/understanding tool."} </tool_response>
```

<think_reasoning>

Since the OCR tool was unable to detect text from the image, we cannot confirm the exact labels for the stages A, B, C, and D. However, based on the typical life cycle of an insect and the visual representation in the image, we can still infer that B is the largest form, which likely represents the adult stage. Therefore, the answer is:

```
<answer> B)adult </answer> (Incorrect)
```

✓ SPECTRA (Ours)

<think_reasoning>

The image shows a life cycle of a flower bug with labeled stages. Stage B appears to be the largest, mature form of the bug, which typically represents the adult stage in the life cycle of insects. The other stages (A, C, and D) seem to represent earlier stages, such as nymphs or eggs. Therefore, the answer is B) adult.

```
<tool_call> {"name": "perception_tool", "arguments": {"image_url": "ai2d/test_image_91"}} </tool_call>
<tool_response> {"success": true, "message": "This image illustrates the life cycle of an insect, likely a type of beetle or true bug, depicted in a circular diagram with four distinct stages labeled A, B, C, and D.- **Stage A**:- The largest insect shown, this is the adult stage. It has a fully developed body with wings, six legs, and prominent antennae. It appears to be the reproductive stage of the life cycle.- **Stage B**:- A smaller, wingless insect with a similar body shape to A but without fully developed wings. This represents the nymph stage, which is an intermediate phase between the egg and adult.- **Stage C**:- A small, oval-shaped object labeled with a Q and an arrow pointing from it to stage D. This represents the egg stage. The Q likely stands for 'egg' or is a placeholder label.- **Stage D**:- Two small, immature forms shown above stage C. These are likely the early nymphal stages or instars, which are the first stages after hatching from the egg. The arrow from C to D indicates that the egg hatches into these early nymphs. The diagram uses curved arrows to show the progression: from the egg (C) to the early nymphs (D), then to later nymphal stages (B), and finally to the adult (A). The cycle then loops back from the adult (A) to the egg (C), indicating that adults lay eggs to start the cycle anew. The diagram is a simplified representation of the insect's development, focusing on the key stages without showing the intermediate molting stages between nymphal instars."} </tool_response>
```

<think_perception>

The tool response confirms that stage B in the life cycle of the flower bug is the nymphal stage. The image shows that stage B is a smaller, wingless insect with a similar body shape to the adult but without fully developed wings, which is characteristic of the nymphal stage in the life cycle of many insects.

<think_reasoning>

Based on the analysis of the image and the tool response, stage B represents the nymphal stage in the life cycle of the flower bug.

```
<answer> D)nymphal </answer> (Correct)
```

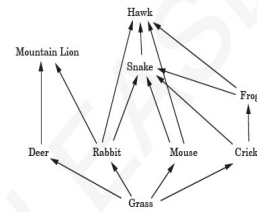
Figure 12: Qualitative comparison on a complex visual reasoning task. The Baseline relies on incorrect visual heuristics (size) and fails when OCR fails. SPECTRA leverages a descriptive perception tool to correctly distinguish the wingless nymph (Stage B) from the adult (Stage A).

Qualitative Analysis: SPECTRA Enforcing Terminal Condition and Syntactical Tool Calls Errors

Question: In the food web, what role does grass play?

Options:

- A) Producer and source of energy for consumers
- B) Decomposer
- C) Predator
- D) Consumer



ai2d/test_image_56

× Baseline Agent

<think_reasoning>

The question asks about the role of grass in the food web. To determine this, we need to understand the relationships shown in the food web diagram. 1. Grass is at the bottom of the food web, meaning it is the starting point for energy flow. 2. Arrows point from grass to other organisms, indicating that these organisms consume grass. 3. The organisms that consume grass are rabbits and mice, which are labeled as consumers in the food web. Given this information, grass plays the role of a producer in the food web, providing energy for the primary consumers. It is not a decomposer, predator, or consumer itself.

Therefore, the correct answer is A) Producer and source of energy for consumers.

