ADAPTIVE LOOK, FLEXIBLE THOUGHT: LEARNING HOW AND HOW LONG TO THINK IN MULTIMODAL REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

The introduction of long chain-of-thought (CoT) reasoning and external tools has substantially improved the performance of multimodal large language models (MLLMs) on general and reasoning tasks. However, existing fast-slow reasoning paradigms mainly regulate textual reasoning length, while visual computation introduced by external tools or visual tokens is often treated as an auxiliary component. This separation prevents MLLMs from jointly optimizing visual perception and textual reasoning. To address this issue, we propose **L2T** (adaptive **L**ook, flexible **T**hought), a unified reasoning framework that jointly optimizes the **form** and **depth** of computation. L2T first discretizes tool-produced visual states into visual tokens, bringing visual perception and textual reasoning into a shared autoregressive space. Based on this unified representation, L2T introduces a modality-aware routing reward to select appropriate reasoning modes and a difficulty-aware fast-slow strategy to adapt reasoning depth according to task complexity. Experiments on general perception and reasoning-intensive tasks demonstrate the effectiveness of **L2T**. Compared with explicit tool-calling approaches, **L2T** maintains competitive performance while achieving 1.7–3.6$\times$ higher throughput.

1 INTRODUCTION

Multimodal large language models (MLLMs) have demonstrated strong capabilities in visual perception and reasoning. However, their performance gains often rely on increasing test-time computation, typically through long Chain-of-Thought (CoT) reasoning or external tool use Xu et al. (2025); Huang et al. (2026); Wang et al. (2026d). While effective, these approaches introduce two key limitations: (i) they lack explicit mechanisms to adapt computation budgets according to sample difficulty, and (ii) it is expressed almost entirely in text, which becomes unreliable when answers hinge on fine-grained visual evidence Cao et al. (2025); Tian et al. (2025).

To address these issues, the 'Thinking with Images' paradigm first introduces external visual tools such as cropping, zooming, and transformation to improve visual grounding Wu & Xie (2024); OpenAI (2025); Hong et al. (2025); Zhang et al. (2025a). Although this improves reliability, it incurs significant overhead due to code generation, tool execution, and image feedback. Recent latent reasoning methods alleviate part of this latency by replacing explicit tool use with continuous latent representations Yang et al. (2025c); Ray et al. (2025); Wang et al. (2026b); Chen et al. (2026). However, they typically rely on fixed-length latent sequences and cannot dynamically adapt computation budgets to task difficulty or visual complexity.

These limitations raise a fundamental question: how should an MLLM allocate computation during multimodal reasoning? We argue that efficient multimodal reasoning is not only about *how much* to think, but also about *what form* the computation should take. Perception-heavy tasks require fine-grained processing of local regions, spatial relations, and visual attributes, where purely textual rationales may drift from image evidence and rely on language priors Tian et al. (2025). In contrast, reasoning-heavy tasks benefit from explicit symbolic abstraction and multi-step textual composition. Therefore, an efficient model should dynamically route computation between visual and textual modalities while also adjusting its reasoning depth according to task difficulty.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

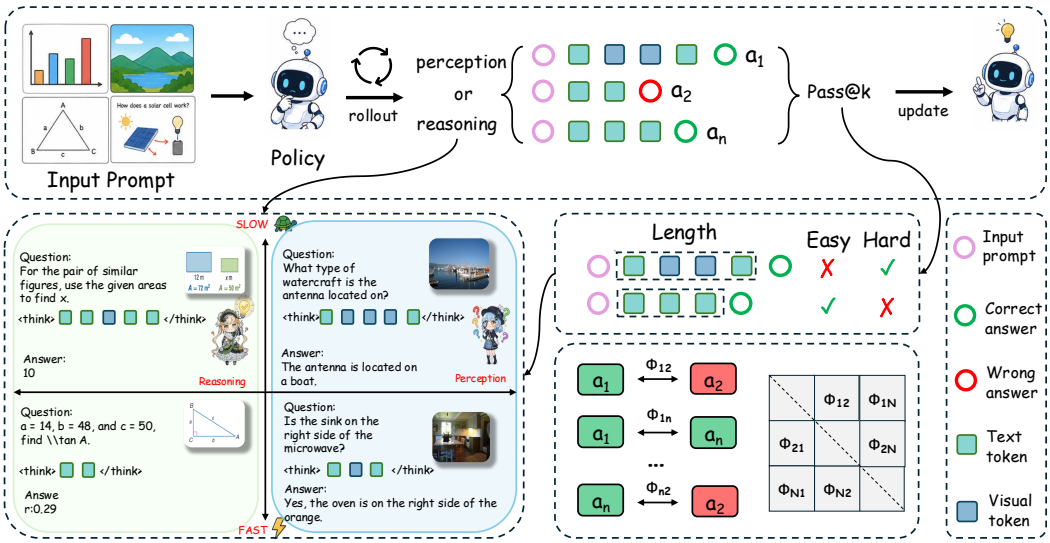


Figure 1: Overview of modality-aware routing and difficulty-aware fast-slow optimization.

Our motivation is shown in Figure 1, we propose an adaptive computation allocation framework for MLLMs. We first discretize tool-produced intermediate visual states into visual tokens, allowing visual evidence that would otherwise require external tool feedback to be represented within a single autoregressive sequence. Built on this representation, we introduce two adaptive training signals: a modality-aware routing reward that allocates computation between visual and textual tokens, and a difficulty-aware fast-slow advantage mechanism that adjusts reasoning length according to sample difficulty. Together, these components allow the model to decide both *how* to think and *how long* to think. Our contributions are summarized as follows:

1) We introduce an adaptive **L**ook, flexible **T**hought (L2T) framework that represents visual perception and textual reasoning within the same autoregressive token space, enabling unified optimization of both computation modality and computation budget for both perception and reasoning tasks.
2) Building upon this unified representation, we propose an adaptive fast-slow reasoning objective that jointly optimizes computation modality and reasoning depth through modality-aware routing and difficulty-aware budget allocation, enabling the model to determine how and how long to think.
3) Experiments demonstrate that L2T improves inference efficiency and token utilization while preserving strong multimodal reasoning performance across perception, reasoning, and OOD benchmarks.

2 Related Work

2.1 Thinking about Images

MLLMs have rapidly advanced visual understanding and multimodal reasoning by aligning visual representations with large language models Qwen Team (2025). Following the success of CoT prompting in LLMs, many MLLM methods improve performance by encouraging longer textual rationales, self-reflection, or additional test-time computation Lightman et al. (2024); Xu et al. (2025); Xiang et al. (2026a). However, thinking about images purely through text can be unreliable. Recent studies show that textual CoT may be insufficient or even harmful for visual reasoning, especially when tasks require fine-grained perception, spatial grounding, or faithful use of image evidence Xiao & Gan (2025); Tian et al. (2025). As textual reasoning becomes longer, models may increasingly rely on language priors rather than visual clues, leading to degraded performance and visual hallucination. These findings suggest that stronger multimodal reasoning should not be reduced to simply generating more textual tokens.

2.2 THINKING WITH IMAGES

To overcome the limitations of text-centric reasoning, recent 'Thinking with Images' methods incorporate visual operations or intermediate visual states into the reasoning process OpenAI (2025). Tool-based approaches augment MLLMs with visual operations such as cropping, zooming, grounding, and other image manipulations Wu & Xie (2024); Cao et al. (2025). Recent systems, including PyVision, Thyme, and DeepEye v2, further extend visual reasoning through executable tool use Zhao et al. (2025); Zhang et al. (2025a); Hong et al. (2025). Despite their effectiveness, these methods rely on code generation, tool execution, and image feedback, resulting in substantial token overhead and latency.

An alternative direction internalizes visual computation through latent reasoning, replacing explicit reasoning chains with continuous latent embeddings. Recent MLLMs extend this paradigm to visual reasoning by aligning latent states with intermediate visual representations Yang et al. (2025c); Ray et al. (2025); Chen et al. (2026). However, these methods typically employ continuous representations with a fixed number of latent tokens, limiting their adaptability and preventing token-level optimization with reinforcement learning Ray et al. (2025); Wang et al. (2026b). In contrast, our method discretizes intermediate visual computation into tokenizer-level visual tokens, enabling unified autoregressive generation and token-level reinforcement learning.

2.3 FAST-SLOW THINKING

Although increasing reasoning depth generally improves performance, it also incurs substantially higher inference costs due to the quadratic complexity of LLMs and MLLMs with respect to sequence length. Recent approaches have alleviated insufficient reasoning by extending CoT or incorporating external tools Huang et al. (2026); Xiang et al. (2026a); Hong et al. (2025). However, uniformly increasing the reasoning budget often leads to overthinking on simple problems. Besides wasting computational resources, excessive reasoning may even degrade accuracy Liu et al. (2026); Li et al. (2026). This issue is particularly pronounced in MLLMs, where overly long reasoning traces can gradually weaken attention to the input image, leading to visual hallucinations Tian et al. (2025). These observations have motivated recent research on fast-slow reasoning, which aims to allocate computation more adaptively. Kimi Team et al. (2025) introduce length-aware objectives to encourage concise reasoning, while TWYN Yang et al. (2025a), DAST Shen et al. (2025), CosineReward Yang et al. (2025b), and CARE Wang et al. (2026c) explicitly favor shorter yet correct reasoning traces. FAST further incorporates both image complexity and question difficulty to adapt reasoning length. DeepEyes v2 identifies samples that benefit from tool use through rule-based filtering and reinforcement learning, enabling adaptive tool invocation Hong et al. (2025). ATLAS introduces a regularization objective to discourage the overuse of latent tokens Guo et al. (2026). Despite these advances, existing methods do not provide a unified framework that jointly allocates textual and visual reasoning according to task requirements.

3 METHOD

3.1 DISCRETE VISUAL TOKENIZATION

Inspired by previous works Peng et al. (2022); Chen et al. (2025a), we introduce the Vector-Quantized Visual Tokenizer (VQ-VT) to bridge visual computation and autoregressive language modeling through a unified token space. Specifically, VQ-VT converts tool-produced intermediate visual states into discrete latent visual tokens that can be generated and consumed alongside textual tokens within the same autoregressive process. This unified representation enables visual and textual computation to be allocated under a common token budget, providing the foundation for our modality-aware routing and adaptive computation allocation. Following previous studies Peng et al. (2022); Chen et al. (2025b), we perform reconstruction in the feature space rather than the pixel space, as feature-level reconstruction has been shown to preserve semantic information more effectively.

Visual Codebook. Following VQ-VAE van den Oord et al. (2017), we build a visual tokenizer on top of a frozen vision encoder. Given an image, the frozen encoder extracts patch features $\{h_i\}_{i=1}^{N}$,

which are mapped by a trainable linear encoder to latent features $\{z_i\}_{i=1}^{N}$. We ℓ_2-normalize both the latent features and the codebook embeddings $\mathcal{V} = \{e_j\}_{j=1}^{K}$, and perform codebook lookup as

$$k_i = \arg\max_{1 \le j \le K} z_i^\top e_j, \qquad q_i = e_{k_i}. \tag{1}$$

A trainable linear decoder D_ϕ reconstructs the original vision feature from the quantized feature q_i, which is obtained by mapping z_i to its nearest neighbor in a codebook of size K. Using the standard straight-through estimator for the hard lookup, we optimize

$$\begin{aligned}\mathcal{L}_{\mathrm{VQ}} = \frac{1}{N}\sum_{i=1}^{N}\Big[&\|h_i - D_\phi(q_i)\|_2^2 \\ &+ \|\mathrm{sg}(z_i) - q_i\|_2^2 + \alpha\|z_i - \mathrm{sg}(q_i)\|_2^2\Big],\end{aligned} \tag{2}$$

where sg denotes the stop-gradient operator, and α is a hyperparameter balancing the training of the codebook and the encoder. After training, the K codebook indices and two boundary markers are added to the MLLM vocabulary as discrete visual tokens, which enable visual and textual reasoning to share a unified autoregressive token space for adaptive reasoning depth.

