# Beyond Static Visual Tokens: Structured Sequential Visual Chain-of-Thought Reasoning

**Anonymous submission**


## Abstract

Current multimodal LLMs encode images as static visual prefixes and rely on text-based reasoning, lacking goal-driven and adaptive visual access. Inspired by human visual perception—where attention is selectively and sequentially shifted from the most informative regions to secondary cues—we propose Structural Sequential Visual CoT (**SSV-CoT**). First, a question-relevant saliency map identifies and organizes key visual regions, explicitly modeling the spatial distribution of visual importance. Second, reasoning is performed following this discriminative order, inducing a curriculum-like semantic progression from primary to secondary cues. This method is trained end-to-end, using text cot and answer supervision, without relying on region-level annotations or specialized external tools. Experiments on diverse visual reasoning benchmarks show gains, validating structured and sequential visual cognition.


## 1 Introduction

Despite rapid advances in multimodal LLMs (Alayrac et al., 2022; Liu et al., 2023; Li et al., 2023; Dai et al., 2023), current systems still suffer from a fundamental limitation in how vision is integrated into reasoning. Most architectures encode an image once into visual tokens or a global embedding and inject it into the LLM as static context (Alayrac et al., 2022; Li et al., 2023; Liu et al., 2023). Reasoning then unfolds purely in text space: as the chain-of-thought grows, the model increasingly ignores the image, cannot revisit or operate on visual content, and lacks goal-directed visual actions such as selecting, refocusing, or comparing regions. As a result, visual perception and language reasoning remain largely disentangled, and vision functions as a passive prefix rather than an active part of cognition.

Existing attempts to mitigate this gap face significant drawbacks. External tool pipelines introduce cropping or detection but require complex orchestration (Wu et al., 2023; Chen et al., 2023). Region-based supervision depends on costly annotations (Lin et al., 2022; Khan et al., 2022). Generative visual-token or image synthesis is inefficient and poorly aligned with reasoning needs (Esser et al., 2021; Ding et al., 2022). Latent-space alignment methods learn global correspondences yet provide no explicit visual access steps (Radford et al., 2021; Li et al., 2023). Interleaved chain-of-thought methods re-inject visual tokens but enable passive re-attention over a large patch space, without modeling where to look next (Zhang et al., 2024b). None of these approaches support internal, sequential, goal-conditioned visual cognition.

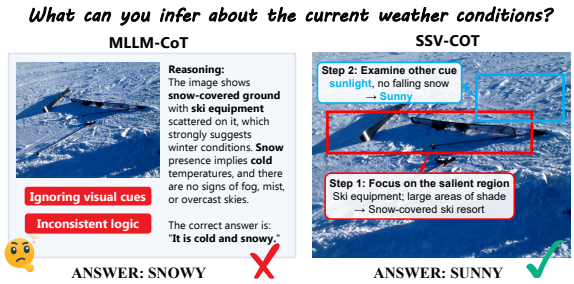



Figure 1: MLLM lacks visual analysis, while SSV integrates cues through structured sequential reasoning to reach the correct answer.

As illustrated in Fig. 1, humans rarely solve visual tasks by processing an entire scene uniformly. Instead, they first form a coarse global impression and then sequentially attend to task-relevant regions as reasoning unfolds, a pattern well documented in studies of human visual attention and eye movements (Yarbus, 1967; Itti et al., 1998). In the example, the left approach (MLLM-CoT) fixates on the presence of snow-covered ground and ski equipment, ignores critical cues such as strong sunlight and sharp shadows, and arrives at an incorrect conclusion. In contrast, the right approach (SSV-CoT) explicitly guides visual attention through a structured sequence: it first focuses on salient re-

gions (e.g., ski equipment and shadows) and then integrates additional cues (e.g., lighting conditions) to refine the hypothesis, ultimately reaching the correct answer. This comparison highlights a fundamental limitation of existing multimodal large language models, which typically compress images into a fixed set of visual tokens and perform reasoning purely in text space (Alayrac et al., 2022; Li et al., 2023). While recent work on chain-of-thought reasoning has shown the benefits of explicit intermediate reasoning in language models (Wei et al., 2022), visual perception in current MLLMs remains largely passive and decoupled from the reasoning process. This disconnect between perception and cognition motivates our central question: can a model learn not only *what* to look at, but also *when* and *in which order* to look, driven directly by its ongoing reasoning?