<tool_call>"name": "None", "arguments": {}**</tool_call>**

addCriterion

<tool_call>"name": "None", "arguments": {}**</tool_call>**

addCriterion

Repeat N times

addCriterion

user

<tool_response>

Error when executing tool: 'None' **</tool_response>**

assistant <think

✓ SPECTRA (Ours)

<think_reasoning>

To determine the role of grass in the food web, let's analyze the image and the options provided: 1. **Option A: Producer and source of energy for consumers** - Grass is typically a producer in a food web. Producers are organisms that can produce their own food through photosynthesis and are the base of the food chain. They provide energy to other organisms when they are consumed. 2. **Option B: Decomposer** - Decomposers break down dead organic matter and waste products into simpler substances. Grass is not a decomposer. 3. **Option C: Predator** - Predators are organisms that hunt and eat other organisms. Grass is not a predator. 4. **Option D: Consumer** - Consumers are organisms that cannot produce their own food and must eat other organisms for energy. Grass is not a consumer. Given the image, we can see that grass has arrows pointing towards it from the Mountain Lion, Rabbit, and Cricket, indicating that it is being consumed. Therefore, grass plays the role of a producer and source of energy for consumers.

</think_reasoning>

<tool_call>"name": "captioning_tool", "arguments": {"image_url": "ai2d/test_image_56"}**</tool_call>**

</tool_call>

addCriterion: The image is a food web diagram showing the relationships between different organisms and their food sources. The diagram includes arrows pointing from one organism to another, indicating the direction of energy flow. The organisms in the diagram are: Mountain Lion, Deer, Hawk, Snake, Rabbit, Frog, Cricket, Mouse, and Grass. The arrows show that Grass is consumed by Deer, Rabbit, and Cricket, and is consumed by the Mountain Lion.

</captioning_tool> [Repeat 3 times same text] **</captioning_tool>**

addCriterion: The image is a food web diagram showing the relationships between different organisms and their food sources. The diagram includes arrows pointing from one organism to another, indicating the direction of energy flow. The organisms in the diagram are: Mountain Lion, Deer, Hawk, Snake, Rabbit, Frog, user

<tool_response> "success": true, "message": "a diagram of the different types of animals"**</tool_response>**

</tool_response>

assistant <

<think_perception>

The tool_response provided the image of the food web, which is crucial for understanding the role of each organism. The arrows indicate the direction of energy flow, with the arrows pointing towards Grass showing that it is consumed by Deer, Rabbit, and Cricket, and by the Mountain Lion.

<think_reasoning>

Based on the food web diagram, Grass is at the base of the food chain, as it is consumed by the other organisms (Deer, Rabbit, Cricket, and Mountain Lion). This means that Grass is a producer and a source of energy for these consumers. Therefore, the correct answer is:

<answer> A **answer </answer>** (Correct)

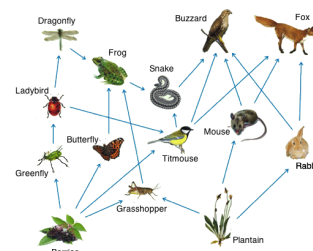
Figure 13: Qualitative comparison showing SPECTRA's ability to recover from minimal tool output. While the Baseline Agent errors out on tool usage, SPECTRA correctly interprets the "captioning_tool" response and validates the producer role of grass in the food web.

Qualitative Analysis: SPECTRA Tool Use Guided Correction

Question: What is most likely to happen if the Plantain population is wiped out?

Options:

- A) The grasshopper population will decrease.
- B) The mouse population will increase.
- C) The ladybug population will increase.
- D) The dragonfly population will decrease.



ai2d/test_image_97

× Baseline Agent

<think_reasoning>

The image provided is a food web diagram that shows the relationships between different organisms. The arrows indicate the direction of energy flow, with the arrow pointing from the prey to the predator. In this context, if the Plantain population is wiped out, we need to identify which organism will be directly affected and how it will impact the populations of other organisms.

1. ****Identify the direct relationship with Plantain**:** The Plantain is directly consumed by the mouse. Therefore, if the Plantain population is wiped out, the mouse population will be directly affected. 2. ****Evaluate the impact on other populations**:** - The grasshopper population is not directly connected to Plantain, so it will not be directly affected. - The ladybug population is not directly connected to Plantain, so it will not be directly affected. - The dragonfly population is not directly connected to Plantain, so it will not be directly affected. 3. ****Determine the correct answer**:** Since the mouse population will lose a food source (Plantain), it is likely that the mouse population will decrease if the Plantain population is wiped out.