3.2 SFT Dataset Construction

We transform existing tool-calling data into discrete visual reasoning trajectories.

Given a sample (X, I, C, A), where X is the query, I is the input image, C denotes tool-calling code, and A is the answer, we execute C on the input image I to obtain intermediate visual states, the intermediate image is encoded by the pre-trained VQ-VT encoder and quantizer:

$$I' = \mathrm{Execute}(C, I). \tag{3}$$

$$V = \mathcal{Q}(\mathcal{E}(I')) = \{v_1, \ldots, v_m\},\ v_j \in \mathcal{V} \tag{4}$$

Finally, we replace the original tool code with the corresponding visual token sequences:

$$D = \{X, text, v_{start}, v_1, v_2, \ldots, v_n, v_{end}, text, A\}. \tag{5}$$

3.3 Three-Stage Training Strategy.

Stage 1: Learning to Think Visually. We first perform supervised fine-tuning on the constructed dataset to initialize visual reasoning. Since visual reasoning trajectories are represented as discrete visual tokens, the model learns to generate these tokens at appropriate reasoning steps via standard next-token prediction:

$$\mathcal{L}_{\mathrm{SFT}} = -\sum_i \log P(s_i | s_{<i}, X, I). \tag{6}$$

After this stage, the model acquires the ability to invoke visual tokens and associate them with corresponding visual transformations.

Stage 2: Learning to Think Reliably. Stage 2 performs reinforcement learning to establish a stable reasoning policy before introducing adaptive routing. Each response is required to follow

$$y = \texttt{<think>}\ z\ \texttt{</think>}\ \texttt{<answer>}\ \hat{a}\ \texttt{</answer>}, \tag{7}$$

where the `<think>` field contains the reasoning process and the `<answer>` field is used for answer verification.

The Stage 2 reward is defined as

$$r_{\text{stable}} = r_{\text{acc}} \cdot r_{\text{fmt}}, \tag{8}$$

where

$$r_{\text{acc}} = \mathbb{I}[\text{Verify}(\hat{a}, a^*)],\ r_{\text{fmt}} = \mathbb{I}[y \in \mathcal{F}]. \tag{9}$$

Here, a^* is the correct answer, $\mathcal{F}$ denotes the set of valid responses following the required <think>–<answer> format.

This warm-up stage encourages the model to generate correct and well-structured responses, providing a stable initialization for the adaptive routing and fast-slow reasoning objectives introduced in Stage 3.

Stage 3: Learning to Think Adaptively. As shown in Figure 1, after learning to produce stable and verifiable responses, the model is further optimized to reason adaptively. Specifically, Stage 3 jointly learns (1) modality-aware routing between visual and textual reasoning and (2) difficulty-aware reasoning budgets. This stage turns stable reasoning into adaptive reasoning. It has two coupled goals: selecting the appropriate reasoning modality for each task and allocating a suitable reasoning token for each sample, which help model learns to allocate computational resources more selectively according to task demands.

Modality-aware Routing. To learn adaptive reasoning modality, we construct a modality-aware Routing reward. Given the reasoning trace z within the <think> field, let $V(z)$ and $T(z)$ denote the numbers of visual and textual reasoning tokens. Their normalized ratios are

$$\rho_v(z) = \frac{V(z)}{V(z) + T(z)}, \qquad \rho_t(z) = \frac{T(z)}{V(z) + T(z)}. \tag{10}$$

The visual and textual activation scores are defined as

$$s_v = \min\left(\frac{\rho_v}{\tau_v}, 1\right) \min\left(\frac{V}{V_{\min}}, 1\right), \tag{11}$$

$$s_t = \min\left(\frac{\rho_t}{\tau_t}, 1\right) \min\left(\frac{T}{T_{\min}}, 1\right), \tag{12}$$

$$s_v^{\text{mix}} = \min\left(\frac{\rho_v}{\tau_v^{\text{mix}}}, 1\right) \min\left(\frac{V}{V_{\min}}, 1\right), \tag{13}$$

$$s_t^{\text{mix}} = \min\left(\frac{\rho_t}{\tau_t^{\text{mix}}}, 1\right) \min\left(\frac{T}{T_{\min}}, 1\right). \tag{14}$$

For each training sample, we assign two binary routing labels. $n_v, n_t \in \{0, 1\}$, where n_v and n_t indicate whether visual-centered and text-centered reasoning are required, respectively. Routing labels are initialized based on dataset/task categories and further refined through keyword-based rules.

The route score is computed according to four routing types:

$$\tilde{s}_{\text{route}} = \begin{cases} s_v, & n_v = 1, n_t = 0, \\ s_t \lambda_{\text{novis}}^{\mathbb{I}[V>0]}, & n_v = 0, n_t = 1, \\ \min(s_v^{\text{mix}}, s_t^{\text{mix}}), & n_v = 1, n_t = 1, \\ \lambda_{\text{simple}}^{\mathbb{I}[V>0]}, & n_v = 0, n_t = 0. \end{cases} \tag{15}$$

The visual-only route rewards visual-token usage through s_v, which increases when the visual ratio ρ_v and the number of visual tokens V are sufficiently large. The text-only route rewards textual reasoning through s_t, but additionally applies the penalty λ_{novis} whenever any visual token is generated. Moreover, when T is fixed, increasing V lowers $\rho_t = T/(T+V)$, so the text-only score can

decrease further as more visual tokens are used. In contrast, the simple route only applies a binary penalty λ_{simple} once $V > 0$; additional visual tokens do not cause extra penalty beyond this indicator. The mixed route requires both visual and textual channels to be active by taking the minimum of the two mixed activation scores.

The route score is incorporated as a soft shaping reward,

$$R_i = (0.9 \cdot r_{\text{acc},i} + 0.1 \cdot s_{\text{route},i}) \cdot r_{\text{stable}}, \tag{16}$$

Thus, correctness remains the primary optimization target, and the route score only acts as a weak shaping signal among valid responses. This prevents routing from overriding answer correctness while still encouraging the model to express computation in the modality required by the task.

Difficulty-aware Fast-Slow Thinking. To learn adaptive reasoning budgets, we construct a pass-rate-based contrastive objective.

For a rollout group G, the group pass rate is

$$q_i = \mathbb{I}[R_i > \tau_{\text{corr}}], \qquad p_G = \frac{1}{|G|} \sum_{i \in G} q_i. \tag{17}$$

Here, τ_{corr} represents the threshold score for correctness. For simplicity, we set $\tau_{\text{corr}} \approx 1$, which requires both the route reward and the final answer to be correct.

Given rollout reward R_i and reasoning length L_i, we define

$$\Delta R_{ij} = R_i - R_j, \qquad \Delta L_{ij} = \tanh\left(\frac{L_i - L_j}{\max(\sigma_L, L_{\min})} \right). \tag{18}$$

Here, σ_L denotes the standard deviation of reasoning lengths within the rollout group. The thresholds τ_{easy} and τ_{hard} divide prompts into easy, medium, and hard groups according to the group pass rate.

The pairwise potential is

$$c_{ij} = \mathbb{I}[R_i > \tau_{\text{corr}}]\mathbb{I}[R_j > \tau_{\text{corr}}], \tag{19}$$
$$h_{ij} = \mathbb{I}[R_i > \tau_{\text{corr}}]\mathbb{I}[R_j \leq \tau_{\text{corr}}]. \tag{20}$$

$$\Phi_{ij} = \begin{cases} \Delta R_{ij} - \beta \Delta L_{ij} c_{ij}, & p_G \geq \tau_{\text{easy}}, \\ \Delta R_{ij}, & \tau_{\text{hard}} \leq p_G < \tau_{\text{easy}}, \\ \Delta R_{ij} + \beta' [\Delta L_{ij}]_+ h_{ij}, & p_G < \tau_{\text{hard}}. \end{cases} \tag{21}$$

This objective favors shorter successful traces on easy samples, preserves longer successful traces on hard samples, and uses only reward ranking for medium-difficulty samples. In this way, the model is not uniformly compressed; instead, its reasoning length is adapted according to empirical sample difficulty.

The sequence-level advantage is computed by aggregating pairwise potentials,

$$A_i = \text{Norm}\left(\frac{1}{|G| - 1} \sum_{j \neq i} \Phi_{ij} \right), \tag{22}$$

where $\text{Norm}(\cdot)$ denotes group-wise normalization.

Policy Optimization. Finally, we optimize the policy using GRPO DeepSeek-AI (2025) with our dynamic advantage A_i:

$$\begin{aligned} \mathcal{L}_{\text{GRPO}} &= -\frac{1}{|G|}\sum_{i\in G}\frac{1}{|y_i|}\sum_{t} g_{i,t} + \lambda_{\text{KL}} D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}}), \\ g_{i,t} &= \min\{\rho_{i,t}A_i,\ \text{clip}(\rho_{i,t}, 1-\epsilon, 1+\epsilon)A_i\}, \\ \rho_{i,t} &= \frac{\pi_\theta(y_{i,t} \mid x, y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t} \mid x, y_{i,<t})}. \end{aligned} \tag{23}$$

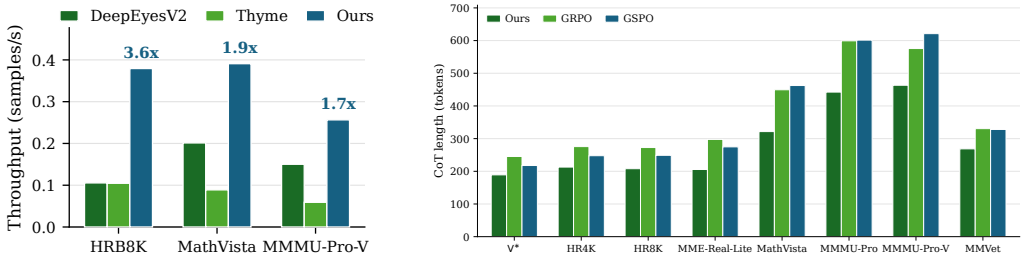
Here, y_i is a rollout in group G, π_{ref} is the reference policy, and A_i is computed from the pass-rate contrastive objective. By encoding routing and length preferences in A_i, the objective favors correct responses with appropriate routing and deliberation length. The pseudo code is shown in Appendix A.6.

4 Experiment and Discussion

4.1 Experimental Setup

We conduct experiments based on Qwen2.5-VL Qwen Team (2025). Experiments cover visual-centric perception and reasoning-centric understanding. We compare our method with a diverse set of baselines, including general open-source models, textual reasoning methods, tool-calling and latent reasoning methods, as well as vanilla slow thinking RLVR baselines trained from the same SFT model. The slow-thinking RLVR baseline is trained under the same settings as our method and is primarily used to evaluate whether our fast-slow thinking framework effectively reduces the reasoning budget. Detailed training configurations, dataset processing, and evaluation protocols are provided in the Appendix A.7. Further experiments on on the out-of-distribution (OOD) benchmarks are reported in Appendix A.1. Additional experiments on visual tokens are presented in Appendix A.2. More experiments on fast–slow thinking and CoT are provided in Appendix A.3 and A.4, respectively.