Inspired by human visual cognition, we propose **SSV-CoT**, a multimodal reasoning framework that couples structured visual representation with sequential, goal-conditioned visual access. Given an image–question pair, SSV-CoT performs a single forward pass through a vision encoder and MLLM to construct a structured visual representation: a question-aware saliency map is generated, binarized, and decomposed via connected component analysis, yielding a bank of candidate region embeddings together with a global scene embedding. Conditioned on the evolving reasoning state, a visual cognition policy scores candidate regions and selects one region (or the global embedding) at each step, injecting its embedding into the MLLM, so that reasoning unfolds as an interleaved process of visual querying and textual inference. To reflect human attentional control, SSV-CoT further incorporates an adaptive stopping mechanism that dynamically determines whether to continue visual access, balancing evidence sufficiency against redundant queries.

The method is trained end-to-end from image–question—text cot-answer without gaze traces, bounding boxes, or external tools: a heuristic supervised stage initializes the policy by imitating a question-conditioned saliency ordering derived from the pretrained MLLM, followed by a GRPO reinforcement learning stage that optimizes visual selection using answer-level rewards with a visual-budget regularizer. Overall, SSV-CoT replaces one-shot visual token injection with a structured, sequential, and adaptive visual reasoning process aligned with human cognitive principles.

Our main contributions are three-fold:

- We model visual access in multimodal reasoning as a *human-inspired, goal-conditioned* process over structured visual units, rather than passive visual token injection.
- We propose a training framework that learns visual selection order from task-level feedback using text-cot and answer supervision, without region-level annotations.
- We introduce an *adaptive, budget-aware* visual reasoning mechanism that decides when to continue or stop visual querying during chain-of-thought reasoning.

## 2 Related Work

### 2.1 Visual Reasoning

A broad class of methods augments MLLMs with external visual tools for cropping, detection, or region selection (Schick et al., 2023; Chen et al., 2023; Yang et al., 2023; Aissi et al., 2025), but these rely on complex, non-differentiable pipelines with manually designed actions decoupled from internal reasoning. Another line of work introduces region-level supervision via bounding boxes, captions, or grounding annotations (Li et al., 2022; Yang et al., 2022; Kamath et al., 2021; Liu et al., 2024a), which improves controllability but requires costly labeling and exposes all regions upfront, preventing sequential selection. Related segmentation-based or region-level abstractions, such as SAM-derived masks or region-level ViTs (Kirillov et al., 2023; Chen et al., 2022; Ravi et al., 2024), provide structured features but are typically used as static inputs rather than a decision space. Overall, these approaches treat vision as fixed or externally controlled and do not enable internal, learnable, sequential visual cognition in MLLMs.

### 2.2 Interleaved Visual Reasoning

Interleaved chain-of-thought methods reintroduce visual tokens during reasoning to alleviate image forgetting (Zhang et al., 2024b; Chen et al., 2025b; Shao et al., 2024; Zhang et al., 2024a; Li and Ma, 2025). While effective for multi-hop visual question answering and chart reasoning, these approaches repeatedly attend to the same static patch set and perform passive re-attention, without modeling where to look next or how visual access should evolve over time. Relatedly, sequential perception models such as glimpse networks and

active-vision agents (Mnih et al., 2014; Ba et al., 2015; Jaegle et al., 2022) demonstrate the benefits of goal-conditioned sequential inspection, but typically rely on small-scale vision architectures or reinforcement learning pipelines that do not scale to LLM-level multimodal reasoning. Overall, although recurrent and interleaved methods improve visual reasoning, they fail to support task-driven, autoregressive visual access that is tightly coupled with the reasoning process, leaving dynamic visual cognition largely unaddressed.