Based on the reasoning above, the correct answer is B) The mouse population will increase.

<answer> B) The mouse population will increase. **</answer>** (Incorrect)

✓ SPECTRA (Ours)

<think_reasoning>

To solve this question, let's analyze the food web in the image. The Plantain is shown to be consumed by the Mouse. Therefore, if the Plantain population is wiped out, the Mouse will lose its primary food source.

Let's evaluate each option: A) The grasshopper population will decrease. There is no direct relationship between Plantain and Grasshopper in the food web. B) The mouse population will increase. Since the Plantain is a primary food source for the Mouse, if it is wiped out, the Mouse population will likely decrease due to a lack of food. C) The ladybug population will increase. There is no direct relationship between Plantain and Ladybug in the food web. D) The dragonfly population will decrease. There is no direct relationship between Plantain and Dragonfly in the food web.

The correct answer is B, as the Mouse population will decrease due to the loss of its primary food source.

<tool_call> {"name": "perception_tool", "arguments": {"image_url": "ai2d/test_image_97"}} **</tool_call>**

<tool_response> {"success": true, "message": "This image is a food web diagram illustrating the complex feeding relationships among various animals and plants in an ecosystem. The diagram is composed of multiple organisms connected by blue arrows, which indicate the direction of energy flow from the organism being eaten to the organism that eats it. At the base of the food web are the producers: Berries and Plantain. These plants provide energy for primary consumers (herbivores). Arrows point from Berries to Greenfly, Butterfly, and Grasshopper, and from Plantain to Grasshopper. These insects are then consumed by secondary consumers: Ladybird (eats Greenfly and Butterfly), Frog (eats Ladybird and Grasshopper), and Titmouse (eats Grasshopper and Butterfly). The food web becomes more complex as it moves upward. The Frog is preyed upon by the Snake and the Buzzard. The Ladybird is eaten by the Frog and the Buzzard. The Grasshopper is also eaten by the Frog and the Titmouse. The Butterfly is eaten by the Frog and the Titmouse. The Snake is a predator that eats the Frog, the Mouse, and the Titmouse. The Mouse is preyed upon by the Snake, the Buzzard, and the Fox. The Rabbit is eaten by the Fox and the Buzzard. The Fox is a top predator in this web, preying on the Mouse, Rabbit, and Buzzard. The Buzzard is a versatile predator, preying on the Ladybird, Frog, Snake, Mouse, and Rabbit. The Titmouse is preyed upon by the Snake. This diagram demonstrates the interconnectedness of species in an ecosystem, showing how energy flows through multiple trophic levels and how the removal or addition of one species can have cascading effects on others. The arrows clearly indicate the direction of energy transfer, with the producers at the bottom and the top predators at the top."} **</tool_response>**

<think_perception>

The tool_response provided a detailed explanation of the food web, confirming that the Plantain is a food source for the Grasshopper. The loss of Plantain would lead to a decrease in the Grasshopper population, as they would not have a primary food source.

The correct answer is A) The grasshopper population will decrease.

<think_reasoning>

The tool_response confirmed that the Plantain is a food source for the Grasshopper. If the Plantain population is wiped out, the Grasshopper population will likely decrease due to a lack of food. This directly supports option A. **<answer>** A) The grasshopper population will decrease. **</answer>**

(Correct)

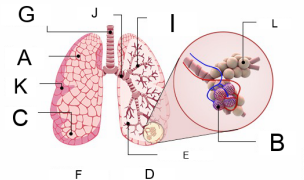
Figure 14: Qualitative comparison on a food web analysis task. The Baseline agent incorrectly interprets the food web, identifying the wrong primary consumer for Plantain and making an illogical prediction about population change. The SPECTRA initially misinterprets the diagram but corrects its reasoning after receiving a detailed description from the perception tool, correctly identifying the Plantain-Grasshopper relationship.

Qualitative Analysis: SPECTRA's Failure Case

Question: Which label represents Left Lung ?