4.2 Main Results



(a) The comparison of throughput between tool-calling method and our method

(b) COT length comparison of RLVR baseline and our method

Figure 2: Throughput and generation length comparison.

Compare Efficiency with tool-calling Method. We compare the inference throughput of our method with tool-calling approaches on a single H100 GPU. Experiments are conducted on three benchmarks with different characteristics: HRBench8K (high-resolution perception), MathVista$_{\text{mini}}$ (reasoning), and MMMU-Pro-V (mixed perception and reasoning). As shown in Figure 2a, our method consistently achieves substantially higher throughput. In particular, it is 3.6× faster than the strongest tool-based baseline, DeepEyesV2, on HRBench8K, and provides 1.7–1.9× speedups on the other two benchmarks. These results demonstrate that our method significantly reduces the computational overhead introduced by explicit tool execution.

Compare Token Length with RLVR Baseline. We further compare the generated respond length with RLVR baselines. As shown in Figure 2b compared with both vanilla GRPO and GSPO, our

Table 1: Performance comparison across perception and reasoningcentric multimodal benchmarks. The best result is shown in **bold**, and the second-best result is underlined. For benchmark results reported in prior papers, we directly cite the published numbers.For missing entries, we evaluate the corresponding models under the same environment and mark these results with †.

Models	Visual-Centric Perception				Reasoning-Centric Understanding				
	V*	HR4K	HR8K	MME-Lite	MMMU-Pro	MMMU-Pro-V	MathVista	MMVet	AVG
Open-Source Models									
Qwen2.5-VL-3B Qwen Team (2025)	74.9†	66.1†	63.5†	42.4	31.6	28.5†	62.3	61.8	53.9
Qwen2.5-VL-7B Qwen Team (2025)	76.4†	68.4†	64.6†	44.9	38.3	35.6†	68.2	67.1	57.9
Textual Reasoning Models									
Vision-R1-7B Huang et al. (2026)	80.6†	75.6†	70.5†	51.5†	38.5†	35.3†	72.1	**76.4**†	62.6
FAST-3B Xiao & Gan (2025)	69.1	61.1	54.7	40.9	35.0	29.8	66.2	64.0	52.6
FAST-7B Xiao & Gan (2025)	76.4†	73.1†	66.5†	48.3†	40.2†	37.1†	73.8	71.2	60.8
PAPO-3B Wang et al. (2026d)	75.9†	65.3†	63.7†	43.6†	27.1	29.1†	61.4	69.7†	54.5
PAPO-7B Wang et al. (2026d)	78.5†	71.5†	67.5†	47.0†	36.6	38.5†	69.5	73.9†	60.4
CARE-3B Wang et al. (2026c)	–	–	–	–	33.1	24.0	66.5	–	–
CARE-7B Wang et al. (2026c)	–	–	–	–	39.7	37.1	**74.7**	–	–
tool-calling and Latent Reasoning Models									
Thyme-RL Zhang et al. (2025a)	82.2	77.0	72.0	55.2	24.6†	33.6†	70.0	71.3†	60.7
Monet Wang et al. (2026b)	83.3	71.0	68.0	55.5	36.0†	31.2†	63.6†	60.4†	58.6
EVA Chen et al. (2026)	80.2	74.0	69.5	49.8	–	–	–	–	
DeepEyesV2 Hong et al. (2025)	81.8	**77.9**	**73.8**	50.4	39.1†	34.6†	70.1	64.7†	61.6
Slow Thinking RLVR Baseline									
SFT-7B+GRPO DeepSeek-AI (2025)	81.7	74.8	71.8	53.8	39.2	36.5	69.9	75.6	62.9
SFT-7B+GSPO Zheng et al. (2025a)	79.1	76.0	69.6	**54.2**	40.3	37.1	67.8	75.9	62.5
Our Models									
L2T-3B (Ours)	73.8	68.6	67.0	49.6	33.9	30.8	64.5	75.2	57.9
L2T-7B (Ours)	**83.8**	75.2	72.7	53.0	**40.7**	37.1	72.9	75.4	**63.9**

method consistently produces fewer tokens while achieving better performance, demonstrating that it improves reasoning efficiency by reducing token consumption without sacrificing accuracy.

Results on Standard Benchmarks. Table 1 summarizes the performance of our method on standard multimodal benchmarks spanning both perception- and reasoning-centric tasks, verifies that the efficiency gains do not come from a collapse in task ac- curacy. Our method achieves the best overall average performance, outperforming textual reasoning, tool-calling reasoning, latent reasoning, and RLVR baselines trained from the same SFT initialization. Notably, existing reasoning-oriented MLLMs generally exhibit a trade-off between perception and reasoning: textual reasoning models excel at reasoning but often degrade visual perception, particularly on high-resolution benchmarks, whereas tool-calling and latent reasoning models achieve stronger perception but limited reasoning gains. In contrast, our method effectively mitigates this trade-off, achieving the highest average score (63.9) while remaining competitive across both perception- and reasoning-centric benchmarks.

4.3 Ablation Studies

Ablation on Route Reward To evaluate the proposed modality-aware routing reward, we remove this objective and first compare token allocation. As shown in Table 2, the full model substantially shortens the average CoT on both visual-centric and reasoning-centric tasks. At the same time, it increases the visual-token ratio much more on visual-centric tasks than on reasoning-centric tasks, indicating that routing does not simply maximize or minimize visual-token usage but instead allocates computational resources more selectively according to task demands while preserving task accuracy.

Ablation on Difficulty-aware Fast-Slow. To verify the necessity of difficulty-aware fast-slow reasoning, we remove the pass-rate-based length preference and first examine generated length. Table 2 shows that the full model uses shorter traces than the ablated variant on both visual-centric and reasoning-centric tasks. This indicates that pass-rate difficulty signals help reduce unnecessary reasoning, especially on samples where shorter successful traces are sufficient. After this length

Table 2: Ablation on Route Reward and Difficulty-aware fast-slow.

Category	standard (%)	w/o route reward (%)	w/o fast-slow
Ration of Visual Token			
Visual-Centric	31.8	21.9 (-9.9)	–
Reasoning-Centric	11.56	11.03 (-0.53)	–
Average Length of CoT			
Visual-Centric	206.8	292.6 (+85.8)	234.7 (+27.9)
Reasoning-Centric	416.2	547.0 (+130.8)	469.9 (+53.7)
Accuracy			
Standard Benchmark	63.9	63.8 (-0.1)	63.7 (-0.2)

reduction, the result also shows that average accuracy is preserved with only small changes. Which indicates that we did not pursue shorter CoT at the expense of accuracy.

Table 3: Ablation on visual token and two-stage rl.

Components	Visual-Centric	Reasoning-Centric	Total Δ
w/o visual token	70.6	55.4	-1.7
w/o two-stage rl	69.3	56.1	-2.3
standard	71.2	56.5	–

Ablation on Visual Token To verify the effectiveness of the proposed Visual Tokens, we replace all Visual Tokens with identical, semantically meaningless ¡unk¿ tokens in stage 1. The training for Stage 2 and Stage 3 is the same as the standard method. As shown in Table 3, our method still outperforms the baseline by nearly one percentage point. This result indicates that the improvement is not simply a consequence of extending the CoT, but instead stems from the meaningful semantic information encoded by the learned Visual Tokens.

Ablation on Two-Stage RL To evaluate the necessity of the proposed two-stage RL framework, we directly initialize Stage 3 RL from the SFT model without performing Stage 2 training, while keeping the number of RL steps identical to that of the standard Stage 2 + Stage 3 setting. As shown in Table 3, bypassing Stage 2 results in a performance degradation of 2.3 percentage points. This observation indicates that the model should first acquire reliable reasoning capabilities before learning to optimize token allocation efficiently.

5 Conclusion

In this paper, we presented a unified computation modality and budgets allocation framework for efficient multimodal reasoning. By utilizing VQ-VT to discretize intermediate visual tool outputs into visual tokens, our framework unifies visual and textual computation within the same autoregressive token space, eliminating the need for external tool execution during inference. Built upon this representation, a modality-aware routing objective determines which form of computation to use, while a difficulty-aware fast-slow objective determines how much computation to allocate according to task difficulty. Experiments demonstrate that our framework consistently improves computational efficiency, without sacrificing multimodal reasoning performance. These results suggest that efficient multimodal reasoning is fundamentally a computation allocation problem. Rather than uniformly increasing reasoning effort, an MLLM should learn both which reasoning modality to employ for different task types and how much computation to allocate for tasks of different difficulty. Future work should develop richer visual representations to enhance perceptual representation.

AI USE STATEMENT

(This section is **required** and does not count toward the page limit.)

In this work, we used generative AI tools to assist with writing and polishing, including translating and improving the clarity and readability of the manuscript. We also used these tools for literature retrieval and discovery, as well as research execution, including assisting with code generation and organization, conducting LLM-based judging, and organizing experimental failure modes.

We did not use generative AI tools to generate synthetic datasets or to prove mathematical claims. Other AI-assisted research tasks not mentioned above were not applicable to this work.

All AI-assisted outputs were reviewed by the authors. AI-assisted code was inspected and tested for correctness. The authors also conducted spot checks of LLM-based judging results and failure mode categorizations against the evaluation criteria and experimental evidence. All AI-assisted writing and translations were reviewed and edited by the authors to ensure accuracy and preserve the intended scientific meaning.

We take full responsibility for the final content of this work, including all text, claims, experimental results, code, and other artifacts produced with the assistance of generative AI.

REPRODUCIBILITY STATEMENT

(This section is **recommended** and does not count toward the page limit.)

We have made efforts to facilitate the reproducibility of our work. Detailed experimental settings and hyperparameters are provided in the Appendix A.7, and example source code is included in the supplementary materials to illustrate the implementation and experimental pipeline. Upon acceptance, we will publicly release the complete source code, pretrained model weights, and datasets used in our experiments, together with the necessary instructions for reproducing our results. These materials are intended to provide the information and resources necessary to reproduce the reported experiments and further investigate our approach.

REFERENCES

Meng Cao, Haoze Zhao, Can Zhang, Xiaojun Chang, Ian Reid, and Xiaodan Liang. Ground-r1: Incentivizing grounded visual reasoning via reinforcement learning, 2025. URL https://arxiv.org/abs/2505.20272.

Quanlin Chen, Dehua Lu, Weisen Feng, Chunjin Ye, Lina Qiu, Jiahui Pan, and Jingcong Li. Harformer: A masked self-supervised transformer-based model for human activity recognition with predicting somatosensory tokens. *IEEE Transactions on Instrumentation and Measurement*, 74: 1–11, 2025a. doi: 10.1109/TIM.2025.3584138.

Quanlin Chen, Chunjin Ye, Yiming Ma, Jiahui Pan, and Jingcong Li. A novel perspective on leveraging hubness in vae for eliminating representative shift vectors in few-shot learning. In *2025 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, 2025b. doi: 10.1109/ICME59968.2025.11209098.

Xiuwei Chen, Wentao Hu, Yongxin Wang, Zisheng Chen, Likui Zhang, Kun Xiang, Jianhua Han, Hui-Ling Zhen, Jingyuan Zou, Hang Xu, and Xiaodan Liang. Latent visual states for efficient multimodal reasoning, 2026. URL https://arxiv.org/abs/2606.24233.

DeepSeek-AI. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, et al. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 11198–11201, 2024.