# 3 Methodology

## 3.1 Overview

Existing interleaved reasoning methods repeatedly inject dense patch-level visual tokens, often causing redundant signals and unstable reasoning. In contrast (Fig. 2), we first abstract the image into a compact set of question-conditioned visual regions and then perform *sequential visual access* over this region space. We introduce **SSV-CoT**, which decomposes multimodal reasoning into two stages: (1) **Structured Visual Cognition**, constructing salient question-aware region embeddings in a single forward pass; and (2) **Sequential Visual Access**, where a lightweight policy dynamically selects regions at each reasoning step.

## 3.2 Problem Formulation and Inference Process

Given an image $x$ and question $q_{\text{txt}}$, the model generates an answer by interleaving textual reasoning with selective visual access. Visual reasoning is formulated as a sequence of region-selection actions, where each step selects a region or terminates. At each reasoning step, the model queries a structured region embedding bank based on the current reasoning state until a `STOP` action produces the final answer.

Formally, while conventional multimodal CoT methods generate purely text-based reasoning steps

$$s^{(1)}, s^{(2)}, \ldots, s^{(K)}, \quad y = \text{LLM}(V, T), \tag{1}$$

where $V = v_{\tau}{}_{\tau=1}^{N}$ denotes static visual tokens extracted from the image and $T$ is the input question, our approach explicitly models reasoning as an interleaved decision process:

$$a_1, s^{(1)}, a_2, s^{(2)}, \ldots, a_T, s^{(T)}, \quad y = \text{LLM}(\mathcal{E}, T), \tag{2}$$

where each action $a_t$ selects a region embedding from the structured visual space $\mathcal{E}$ (or triggers `STOP`), grounding each reasoning step in task-relevant visual evidence.

## 3.3 Structured Visual Cognition

The first stage builds a compact, question-conditioned, and spatially structured image representation (Fig. 2), aiming to identify a small set of discriminative regions relevant to the question without region-level supervision. Region geometry and base features come from the pretrained vision encoder, while region importance is inferred through question-conditioned multimodal interactions.

**Question-conditioned saliency map.** Given an image $x$, we extract patch-level features $h_v = z_{ij}$ using a vision encoder, where $z_{ij}$ denotes patch embeddings on an $H \times W$ grid. We then run the pretrained MLLM on the image–question pair $(x, q_{\text{txt}})$ and obtain the final textual representation $q$ together with multimodal visual tokens $\tilde{z}_{ij}$. Since they lie in a shared embedding space, we compute a question-conditioned saliency score for each patch via

$$S_{ij} = \text{sim}(q, \tilde{z}_{ij}), \tag{3}$$

which reflects the relevance of each spatial location to the question. The saliency map is further normalized to improve robustness:

$$\tilde{S}ij = \frac{S_{ij} - \min(S)}{\max(S) - \min(S)}. \tag{4}$$

**Binary masking and connected component analysis.** We convert the normalized saliency map $\tilde{S}$ into a binary mask by thresholding:

$$M(i,j) = \mathbb{I}\big[\tilde{S}_{ij} \geq \tau\big], \tag{5}$$

where $\tau$ is an adaptive threshold (e.g., Otsu's method) and $\mathbb{I}[\cdot]$ denotes the indicator function.

We then apply connected-component analysis on $M$ under an 8-neighborhood connectivity criterion to obtain a set of candidate regions $\{\mathcal{R}_k\}$. Each region $\mathcal{R}_k$ consists of a connected set of salient patch indices. Small components below a minimum area threshold are discarded to remove spurious noise.