Options:

- A) J
- B) D
- C) G
- D) F



ai2d/test_image_180

× Baseline Agent

<think_reasoning>

To determine which label represents the Left Lung, let's analyze the image:

1. The image shows a diagram of the human respiratory system, with the lungs labeled. 2. The left lung is typically positioned on the left side of the chest cavity. 3. In the diagram, the left lung is labeled with the letter 'D'. 4. The right lung is on the right side of the chest cavity and is labeled with the letter 'F'.

Given this information, the label 'D' corresponds to the Left Lung.

<answer> **</answer>** (Correct)

</think_reasoning>

✓ SPECTRA (Ours)

<think_reasoning>

The image provided is a diagram of the human respiratory system. The left lung is typically the one on the left side of the diagram when viewed from the front. In this image, the left lung is labeled with the letter "F". Therefore, the correct answer should be associated with the label "F".

Let's analyze the options:

- A) J: This is the trachea, not the left lung.
- B) D: This is the right lung, not the left lung.
- C) G: This is the right lung, not the left lung.
- D) F: This is the left lung.

Thus, the correct label for the left lung is "F".

</think_reasoning>

```
<tool_call name": "captioning_tool", "arguments": "image_url":  
"ai2d/test_image_180"</tool_call>  
<tool_response> "success": true, "message": "a diagram of the lungs and the respiratory system"  
</tool_response>
```

assistant

<think_perception>

The captioning tool provided the correct information about the diagram, confirming that the left lung is indeed labeled as "F".

</think_perception>

<think_reasoning>

The analysis and the tool response confirm that the left lung is labeled as "F". Therefore, the correct answer is:

</think_reasoning>

<answer> **</answer>** (Incorrect)

Figure 15: Qualitative comparison on an anatomical diagram analysis task. The Baseline agent correctly identifies the Left Lung by recognizing the standard anatomical orientation. The SPECTRA fails due to a initial incorrect reasoning, assuming the visual left corresponds to the anatomical left. It attempts to verify this via a captioning tool, but the generic response lacks specific label details, causing the agent failing in confirmation of its initial bias and select the incorrect label

Notation	Description
h_t	Hidden latent representation of the model at time step t .
τ	Represents the distinct rollout trajectory.
G	G is the number of trajectories (response) per question.
π_θ	It is the policy model at current step.
$\pi_{\theta_{old}}$	It is the policy model at the previous step.
$\pi_{\theta_{ref}}$	It is frozen base policy model for calculating KL penalty.
E_ϕ	It is the frozen vision encoder.
ρ	Importance sampling ratio or probability ratio.
a_t	Action taken by the model at time step t .
s_t	State at time step t .
ϵ_l	This is the lower bound for the clip ratio.
ϵ_h	This is the upper bound for the clip ratio.
ψ	It is the KL regularization coefficient.
R_{corr}	This is the task correctness reward.
R_{struct}	It represents the hierarchical structure reward.
R_{tool}	It is defined as the tool utility reward.
R_{term}	This is the terminal reward.
R_{div}	It is the diversity reward term.
R_{Total}	This is the total reward.
λ_1	This is the weight for R_{corr} term.
λ_2	This is the weight for R_{struct} term.
λ_3	It is the weight for R_{tool} term.
λ_4	This is the weight for R_{term} term.
\mathbb{I}	This represents the binary indicator function.
α	Constant maximum possible hierarchical structure reward.
r	Pearson correlation between tool use and correct answer.
β	It is per tool usage reward.
γ	It is the damping factor for structural reward term.
δ	It is the numerical stability constant.
κ	This is the per tool usage cap.
η	It is the limit for maximum total diversity reward.
S	It is the scaling factor for the total reward.
N_{norm}	It is the normalization factor (maximum possible reward).
T_{cap}	It is the image captioning tool.
T_{det}	It defines the object detection tool.
T_{ocr}	This is the optical character recognition (OCR) tool.
T_{vp}	This is the visual perception tool.
\mathcal{A}	It defines the action space.
$C1$	It is the reward magnitude for correct answer.
$C2$	It is the reward magnitude for true terminal case.
$\phi(\tau)$	This is the mapping for partial structure.
$Z1$	This represents the optimal (gold standard) rollout.
$Z2$	This represents the second prioritized valid rollout.
$Z3$	It is the rollout with least priority.
K	K is the total number of tools used.

Table 9: Nomenclature.