Sachin Goyal, Ziwei Ji, Ankit Singh Rawat, Aditya Krishna Menon, Sanjiv Kumar, and Vaishnavh Nagarajan. Think before you speak: Training language models with pause tokens. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=ph04CRkPdC.

Ziyu Guo, Rain Liu, Xinyan Chen, and Pheng-Ann Heng. Atlas: Agentic or latent visual reasoning? one word is enough for both, 2026. URL https://arxiv.org/abs/2605.15198.

Jack Hong, Chenxiao Zhao, ChengLin Zhu, Weiheng Lu, Guohai Xu, and Xing Yu. Deepeyesv2: Toward agentic multimodal model, 2025. URL https://arxiv.org/abs/2511.05271.

Wenxuan Huang, Bohan Jia, Shaosheng Cao, Zheyu Ye, Fei zhao, Zhe Xu, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=UZIjskfbfU.

Ang Li, Charles Wang, Deqing Fu, Kaiyu Yue, Zikui Cai, Wang Bill Zhu, Ollie Liu, Peng Guo, Willie Neiswanger, Furong Huang, Tom Goldstein, and Micah Goldblum. Zebra-cot: A dataset for interleaved vision language reasoning, 2025. URL https://arxiv.org/abs/2507.16746.

Ziming Li, Jiatan Huang, Xiaoguang Guo, Guilin Wang, and Chuxu Zhang. Same signal, opposite meaning: Direction-informed adaptive learning for llm agents, 2026. URL https://arxiv.org/abs/2605.06908.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

Fan Liu, Yanhao Wang, Min Zhang, Zhikang Chen, Zeyuan Li, Lewei He, and Jiahui Pan. Think faster than words: Efficient LLM chain-of-thought reasoning via dynamic shortcut decoding. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 28825–28836, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.1330. URL https://aclanthology.org/2026.acl-long.1330/.

Pan Lu, Ran Gong, Shibiao Jiang, Liang Qiu, Siyuan Huang, Xiaodan Liang, and Song-Chun Zhu. Inter-GPS: Interpretable geometry problem solving with formal language and symbolic reasoning. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 6774–6786, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.528. URL https://aclanthology.org/2021.acl-long.528/.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 23439–23554, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/663bce02a0050c4a11f1eb8a7f1429d3-Paper-Conference.pdf.

OpenAI. Thinking with images. https://openai.com/index/thinking-with-images/, 2025. Accessed: 2025-01-26.

OpenAI. Gpt-5.5 system card. https://openai.com/index/gpt-5-5-system-card/, 2026.

Zhiliang Peng, Li Dong, Hangbo Bao, Qixiang Ye, and Furu Wei. Beit v2: Masked image modeling with vector-quantized visual tokenizers, 2022. URL https://arxiv.org/abs/2208.06366.

Alibaba Group Qwen Team. Qwen2.5-vl technical report, 2025. URL https://arxiv.org/abs/2502.13923.

Arijit Ray, Ahmed Abdelkader, Chengzhi Mao, Bryan A. Plummer, Kate Saenko, Ranjay Krishna, Leonidas Guibas, and Wen-Sheng Chu. Mull-tokens: Modality-agnostic latent thinking, 2025. URL https://arxiv.org/abs/2512.10941.

Yi Shen, Jian Zhang, Jieyun Huang, Shuming Shi, Wenjing Zhang, Jiangze Yan, Ning Wang, Kai Wang, Zhaoxiang Liu, and Shiguo Lian. DAST: Difficulty-adaptive slow-thinking for large reasoning models. In Saloni Potdar, Lina Rojas-Barahona, and Sebastien Montella (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 2322–2331, Suzhou (China), November 2025. Association for Computational Linguistics. ISBN 979-8-89176-333-3. doi: 10.18653/v1/2025.emnlp-industry.160. URL https://aclanthology.org/2025.emnlp-industry.160/.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1.5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Xinyu Tian, Shu Zou, Zhaoyuan Yang, Mengqi He, Fabian Waschkowski, Lukas Wesemann, Peter Tu, and Jing Zhang. More thought, less accuracy? on the dual nature of reasoning in vision-language models, 2025. URL https://arxiv.org/abs/2509.25848.

Aaron van den Oord, Oriol Vinyals, and koray kavukcuoglu. Neural discrete representation learning. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/7a98af17e63a0ac09ce2e96d03992fbc-Paper.pdf.

Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhu Chen. VL-rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026a. URL https://openreview.net/forum?id=4oYxzssbVg.

Qixun Wang, Yang Shi, Yifei Wang, Yuanxing Zhang, Pengfei Wan, Kun Gai, Xianghua Ying, and Yisen Wang. Monet: Reasoning in latent visual space beyond images and language. In *2026 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026b.

Wenbin Wang, Liang Ding, Minyan Zeng, Xiabin Zhou, Li Shen, Yong Luo, Wei Yu, and Dacheng Tao. Divide, conquer and combine: A training-free framework for high-resolution image perception in multimodal large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 7907–7915, 2025.

Yongxin Wang, Zhicheng Yang, Meng Cao, Mingfei Han, Haokun Lin, Yingying Zhu, Xiaojun Chang, and Xiaodan Liang. Care what fails: Contrastive anchored-reflection for verifiable multimodal reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11976–11986, June 2026c.

Zhenhailong Wang, Xuehang Guo, Sofia Stoica, Haiyang Xu, Hongru WANG, Hyeonjeong Ha, Xiusi Chen, Yangyi Chen, Ming Yan, Fei Huang, and Heng Ji. Perception-aware policy optimization for multimodal reasoning. In *The Fourteenth International Conference on Learning Representations*, 2026d. URL https://openreview.net/forum?id=izbBqTL8vb.

Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13084–13094, June 2024.

Kun Xiang, Heng Li, Terry Jingchen Zhang, Yinya Huang, Zirong Liu, Peixin Qu, Jixi He, Jiaqi Chen, Yu-Jie Yuan, Jianhua Han, Hang Xu, Hanhui Li, Mrinmaya Sachan, and Xiaodan Liang. Seephys: Does seeing help thinking? –benchmarking vision-based physics reasoning. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.),

Advances in Neural Information Processing Systems, volume 38. Curran Associates, Inc., 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/86a472735333e2a9195b9c04eb324dea-Paper-Datasets_and_Benchmarks_Track.pdf.

Kun Xiang, Zhili Liu, Terry Jingchen Zhang, Yinya Huang, Yunshuang Nie, Kaixin Cai, Yiyang Yin, Runhui Huang, Hanhui Li, Yihan Zeng, Yu-Jie Yuan, Jianhua Han, Lanqing Hong, Hang Xu, and Xiaodan Liang. Atomthink: Multimodal slow thinking with atomic step reasoning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(5):5725–5741, 2026a. doi: 10.1109/TPAMI.2026.3653573.

Kun Xiang, Terry Jingchen Zhang, Zirong Liu, Bokai Zhou, Yueling Tang, Junjie Yu, Jiacong Lu, Shangrui Huang, Heng Li, Likui Zhang, Kunkun Liu, Changzheng Zhang, Yangle Fang, Boqiang Guo, Hui-Ling Zhen, Dandan Tu, Yinya Huang, and Xiaodan Liang. Seephys pro: Diagnosing modality transfer and blind-training effects in multimodal rlvr for physics reasoning, 2026b. URL https://arxiv.org/abs/2605.09266.

Wenyi Xiao and Leilei Gan. Fast-slow thinking GRPO for large vision-language model reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=MI1uT5rReV.

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

Junjie Yang, Ke Lin, and Xing Yu. Think when you need: Self-adaptive chain-of-thought learning, 2025a. URL https://arxiv.org/abs/2504.03234.

Shiming Yang, Yuxuan Tong, Xinyao Niu, Graham Neubig, and Xiang Yue. Demystifying long chain-of-thought reasoning. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 71177–71209. PMLR, 13–19 Jul 2025b. URL https://proceedings.mlr.press/v267/yang25ae.html.

Zeyuan Yang, Xueyang Yu, Delin Chen, Maohao Shen, and Chuang Gan. Machine mental imagery: Empower multimodal reasoning with latent visual tokens, 2025c. URL https://arxiv.org/abs/2506.17218.

Huanjin Yao, Jiaxing Huang, Wenhao Wu, Jingyi Zhang, Yibo Wang, Shunyu Liu, Yingjie Wang, YuXin Song, Haocheng Feng, Li Shen, and Dacheng Tao. Mulberry: Empowering MLLM with o1-like reasoning and reflection via collective monte carlo tree search. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL https://openreview.net/forum?id=lwOV2ACEK9.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. MM-vet: Evaluating large multimodal models for integrated capabilities. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=KOTutrSR2y.

Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, Yu Su, Wenhu Chen, and Graham Neubig. MMMU-pro: A more robust multi-discipline multimodal understanding benchmark. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15134–15186, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.736. URL https://aclanthology.org/2025.acl-long.736/.

YiFan Zhang, Xingyu Lu, Shukang Yin, Chaoyou Fu, Wei Chen, Xiao Hu, Bin Wen, Kaiyu Jiang, Changyi Liu, Tianke Zhang, Haonan Fan, Kaibing Chen, Jiankang Chen, Haojie Ding, Kaiyu Tang, Zhang Zhang, Liang Wang, Fan Yang, Tingting Gao, and Guorui Zhou. Thyme: Think

beyond images. In *NeurIPS 2025 Fourth Workshop on Deep Learning for Code*, 2025a. URL https://openreview.net/forum?id=yFT6rVARsP.

YiFan Zhang, Huanyu Zhang, Haochen Tian, Chaoyou Fu, Shuangqing Zhang, Junfei Wu, et al. Mme-realworld: Could your multimodal llm challenge high-resolution real-world scenarios that are difficult for humans? In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 89655–89701, 2025b. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/df29d63af05cb91d705cf06ba5945b9d-Paper-Conference.pdf.

Shitian Zhao, Haoquan Zhang, Shaoheng Lin, Ming Li, Qilong Wu, Kaipeng Zhang, and Chen Wei. Pyvision: Agentic vision with dynamic tooling. In *First Workshop on Multi-Turn Interactions in Large Language Models*, 2025. URL https://openreview.net/forum?id=l7NYIYODoQ.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization, 2025a. URL https://arxiv.org/abs/2507.18071.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. LlamaFactory: Unified efficient fine-tuning of 100+ language models. In Yixin Cao, Yang Feng, and Deyi Xiong (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pp. 400–410, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-demos.38. URL https://aclanthology.org/2024.acl-demos.38/.

Yaowei Zheng, Junting Lu, Shenzhi Wang, Zhangchi Feng, Dongdong Kuang, Yuwen Xiong, and Richong Zhang. Easyr1: An efficient, scalable, multi-modality rl training framework. https://github.com/hiyouga/EasyR1, 2025b.

A Appendix

A.1 Additional Experiment

Results on OOD Physics Reasoning To further assess the generalization ability of our method beyond the training distribution, we evaluate it on the out-of-distribution (OOD) SeePhys-family physics reasoning benchmarks, including SeePhys and SeePhysPro. As shown in Table 4, our method achieves the highest average score among the compared approaches, demonstrating its effectiveness in handling physics reasoning tasks under distribution shifts. The consistent performance across the two benchmarks suggests that our method can improve the transferability of learned reasoning capabilities to unseen physics-related scenarios, rather than merely exploiting patterns in the training data. These results further highlight the potential of our approach to enhance the robustness of multimodal reasoning in OOD settings.

Table 4: Performance on SeePhys-family physics reasoning benchmarks.