**Global complement embedding.** To ensure full image coverage and provide robustness against imperfect saliency estimation, all remaining patches not covered by the Top-$N$ regions are aggregated into a single global complement region:

$$e_{\text{global}} = \frac{1}{|\mathcal{I}_{\text{rest}}|} \sum_{(i,j) \in \mathcal{I}_{\text{rest}}} z_{ij}. \tag{6}$$

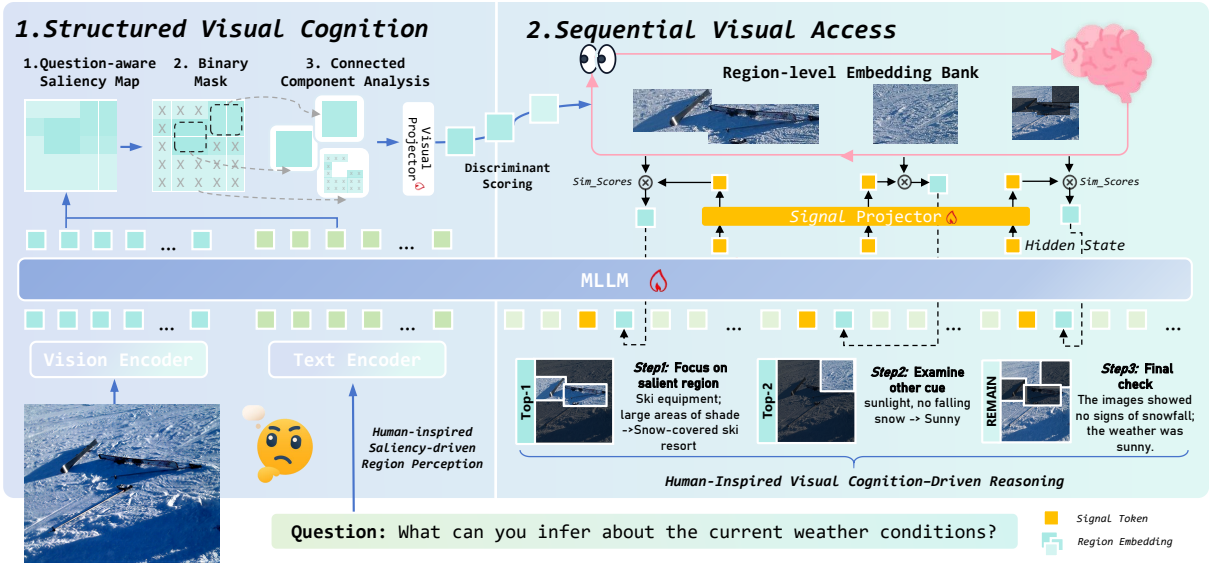



Figure 2: Overview of the **SSV-CoT** framework. The model first constructs question-aware structured visual regions, then performs sequential visual access during chain-of-thought reasoning to progressively integrate visual evidence and generate the final answer.

**Discriminative region construction.** Given the connected components $\{\mathcal{R}_k\}$ from the binary saliency mask, we treat each $\mathcal{R}_k$ as a candidate visual region and collect its patch-level visual tokens $\{v_j\}_{j=1}^{m}$. We adopt an adaptive token compression strategy with a fixed budget $n = 48$. If $m \leq n$, all tokens are preserved and projected individually via a visual projector. Otherwise, we select the top-$n$ tokens by saliency and apply $k$-means clustering to obtain representative region tokens. Formally, the region representation is a set of projected tokens:

$$E_k = g_{\text{proj}}\big(\{z_{ij} \mid (i,j) \in \mathcal{R}_k\}\big), \quad k = 1, \ldots, N, \tag{7}$$

where $g_{\text{proj}}(\cdot)$ denotes a permutation-invariant region projector that produces a fixed-size set of region tokens.