Models	SeePhys		SeePhysPro		AVG
	Vanilla	Vision-only	Level 2	Level 1	
Qwen2.5-VL-7B	11.7	8.5	11.5	12.0	10.9
FAST-7B	**13.4**	**10.1**	11.5	13.0	12.0
PAPO-7B	11.6	9.7	9.0	12.0	10.6
Thyme-RL	10.7	7.6	11.0	11.5	10.2
Monet	11.1	5.8	<u>15.0</u>	12.5	11.1
DeepEyesV2	11.1	8.8	10.5	5.0	8.9
L2T-7B (Ours)	<u>12.9</u>	<u>9.2</u>	**17.0**	**15.0**	**13.5**

Pareto analysis To more intuitively illustrate the advantages of our method over the baselines, we present the results using a Pareto frontier analysis. As shown in Figure 3, compared with the slow-thinking baseline, our method achieves higher accuracy while consuming fewer tokens, demonstrating a more favorable trade-off between reasoning performance and computational efficiency. This suggests that our method can produce more accurate results with less reasoning overhead, improving token efficiency without sacrificing—and potentially even enhancing—reasoning quality.

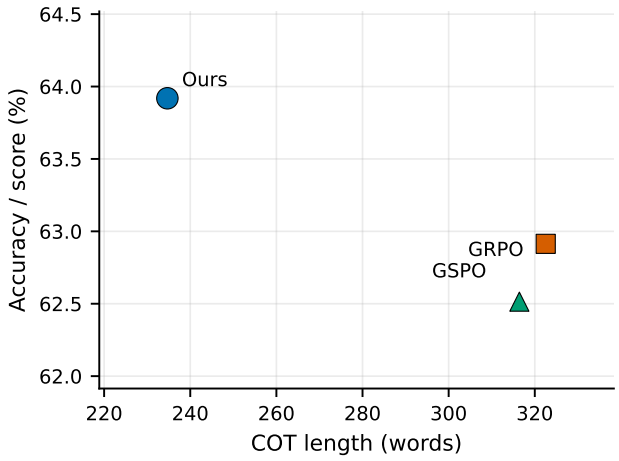


Figure 3: Pareto analysis of our method and slow-thinking RLVR Baseline.

A.2 Analysis of VQ Visual Token

As shown in Figure 4, we introduce new latent visual tokens to facilitate visual-centric reasoning. To verify that these tokens encode meaningful visual information, we conduct a quantitative analysis based on image perturbation and random image replacement. Specifically, we compare the overlap between visual tokens generated from the original image and those generated from its perturbed version, using tokens generated from a randomly replaced image as a reference. If the original and perturbed images yield a higher token overlap than the original and randomly replaced images, this indicates that our visual tokens capture image-specific information and preserve relevant visual representations under perturbations, rather than merely reflecting generic pause token patternsGoyal et al. (2024). We propose a small experiment on 24 examples sampled from HRBench4K using

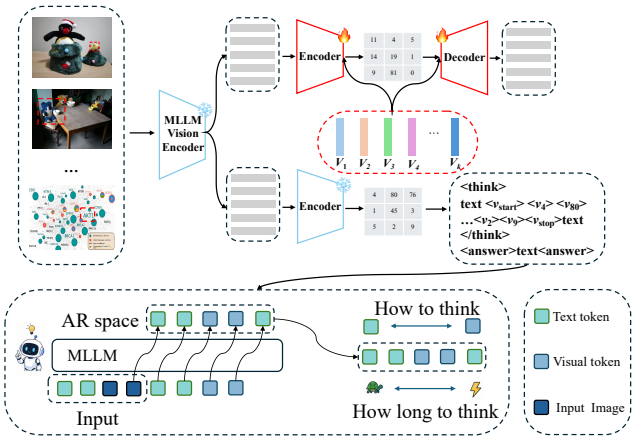


Figure 4: VQ-VT converts tool-produced intermediate visual states into discrete visual-token sequences, enabling visual and textual reasoning to share a unified autoregressive token space for adaptive reasoning depth.

Table 5: VQ code similarity to the original 4K image.

Image condition	Similarity
Repeated original	90.2%
Same image, 2× upscale	84.0%
stretch	83.0%
Horizontal flip	72.8%
Noise + small mask	69.6%
Paired HRBench8K image	67.6%
Unrelated image	41.5%

the model. For each example, we kept the question and prompt fixed and changed only the image. The image conditions included a 2× upscale, horizontal flip, stretch, mild noise with a small mask, and the paired HRBench8K image, which may have a different crop. The unrelated image is the baseline. We also repeated the original image as a stability check.

We compared the generated latent visual token with normalized Levenshtein similarity,

$$S(x, y) = 1 - \frac{d_{\mathrm{Lev}}(x, y)}{\max(|x|, |y|)}, \tag{24}$$

which supports sequences of different lengths of visual token. Higher values mean more similar sequences.

As shown in Table 5, most image changes that preserve the scene produced more similar visual token sequences than replacing the image. Even when replacing the input with a different image from the same question in HRBench 8K, the token overlap remains significantly higher than with a random image. This demonstrates that the latent visual tokens are sensitive to the input image itself and that the model indeed leverages visual information during reasoning.

A.3 Analysis of Difficulty-aware Fast-Slow Thinking

Theoretical Analysis. In the main paper, we restrict the comparison and length penalty to samples that receive the maximum reward, our framework is inherently more flexible. Specifically, it allows different reward thresholds to be specified, enabling even incorrectly routed samples to participate in the comparison while ensuring that samples with incorrect final answers never receive higher rewards than correct ones. This flexibility makes the proposed framework readily extensible to other domains with more complex reward functions.

Let

$$d_{ij} = \tanh\left(\frac{L_i - L_j}{\max(\sigma_L, L_{\min}) + \epsilon}\right). \tag{25}$$

Since the denominator is positive and $\tanh(x) \in (-1, 1)$, we have $|d_{ij}| < 1$. Thus, regardless of the length difference, the magnitude of the compression term is bounded by β.

Let

$$\mathcal{C}_G = \{i : r_i > \tau_{\mathrm{corr}}\}, \qquad \mathcal{F}_G = G \setminus \mathcal{C}_G \tag{26}$$

denote the correct and incorrect subsets of group G. For any $i \in \mathcal{C}_G$ and $k \in \mathcal{F}_G$, we have $r_i > r_k$. Moreover, $c_{ik} = 0$ and $h_{ik} = 1$. Therefore, under the current objective,

$$\Phi_{ik} = \begin{cases} r_i - r_k, & \text{easy or medium group,} \\ r_i - r_k + \beta_{\mathrm{rep}}[d_{ik}]_+, & \text{hard group,} \end{cases} \tag{27}$$

which is strictly positive. Hence, at the pairwise level, neither compression nor repair can reverse the ordering between a correct and an incorrect rollout.

We next consider the aggregated advantage

$$\bar{A}_i = \frac{1}{M-1} \sum_{\substack{j \in G \\ j \neq i}} \Phi_{ij}, \qquad M = |G|. \tag{28}$$

Let $K = |\mathcal{C}_G|$ and define the within-group reward margin

$$\gamma_G = \min_{i \in \mathcal{C}_G} r_i - \max_{k \in \mathcal{F}_G} r_k. \tag{29}$$

For an easy group, only correct–correct pairs receive the length penalty. Consequently, for $i \in \mathcal{C}_G$ and $k \in \mathcal{F}_G$,

$$\bar{A}_i - \bar{A}_k = \frac{M(r_i - r_k)}{M-1} - \frac{\beta}{M-1} \sum_{\ell \in \mathcal{C}_G \setminus \{i\}} d_{i\ell} \tag{30}$$

$$\geq \frac{M\gamma_G - \beta(K-1)}{M-1}. \tag{31}$$

Therefore, a sufficient condition for preserving the ordering after group aggregation is

$$\beta(K-1) < M\gamma_G. \tag{32}$$

Since $K - 1 < M$, the simpler condition $\beta < \gamma_G$ is sufficient for every possible group composition.

In the normalized binary-reward case, where correct rollouts receive $r_i = 1$ and incorrect rollouts satisfy $r_k \leq \tau_{\text{corr}}$, we have $\gamma_G \geq 1 - \tau_{\text{corr}}$. It follows that

$$\beta < 1 - \tau_{\text{corr}} \tag{33}$$

is a group-independent sufficient condition ensuring that every correct rollout receives a larger aggregated advantage than every incorrect rollout. Medium groups contain no length term, while the repair term in hard groups is nonnegative, so their correctness ordering is preserved without this additional restriction.

Finally, the normalized advantage is

$$A_i = \frac{\bar{A}_i - \mu_G}{\sigma_G + \epsilon}. \tag{34}$$

Because $\sigma_G + \epsilon > 0$, this is a strictly increasing affine transformation and therefore preserves the ordering of $\bar{A}_i$.

More generally, for continuous or RLHF-style rewards, let $y_i \in \{0, 1\}$ denote the correctness label and define

$$\gamma = \inf_{i:\, y_i = 1} r_i - \sup_{j:\, y_j = 0} r_j. \tag{35}$$

The condition $\gamma > 0$ means that the primary reward maintains a strict separation between correct and incorrect rollouts.

Consider the generalized pairwise potential

$$\Phi_{ij}^{\text{gen}} = \Delta r_{ij} - \beta_L g_{ij}^L + \sum_{m=1}^{P} \lambda_m g_{ij}^{(m)}, \tag{36}$$

where $\beta_L \geq 0$, $|g_{ij}^L| \leq 1$, and $|g_{ij}^{(m)}| \leq 1$. The auxiliary terms may represent reasoning length, route consistency, response quality, style preference, or normalized RLHF signals. Define their total worst-case influence as

$$B_{\text{aux}} = \beta_L + \sum_{m=1}^{P} |\lambda_m|. \tag{37}$$

For every correct–incorrect pair (i, j), we then have

$$\Phi_{ij}^{\text{gen}} \geq \gamma - B_{\text{aux}}. \tag{38}$$

Therefore, the pairwise correctness ordering is preserved whenever

$$B_{\text{aux}} < \gamma. \tag{39}$$

In the normalized unit-correctness-reward case, where correct rollouts receive reward 1 and incorrect rollouts receive at most τ_{corr}, we have

$$\gamma \geq 1 - \tau_{\text{corr}}. \tag{40}$$

Consequently, the group-independent condition

$$\beta_L + \sum_{m=1}^{P} |\lambda_m| < 1 - \tau_{\text{corr}} \tag{41}$$

is sufficient to preserve pairwise correctness ordering.

This result interprets the correctness margin as an auxiliary reward budget. The relative magnitudes of β_L and λ_m determine the importance of length efficiency and other preferences, whereas their total worst-case influence remains below the correctness margin. Thus, auxiliary objectives may rank rollouts within the same correctness level without making an incorrect rollout preferable to a correct one.

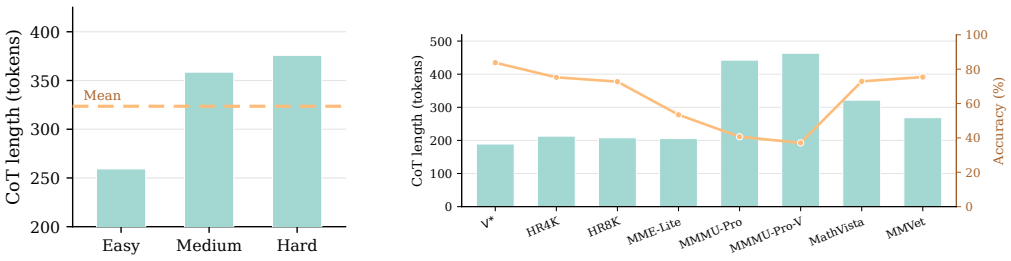
For arbitrary nonseparable pairwise shaping terms, a conservative group-independent condition for preserving the aggregated ordering is $B_{\text{aux}} < \gamma/2$.

Ablation on Thresholds We further evaluate the importance of difficulty discrimination based on the pass-rate threshold. In this ablation, we remove the threshold and instead adopt a strategy similar to previous work, where the shortest correct response is treated as the optimization target Yang et al. (2025b;a); Wang et al. (2026c). As shown in Table 6, removing the threshold leads to a degradation in performance. This finding indicates that distinguishing tasks of different difficulty is beneficial. In the next section, we further investigate the impact of slow thinking on tasks with varying levels of difficulty.

Table 6: Ablation on thresholds.

Components	Visual-Centric	Reasoning-Centric	Total Δ
w/o threshold	70.5	56.0	-1.2
standard	71.2	56.5	–

A.4 Effectiveness of Long CoT



(a) The average response length in different complexity.

(b) Overview of modality-aware routing and difficulty-aware fast-slow optimization.

Figure 5: Throughput and generation length comparison.

A.4.1 Effectiveness of Fast-slow thinking.

To investigate whether our model learns difficulty-aware reasoning, we analyze CoT length distributions under different difficulty levels. Samples are categorized by the correctness of our model and Qwen2.5-VL: both correct as easy, only one correct as medium, and both incorrect as hard. As shown in Figure 5a, our model adaptively allocates reasoning budgets, generating the shortest CoTs for easy samples and the longest for hard samples. Medium samples receive moderate additional

reasoning, suggesting that extra computation is beneficial when deeper reasoning is required. These results demonstrate that L2T learns adaptive reasoning allocation rather than simply favoring shorter or longer CoTs.

A.4.2 CoT Length Across Benchmark

We further analyze the relationship between CoT length and model performance across different benchmarks. As shown in Figure. 5b, CoT length exhibits different patterns between perception- and reasoning-oriented tasks. On perception-centric benchmarks, we observe no clear correlation between reasoning length and accuracy, suggesting that increasing textual reasoning alone provides limited benefits when failures primarily originate from visual perception errors. In contrast, on reasoning-centric benchmarks, incorrect samples generally produce longer CoTs, indicating that challenging reasoning problems naturally require more extensive exploration and self-reflection. These observations highlight that different task types benefit from different forms of computation: visual tasks require effective perceptual computation, while reasoning tasks can benefit more from additional reasoning depth.

A.4.3 Ablation on Collapsed CoT

To further investigate the role of CoT reasoning, we increase the number of Stage 3 training steps and remove the online filtering strategy, allowing a larger proportion of easy samples to dominate optimization and encouraging progressively shorter reasoning traces. As shown in Table 7, under this setting, the CoT eventually collapses into a nearly fixed template of only about 10 tokens. Compared with the standard model, performance on perception-oriented benchmarks decreases by only 2.6%, whereas reasoning-oriented benchmarks suffer a much larger drop of 12.2%. These results suggest that long CoT provides limited benefit for perception tasks but remains crucial for reasoning-intensive problems.

Table 7: Ablation on Route Reward

Category	standard (%)	overtraining (%)	Δ
Average Length of CoT			
Visual-Centric	206.8	10	+196.8
Reasoning-Centric	416.2	10	+406.2
Accuracy			
Visual-Centric	71.2	68.6	+2.6
Reasoning-Centric	56 .5	44.3	+12.2

A.4.4 CoT Length Across Failure Modes

We further analyze the failure cases by categorizing the evaluation samples into correct predictions and three error types: perception failure, knowledge deficiency, and reasoning failure. We use GPT-5.5 to categorize the types of errorsOpenAI (2026). Table 8 summarizes 8,230 responses. Interestingly, both the median and mean reasoning lengths of perception failures and knowledge-deficiency cases are comparable to those of correctly solved samples. Moreover, these failures typically emerge at the early stages of reasoning, rather than after extended deliberation. In contrast, reasoning failures are associated with substantially longer CoT traces and usually occur at later stages of the reasoning process. We conjecture that perception failures and knowledge deficiencies primarily stem from limitations in visual perception or internal knowledge, and therefore cannot be effectively remedied by simply extending the reasoning process. This observation also indicates that using rollout pass rate alone as a proxy for task difficulty is insufficient for adaptive CoT allocation, as it may unnecessarily encourage longer reasoning on tasks that are unlikely to benefit from additional computation. We leave a more fine-grained difficulty estimation strategy for future work.

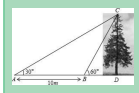


Question:
What is the number displayed above the entrance where the woman is standing?
A. 27B B. 37B
C. 27D D. 27E

<think>The number above the entrance is clearly visible in the image, and there is no ambiguity in its display. The number "27B" is legible, and no further enhancement is needed. However, to ensure clarity and confirm the number, a slight zoom and crop on the area above the entrance would be beneficial. This will help in isolating the number and removing any potential distractions from the rest of the image.
<v_start>...<v_end> Upon examining the processed image, the number displayed above the entrance is clearly visible. The number "27B" is legible and distinct. There are no other numbers or letters that could be mistaken for the correct answer. The slight zoom and crop on the area above the entrance confirm the number without any ambiguity.</think>

<answer> A </answer> ✓

(a) Case study on visual sample.



As shown in the figure, a teaching interest group wants to measure the height of a tree CD. They firstly measured the elevation angle of the tree top C at point A as 30.0, and then proceeded 10.0 along the direction of AD to point B, and the elevation angle of tree top C measured at B is 60.0 (the three points A, B, and D are on the same straight line), then the height of the tree CD is ()
(A) 10m (B) 5m
(C) 5√{3}m (D) 10√{3}m

<think>The image provided is clear and does not require further enhancement. The relevant geometric relationships and angles are visible, allowing for the solution of the problem. Therefore, no cropping or zooming is needed.
To solve the problem, we can model the scenario using trigonometry. Let's denote the height of the tree CD as \(h \). The points A, B, and D are collinear, with AB = 10 meters. The elevation angles from points A and B to the top of the tree C are 30 degrees and 60 degrees, respectively.
First, consider the triangle formed by points A, D, and C. The angle of elevation from A to C is 30 degrees. ... Therefore, the height of the tree CD is \(5\sqrt{3}\) meters. The correct option is (C).
Since the problem is already solved and the image is clear, no further adjustments are needed.</think>

<answer> (C) 5√{3}m </answer> ✓

(b) Case study on reasoning sample.

Figure 6: Successful case study of our method.

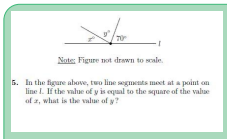


Answer the following multiple-choice question in the image.
Question in Figure: This figure shows the _____.
(A) Baddeley's model of working memory; (B) Divided attention model; (C) selective priming; (D) Deutsch and Deutsch's Late Selection Model; (E) cocktail party effect; (F) Triesman's Feature Integration Theory; (G) John's Interference Theory; (H) Anne Treisman's Attentuation Model; (I) Broadbent Filter Model of Selective Attention; (J) Posner's attention model

<think>The image provided is clear and does not require further enhancement. The relevant text and diagram are easily readable, and there are no issues with resolution or clarity. Therefore, no cropping or zooming is needed.
To answer the question, we need to identify which model the diagram represents. The diagram shows a flow from sensory input to working memory, with a selective filter and a bottleneck. This aligns with the Broadbent Filter Model of Selective Attention, which is option (I).
Therefore, the correct answer is (I) Broadbent Filter Model of Selective Attention.
Since the image is already clear, no further processing is needed.<think>

<answer> (I) Broadbent Filter Model of Selective Attention </answer> ✓

(a) Case study on interleaved sample.



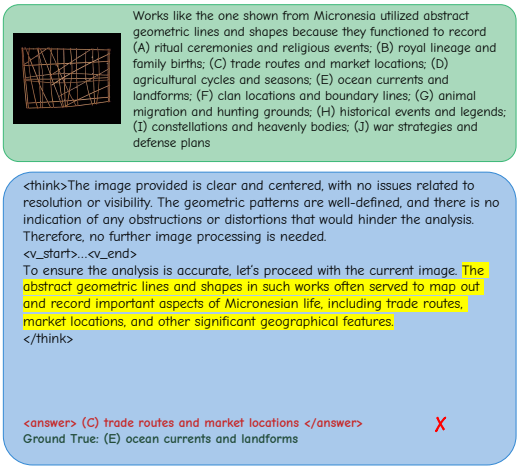
In the figure above, two line segments meet at a point on line l. If the value of y is equal to the square of the value of x, what is the value of y?

<think>The image provided is clear and does not require further enhancement. The angles and the note are visible, and there are no issues with scale or clarity. Therefore, no cropping or zooming is needed.
<v_start>...<v_end>
Based on the geometric diagram, the two line segments meet at a point on line \(l \), forming two adjacent angles, \(x^\circ \) and \(y^\circ \), and a third angle of \(70^\circ\). The sum of these three angles should be equal to \(180^\circ\) since they lie on a straight line.
So, we can set up the equation:...
Given that \(y = x^2 \):
\[y = 10^2 = 100 \]
Thus, the value of \(y \) is 100.

<answer> 100</answer> ✓

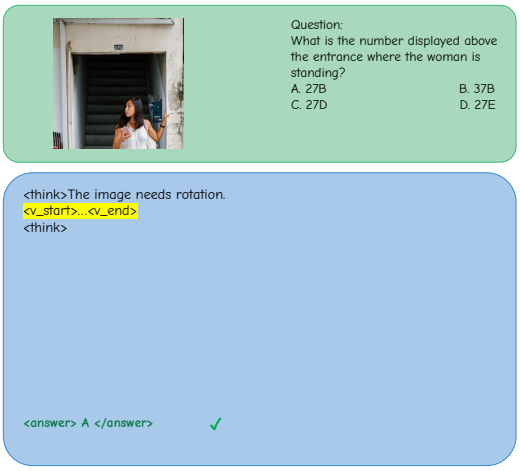
(b) Case study on knowledge-based sample.

Figure 7: Successful case study of our method.



(a) Case study on knowledge-based sample. (b) Case study on visual sample.

Figure 8: Failed case study of our method.



(a) Case study on visual sample. (b) Case study on reasoning sample.

Figure 9: Case study on Collapsed CoT.

Table 8: Mean and median CoT length for each answer outcome and failure mode.

Outcome / failure mode	n	Mean	Median
Correct-all	4,573	277.3	214
Correct-Perception	2,370	200.6	198
Correct-Reasoning	2,203	359.9	260
Perception failure	1,758	241.9	216
Knowledge failure	517	430.0	281
Reasoning-process failure	724	505.8	304.5
Mixed failure	302	461.9	308
Correct CoT, wrong output	262	356.5	244
Termination/completeness failure	94	1,644.6	2,046

A.5 Case Study

Successful Case In this section, we present representative successful cases. Figure 6a illustrates a perception-oriented task in which the model correctly identifies the task type and generates appropriate visual tokens to facilitate visual perception. Figure 6b shows a reasoning-oriented example that is solved solely through textual reasoning without invoking visual tokens. Figure 7a presents a knowledge-based question, where the model answers directly from its internal knowledge without generating either visual tokens or unnecessary textual reasoning. Figure 7b demonstrates a task requiring both perception and reasoning, where the model first employs visual tokens to enhance visual representations and subsequently performs textual reasoning to derive the final answer.