After obtaining region embeddings, we quantify the discriminativeness of each region by computing a region-level score as the average of the normalized saliency values within the region:

$$\rho_k = \frac{1}{|\mathcal{R}_k|} \sum_{(i,j)\in\mathcal{R}_k} \tilde{S}_{ij}. \tag{8}$$

This score measures the overall saliency consistency of the region with respect to the given question. Regions are then ranked according to $\rho_k$, and the Top-$N$ highest-scoring regions are selected as discriminative local regions for subsequent sequential visual reasoning. The final output of structured visual cognition is a compact region embedding bank:

$$\mathcal{E} = \{e_1, \ldots, e_N, e_{\text{global}}\}, \tag{9}$$

which provides a structured, question-conditioned abstraction of the image and serves as the candidate visual space for subsequent sequential visual access.

The structured region construction—including Otsu-based thresholding, binary masking, connected-component analysis, and Top-N selection—is treated as a deterministic, non-learnable preprocessing step performed at inference time, while all learnable parameters are confined to the projector, policy module, and the MLLM.

### 3.4 Sequential Visual Access

The second stage performs task-driven, step-aligned visual reasoning over the structured region space, corresponding to the right part of Figure 2.

**Step-aware triggering.** During chain-of-thought decoding, we treat the newline character \n as an implicit step boundary. Each time a newline is generated, the model completes a reasoning step and triggers a visual selection decision.

Let $h_t \in \mathbb{R}^{D_L}$ denote the final-layer hidden state of the newline token at step $t$. This hidden state summarizes the current reasoning context and serves as the query for visual selection.

**Similarity-based region scoring.** At each reasoning step $t$, we obtain the hidden state of the signal token $h_t \in \mathbb{R}^{D_L}$, which summarizes the current reasoning context. We first project the signal

token through a signal projector:

$$\tilde{h}_t = W_{\text{sig}}\, h_t \in \mathbb{R}^{D_L}. \tag{10}$$

Similarly, each region token embedding $e_k \in \mathbb{R}^{D_v}$ is projected into the language space via a visual projector:

$$\tilde{e}_k = W_{\text{vis}}\, e_k \in \mathbb{R}^{D_L}. \tag{11}$$

We then compute similarity scores between the projected signal token and all projected region embeddings:

$$s_{t,k} = \text{sim}(\tilde{h}_t, \tilde{e}_k), \quad k \in \{1, \ldots, K\}. \tag{12}$$

To enable adaptive termination of visual access, we additionally include a learnable `STOP` embedding, which is scored in the same manner as region tokens.

**Policy and action execution.** The policy head $g_\theta$ induces a distribution over actions:

$$\pi_\theta(a_t = k \mid s_t) = \text{softmax}(s_t)_k, \tag{13}$$

where $k \in \{1, \ldots, K, \text{STOP}\}$. If $a_t = k$, the corresponding region embedding is injected to condition subsequent reasoning; if $a_t = \text{STOP}$, the model produces the final answer without further visual input.

### 3.5 Training Procedure

We train SSV-CoT with a two-stage protocol using image–question–textual CoT–answer data, without any supervision on visual regions or access order. To mitigate noisy early saliency estimates, we adopt a probabilistic curriculum that gradually shifts from random visual access to structured, discriminative region reasoning. The pipeline consists of (1) heuristic supervised fine-tuning to initialize the visual access policy aligned with textual chain-of-thought reasoning, and (2) GRPO-based reinforcement learning to refine the visual cognition order using answer-level rewards.

**Stage I: Heuristic Supervised Fine-Tuning with Probabilistic Curriculum** The goal of the first stage is to teach the model how to interleave textual reasoning with visual region access, rather than to learn a perfect visual selection strategy. However, saliency maps estimated from a pretrained MLLM are often noisy at early training stages, which may lead to unstable optimization if directly used as supervision.

To stabilize training, we adopt a probabilistic curriculum learning strategy. Specifically, we define a curriculum coefficient $\lambda_e \in [0, 1]$ for epoch $e$, which linearly increases from 0 to 1 during a warm-up period of $E_{\text{warm}}$ epochs:

$$\lambda_e = \min\left(1, \frac{e}{E_{\text{warm}}}\right). \tag{14}$$

For each training sample $(x, q_{\text{txt}}, a_{\text{gold}})$, we construct a pseudo-expert region ordering as follows: With probability $1 - \lambda_e$, we sample regions using a random region strategy, which selects spatially contiguous image blocks independent of the question. With probability $\lambda_e$, we apply the proposed question-conditioned discriminative region selection method described in Section 3.3, and rank the resulting regions by their saliency scores.