Failed Case In this section, we analyze representative failure cases. In Figure 8a, the question primarily depends on the model's internal knowledge and therefore does not require additional reasoning. However, the model unnecessarily generates visual tokens and ultimately produces an incorrect answer due to insufficient internal knowledge. In Figure 8b, the model makes an incorrect perception at the beginning of the reasoning process. Although it subsequently generates visual tokens to re-examine the image, it fails to correct the initial perceptual error.

These cases suggest that extending the reasoning process provides limited benefit for correcting knowledge deficiencies or perceptual errors, consistent with our quantitative observations that such failure cases exhibit reasoning lengths comparable to those of correctly solved samples. We attribute these failures primarily to intrinsic limitations of the underlying model, namely insufficient domain knowledge and inadequate visual perception capabilities. When the required prior knowledge is absent, textual reasoning alone cannot compensate for the missing information to produce correct answers. Likewise, when visual perception is fundamentally inadequate, reprocessing images through the thinking-with-image paradigm is unlikely to recover the missing visual evidence. Therefore, future improvements should focus on enriching the model's foundational knowledge and strengthening the visual perception capability of the vision encoder, rather than relying solely on more sophisticated reasoning strategies.

Case on Collapsed CoT In this section, we investigate the behavior of the model after overtraining, where the CoT collapses into a short, fixed reasoning template. We revisit the previous case studies for comparison. Figure 6a and Figure 9a present the model's responses to a perception-oriented task before and after CoT collapse, respectively. The model correctly answers the question in both cases. In contrast, Figure 6b and Figure 9b compare the responses to a reasoning-intensive task before and after CoT collapse. Once the reasoning process is lost, the model fails to produce the correct answer. These observations suggest that different task types exhibit different levels of dependence on long CoT reasoning. In particular, reasoning-intensive tasks are substantially more affected by CoT collapse than perception-oriented tasks.

A.6 Pseudo Code

In this section, we present the pseudocode of the proposed algorithm to provide a clearer and more intuitive overview of the overall training procedure.

Algorithm 1 VQ-VT Training

Require: Images $\mathcal{D}_{\text{img}}$; frozen E_{vis}; trainable E_ω and D_ϕ
Require: Codebook $\mathcal{V} = \{e_j\}_{j=1}^{K}$; coefficient β
Ensure: Quantizer $Q \circ E_\omega$ and K visual tokens

while the VQ-VT objective has not converged **do**
  Sample a minibatch of images $I \sim \mathcal{D}_{\text{img}}$
  $\{h_i\}_{i=1}^{N} \leftarrow E_{\text{vis}}(I)$ ▷ frozen patch features
  for $i = 1, \dots, N$ **do**
    $z_i \leftarrow \text{Normalize}_{\ell_2}(E_\omega(h_i))$
    for $j = 1, \dots, K$ **do**
      $\bar{e}_j \leftarrow \text{Normalize}_{\ell_2}(e_j)$
    $k_i \leftarrow \arg\max_{1 \le j \le K} z_i^\top \bar{e}_j$
    $q_i \leftarrow \bar{e}_{k_i}$ ▷ Eq. (1)
    $q_i^{\text{ST}} \leftarrow z_i + \text{sg}(q_i - z_i)$ ▷ STE
  $\ell_i^{\text{rec}} \leftarrow \|h_i - D_\phi(q_i^{\text{ST}})\|_2^2$
  $\ell_i^{\text{code}} \leftarrow \|\text{sg}(z_i) - q_i\|_2^2$
  $\ell_i^{\text{com}} \leftarrow \beta \|z_i - \text{sg}(q_i)\|_2^2$
  $\ell_i \leftarrow \ell_i^{\text{rec}} + \ell_i^{\text{code}} + \ell_i^{\text{com}}$
  $\mathcal{L}_{\text{VQ}} \leftarrow \dfrac{1}{N} \sum_{i=1}^{N} \ell_i$ ▷ Eq. (2)
  Update E_ω, D_ϕ, and $\mathcal{V}$ by descending $\nabla \mathcal{L}_{\text{VQ}}$
Add $\{\texttt{<vstart>}, \texttt{<v1>}, \dots, \texttt{<vK>}, \texttt{<vend>}\}$ to the MLLM vocabulary
return E_ω, Q, D_ϕ and the extended vocabulary

Algorithm 2 Tool-Trajectory Discretization

Require: Tool data $\mathcal{D}_{\text{tool}}$; VQ-VT encoder E_ω; quantizer Q
Ensure: Discrete visual-reasoning dataset $\mathcal{D}_{\text{VT}}$

$\mathcal{D}_{\text{VT}} \leftarrow \varnothing$
for all $(X, I, C, A; t_{\text{pre}}, t_{\text{post}}) \in \mathcal{D}_{\text{tool}}$ **do**
  $I' \leftarrow \textsc{Execute}(C, I)$ ▷ Eq. (3)
  $(v_1, \dots, v_m) \leftarrow Q(E_\omega(I'))$ ▷ Eq. (4)
  $\mathbf{v} \leftarrow (\texttt{<vstart>}, v_1, \dots, v_m, \texttt{<vend>})$
  $d \leftarrow (X, t_{\text{pre}}, \mathbf{v}, t_{\text{post}}, A)$ ▷ Eq. (5)
  Add d to $\mathcal{D}_{\text{VT}}$
return $\mathcal{D}_{\text{VT}}$

Algorithm 3 Three-Stage Adaptive Training

Require: Policy π_θ; visual data $\mathcal{D}_{\mathrm{VT}}$; RL data $\mathcal{D}_{\mathrm{RL}}$
Require: Group size M; route and length hyperparameters
Ensure: Adaptively trained policy π_θ

Stage 1: visual supervised fine-tuning.
while the SFT objective has not converged **do**
  Sample a minibatch $(X, I, s) \sim \mathcal{D}_{\mathrm{VT}}$
  $\mathcal{L}_{\mathrm{SFT}} \leftarrow -\sum_k \log \pi_\theta(s_k \mid s_{<k}, X, I)$ ▷ Eq. (6)
  Update θ by descending $\nabla \mathcal{L}_{\mathrm{SFT}}$
Stage 2: stable-reward GRPO.
while the stable reasoning policy has not converged **do**
  Sample $(x, a^*) \sim \mathcal{D}_{\mathrm{RL}}$ and draw $G = \{y_i\}_{i=1}^M \sim \pi_\theta(\cdot \mid x)$
  for all $y_i \in G$ **do**
    $(z_i, \hat{a}_i) \leftarrow \textsc{ParseResponse}(y_i)$
    $r_{\mathrm{acc},i} \leftarrow \mathbb{I}[\textsc{Verify}(\hat{a}_i, a^*)]$
    $r_{\mathrm{fmt},i} \leftarrow \mathbb{I}[y_i \in \mathcal{F}]$
    $r_{\mathrm{stable},i} \leftarrow r_{\mathrm{acc},i} r_{\mathrm{fmt},i}$ ▷ Eqs. (7)–(9)
  Update π_θ by standard GRPO using $\{r_{\mathrm{stable},i}\}_{i=1}^M$
Stage 3: routing + fast–slow GRPO.
while the adaptive policy has not converged **do**
  Sample $(x, a^*, n_v, n_t) \sim \mathcal{D}_{\mathrm{RL}}$ and draw $G = \{y_i\}_{i=1}^M \sim \pi_\theta(\cdot \mid x)$
  for all $y_i \in G$ **do**
    $(R_i, L_i) \leftarrow \textsc{RoutingReward}(y_i, a^*, n_v, n_t)$ ▷ Alg. 4
  $\mathbf{A} \leftarrow \textsc{FastSlowAdvantage}(\{R_i, L_i\}_{i=1}^M)$ ▷ Alg. 5
  $\mathcal{L}_{\mathrm{GRPO}} \leftarrow \textsc{GRPOObjective}(x, G, \mathbf{A})$ ▷ Alg. 6
  Update θ by descending $\nabla \mathcal{L}_{\mathrm{GRPO}}$
return π_θ

Algorithm 4 Modality-Aware Routing Reward

function $\textsc{RoutingReward}(y, a^*, n_v, n_t)$
Require: Thresholds $\tau_v, \tau_t, \tau_v^{\mathrm{mix}}, \tau_t^{\mathrm{mix}}, V_{\min}, T_{\min}$
Require: Penalties $\lambda_{\mathrm{novis}}, \lambda_{\mathrm{simple}}$
  $(z, \hat{a}) \leftarrow \textsc{ParseResponse}(y)$
  $V \leftarrow V(z)$; $T \leftarrow T(z)$; $L \leftarrow V + T$
  $\rho_v \leftarrow V/(V+T)$; $\rho_t \leftarrow T/(V+T)$ ▷ Eq. (10)
  ▷ A valid trace has $V + T > 0$.
  $\mathcal{S}(\rho, N; \tau, N_0) \leftarrow \min(\rho/\tau, 1)\min(N/N_0, 1)$
  $s_v \leftarrow \mathcal{S}(\rho_v, V; \tau_v, V_{\min})$ ▷ Eq. (11)
  $s_t \leftarrow \mathcal{S}(\rho_t, T; \tau_t, T_{\min})$ ▷ Eq. (12)
  $s_v^{\mathrm{mix}} \leftarrow \mathcal{S}(\rho_v, V; \tau_v^{\mathrm{mix}}, V_{\min})$ ▷ Eq. (13)
  $s_t^{\mathrm{mix}} \leftarrow \mathcal{S}(\rho_t, T; \tau_t^{\mathrm{mix}}, T_{\min})$ ▷ Eq. (14)
  if $n_v = 1$ and $n_t = 0$ **then**
    $s_{\mathrm{route}} \leftarrow s_v$
  else if $n_v = 0$ and $n_t = 1$ **then**
    $s_{\mathrm{route}} \leftarrow s_t \lambda_{\mathrm{novis}}^{\mathbb{I}[V>0]}$
  else if $n_v = 1$ and $n_t = 1$ **then**
    $s_{\mathrm{route}} \leftarrow \min(s_v^{\mathrm{mix}}, s_t^{\mathrm{mix}})$
  else
    $s_{\mathrm{route}} \leftarrow \lambda_{\mathrm{simple}}^{\mathbb{I}[V>0]}$
  ▷ Four routing types in Eq. (15)
  $r_{\mathrm{acc}} \leftarrow \mathbb{I}[\textsc{Verify}(\hat{a}, a^*)]$
  $r_{\mathrm{fmt}} \leftarrow \mathbb{I}[y \in \mathcal{F}]$
  $R \leftarrow (0.9 r_{\mathrm{acc}} + 0.1 s_{\mathrm{route}}) r_{\mathrm{fmt}}$ ▷ Eq. (16)
  return (R, L)

Algorithm 5 Difficulty-Aware Fast–Slow Advantage

function FASTSLOWADVANTAGE($\{R_i, L_i\}_{i=1}^{M}$)
Require: $\tau_{\text{corr}} = 0.99$; $\tau_{\text{easy}}, \tau_{\text{hard}}$
Require: Length floor $L_{\min}$; coefficients β, β'
for $i = 1, \dots, M$ **do**
  $q_i \leftarrow \mathbb{I}[R_i > \tau_{\text{corr}}]$
$p_G \leftarrow M^{-1} \sum_{i=1}^{M} q_i$ ▷ Eq. (17)
$\sigma_L \leftarrow \text{Std}(\{L_i\}_{i=1}^{M})$; $d_L \leftarrow \max(\sigma_L, L_{\min})$
for all ordered pairs (i, j) with $i \neq j$ **do**
  $\Delta R_{ij} \leftarrow R_i - R_j$
  $\Delta L_{ij} \leftarrow \tanh((L_i - L_j)/d_L)$ ▷ Eq. (18)
  $c_{ij} \leftarrow q_i q_j$; $h_{ij} \leftarrow q_i(1 - q_j)$ ▷ Eqs. (19)–(20)
  if $p_G \geq \tau_{\text{easy}}$ **then**
    $\Phi_{ij} \leftarrow \Delta R_{ij} - \beta \Delta L_{ij} c_{ij}$ ▷ easy
  else if $p_G \geq \tau_{\text{hard}}$ **then**
    $\Phi_{ij} \leftarrow \Delta R_{ij}$ ▷ medium
  else
    $\Phi_{ij} \leftarrow \Delta R_{ij} + \beta' [\Delta L_{ij}]_+ h_{ij}$ ▷ hard
▷ Pairwise potential in Eq. (21)
for $i = 1, \dots, M$ **do**
  $A_i \leftarrow \text{Norm}_G \left(\frac{1}{M-1} \sum_{j \neq i} \Phi_{ij} \right)$ ▷ Eq. (22)
return $\{A_i\}_{i=1}^{M}$

Algorithm 6 Clipped GRPO Update

function GRPOOBJECTIVE($x, G, \mathbf{A}$)
Require: Policies $\pi_\theta, \pi_{\theta_{\text{old}}}, \pi_{\text{ref}}$
Require: Clip radius ϵ; KL weight λ_{KL}
for all $y_{i,t}$ in rollout $y_i \in G$ **do**
  $p_{i,t} \leftarrow \pi_\theta(y_{i,t} \mid x, y_{i,<t})$
  $p_{i,t}^{\text{old}} \leftarrow \pi_{\theta_{\text{old}}}(y_{i,t} \mid x, y_{i,<t})$
  $\rho_{i,t} \leftarrow p_{i,t} / p_{i,t}^{\text{old}}$
  $g_{i,t}^{\text{raw}} \leftarrow \rho_{i,t} A_i$
  $\bar{\rho}_{i,t} \leftarrow \text{clip}(\rho_{i,t}, 1 - \epsilon, 1 + \epsilon)$
  $g_{i,t}^{\text{clip}} \leftarrow \bar{\rho}_{i,t} A_i$
  $g_{i,t} \leftarrow \min(g_{i,t}^{\text{raw}}, g_{i,t}^{\text{clip}})$
for $i = 1, \dots, M$ **do**
  $\bar{g}_i \leftarrow \frac{1}{|y_i|} \sum_t g_{i,t}$
$\mathcal{L}_{\text{policy}} \leftarrow -\frac{1}{M} \sum_{i=1}^{M} \bar{g}_i$
$K \leftarrow D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}})$
$\mathcal{L}_{\text{GRPO}} \leftarrow \mathcal{L}_{\text{policy}} + \lambda_{\text{KL}} K$ ▷ Eq. (23)
return $\mathcal{L}_{\text{GRPO}}$

A.7 Experimental Details

A.7.1 overall

Training Data and Setup. Our experiments are based on Qwen2.5-VL Qwen Team (2025) .Our SFT data consists of Thyme-SFT Zhang et al. (2025a), Zebra-CoT Li et al. (2025) comes from Monet-SFT Wang et al. (2026b), and Vision-R1-SFT Huang et al. (2026). For Thyme-SFT and Monet-SFT, the intermediate auxiliary reasoning images are converted into discrete visual tokens using VQ-VT, and the model is trained with LLaMA-Factory Zheng et al. (2024). We further mix these data with the Mulberry dataset Yao et al. (2026) from Vision-R1-SFT Huang et al. (2026). For reinforcement learning, we construct the RL dataset by combining Thyme-RL Zhang et al. (2025a), ViRL39K Wang et al. (2026a), and Geo3K Lu et al. (2021), and train the model using EasyR1 Zheng et al. (2025b). During RL training, we enable online filtering to ensure a consistent rollout batch size throughout training, we optimize the policy using GRPO DeepSeek-AI (2025) in both Stage 2 and Stage 3. The SFT stage was conducted on 8 Nvidia H200 GPUs, while the RL and evaluation stages were conducted on 8 Nvidia H100 GPUs.

During the SFT stage, the models use 1,024 visual tokens. The vision tower and multimodal projector are frozen, while only the language model is trained. We adopt a conversational template consisting of system and user prompts, with the loss computed exclusively on the completion. The maximum sequence length is set to 8,196. The per-device training batch size is 8, with one gradient accumulation step, resulting in an effective global batch size of 64. We use the AdamW optimizer with a learning rate of 1×10^{-5}, a cosine learning rate scheduler, and two training epochs. Training is performed using BF16 precision and DeepSpeed ZeRO-3.

During the RL stage, we employ the GRPO algorithm with a training batch size of 256. Each prompt generates 16 candidate responses, and the PPO mini-batch size is set to 128. The maximum prompt and response lengths are both 4,096 tokens. We use the AdamW optimizer with a learning rate of 5×10^{-6} and a weight decay of 0.01, enabling KL regularization with a coefficient of 0.01 and online filtering. Stage 2 and Stage 3 are trained for 90 and 30 steps, respectively. vLLM is used for inference and both the generation temperature and top-p set to 0.9.

For the reward mechanism and thinking strategy, the visual and textual components of the route reward are assigned weights of 30% and 70%, respectively, while the corresponding mixed reward weights are 10% and 20%. The no-vision penalty coefficient is set to 0.50, the simple-sample penalty coefficient to 0.70, the routing coefficient to 0.10, and the budget coefficient to 1.00. In the difficulty-aware fast–slow thinking mechanism, the thresholds for easy and hard samples are set to 0.80 and 0.50, respectively. The minimum thinking length is 50, and the pass threshold is set to 0.99.

Evaluation Setup. We conduct all evaluations using the VLMEvalKit framework Duan et al. (2024) and locally deploy Qwen3-30B-A3B Team (2025) as the judge model. Our evaluation covers three groups of benchmarks: Visual-Centric Perception, including V^* Wu & Xie (2024), HRBench-4K Wang et al. (2025), HRBench-8K Wang et al. (2025), and MME-RealWorld-Lite Zhang et al. (2025b); Reasoning-Centric Understanding, including $\text{MathVista}_{\text{mini}}$ Lu et al. (2024), MMMU-Pro-10-option Yue et al. (2025), MMMU-Pro-V Yue et al. (2025), and MMVet Yu et al. (2024); and Physics-Reasoning benchmark (Out-of-Distribution), including SeePhys Xiang et al. (2025) and SeePhysPro Xiang et al. (2026b).

A.7.2 Datasets

We construct two datasets for different training stages: one for SFT and the other for RL.

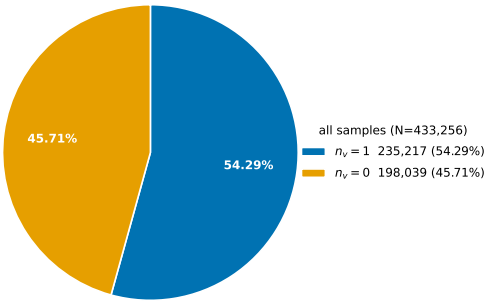
SFT dataset The SFT dataset is built from Thyme-SFT Zhang et al. (2025a) ('2round', 'wo thinking thyme single round'), Zebra-CoT Li et al. (2025) (from Monet-SFT Wang et al. (2026b)), and Mulberry Yao et al. (2026) (from Vision-R1 SFT Huang et al. (2026)). We convert the intermediate visual trajectories in Thyme-RL and Zebra-CoT into discrete visual tokens using VQ-VT, while Mulberry is kept unchanged since it does not contain tool-produced visual states. As illustrated in Figure 10a, we maintain an approximately balanced distribution, with around half of the training

samples containing visual tokens. This design encourages the model to learn when to generate visual tokens without biasing it toward producing them for every sample.

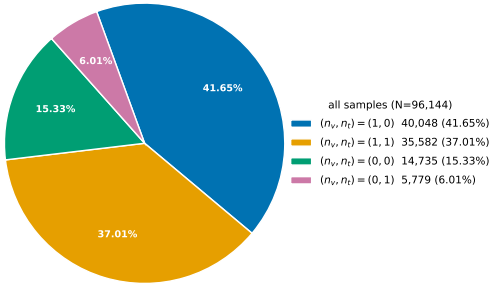
RL dataset The RL dataset consists of Thyme-RL Zhang et al. (2025a), ViRL39K Wang et al. (2026a), and Geo3K Lu et al. (2021). We first assign coarse routing labels according to the dataset source and the task categories provided in ViRL39K, and then further refine these labels using rule-based keyword matching. This lightweight annotation strategy provides a practical initialization for modality-aware routing without requiring expensive manual labeling.

As shown in Figure 10b, since Thyme-RL contains substantially more samples than the other datasets, the resulting routing labels naturally exhibit a higher proportion of visual-token supervision. In addition, we observe that Thyme-RL includes many relatively simple perception tasks, which rapidly converge during RL training, we shown the result in Figure 11. For example, after only 30 training steps, approximately 75% of the sampled groups are either entirely correct or entirely incorrect, leaving only 25% that contribute effective optimization signals. In contrast, the more reasoning-oriented ViRL39K and Geo3K datasets still retain around 40% informative groups even after 175 training steps.

To alleviate this imbalance, we employ an online filtering strategy that automatically removes fully solved perception samples from optimization. Consequently, RL updates are no longer dominated by easy perception tasks, allowing the model to focus on more informative training signals. More importantly, this prevents the policy from overemphasizing visual-token generation simply due to dataset composition, encouraging a more balanced allocation between visual and textual reasoning according to task requirements rather than sample frequency.



(a) Pie chart of the SFT dataset.

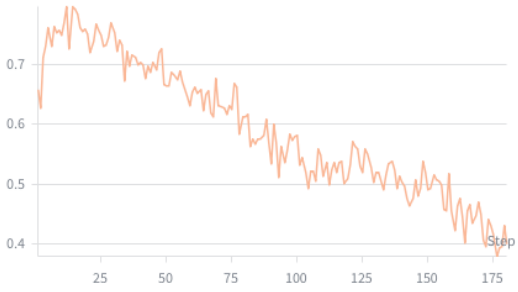


(b) Pie chart of the RL dataset.

Figure 10: Pie chart of the datasets.



(a) Remaining ratio of informative rollout groups after online filtering during RL training on Thyme-RL.



(b) Remaining ratio of informative rollout groups after online filtering during RL training on ViRL39K+Geo3k.

Figure 11: Remaining ratio of informative rollout groups after online filtering during RL training.

A.8 LIMITATIONS

Our method still has several limitations. The visual codebook compresses intermediate visual states into discrete visual tokens, inevitably sacrificing part of the fine-grained visual information. More importantly, although visual tokens improve reasoning performance by encouraging the model to re-examine visual evidence, they cannot fundamentally correct perception errors caused by the limited representation capability of the underlying vision encoder. As demonstrated in our case studies and error analysis, once incorrect visual evidence is encoded, subsequent reasoning—whether visual or textual—has limited ability to recover from these early perception failures. This suggests that improving reasoning alone cannot fully compensate for weak visual perception, and future work should jointly enhance perceptual representation learning and reasoning.