The resulting region sequence $(r_1, \ldots, r_T)$ is treated as a heuristic viewing trajectory. During SFT, we perform behavior cloning on the visual access policy by minimizing the cross-entropy loss:

$$\mathcal{L}_{\text{SFT}} = -\sum_{t=1}^{T} \log \pi_\theta(a_t = r_t \mid s_t), \tag{15}$$

where $s_t$ denotes the policy state at reasoning step $t$.

This curriculum-guided SFT initializes the policy with stable region–text interleaving behavior while avoiding overfitting to unreliable early saliency signals.

**Stage II: GRPO-Based Reinforcement Learning with Mixed Exploration** While the heuristic policy learned in Stage I provides a stable initialization, it is not directly optimized for end-task performance. In the second stage, we refine the visual access policy using reinforcement learning with answer-level supervision.

Given a training sample $(x, q_{\text{txt}}, a_{\text{gold}})$, we fix the image and question and sample a group of $K$ trajectories using a mixed sampling policy:

$$\pi_{\text{mix}} = (1 - \lambda_e)\, \pi_{\text{rand}} + \lambda_e\, \pi_\theta, \tag{16}$$

where $\pi_{\text{rand}}$ denotes a random region selection policy and $\pi_\theta$ is the current learned policy. The same curriculum coefficient $\lambda_e$ is used to gradually shift from exploration to exploitation.

Each trajectory $i$ yields a region access sequence $g^{(i)}$, a generated answer $\hat{a}^{(i)}$, and a scalar reward $R^{(i)}$ computed solely based on answer quality.

Within each group, we compute the group-relative advantage:

$$A^{(i)} = R^{(i)} - \frac{1}{K}\sum_{j=1}^{K} R^{(j)}. \tag{17}$$

The policy is optimized using the GRPO objective:

$$\mathcal{L}_{\text{RL}} = -\sum_{i=1}^{K}\sum_{t=1}^{T_i} A^{(i)} \log \pi_\theta(a_t^{(i)} \mid s_t^{(i)}). \tag{18}$$

To prevent large deviations from the SFT-initialized policy, we additionally apply a KL regularization term:

$$\mathcal{L}_{\text{KL}} = \mathbb{E}_s\big[\text{KL}\big(\pi_\theta(\cdot \mid s) \,\|\, \pi_{\text{SFT}}(\cdot \mid s)\big)\big]. \tag{19}$$

The final optimization objective is:

$$\min_\theta \; \mathcal{L}_{\text{RL}} + \beta\, \mathcal{L}_{\text{KL}}. \tag{20}$$

The reward is a linear combination of four terms:

$$r = r_{\text{task}} + \lambda_1\, r_{\text{format}} - \lambda_2\, r_{\text{length}} - \lambda_3\, r_{\text{vision}}. \tag{21}$$

Here, $r_{\text{task}}$ is the task reward: for multiple-choice or numeric questions, it is 1 if the answer is exactly correct and 0 otherwise; for open-ended math reasoning, we use exact match with numerical-tolerance checking. The format reward $r_{\text{format}}$ encourages compliance with a predefined *reasoning–answer* output template, and invalid formatting or missing a final answer is penalized. The length penalty $r_{\text{length}}$ is proportional to the number of generated tokens to discourage overly verbose reasoning. The vision penalty $r_{\text{vision}}$ is proportional to the number of visual access steps (or queried regions) to discourage excessive visual reliance and promote discriminative, budget-aware visual usage with adaptive stopping.

# 4 Experiments

## 4.1 Experimental Datasets

We evaluate our method on a diverse set of visual reasoning benchmarks. For commonsense visual reasoning, we use M3CoT (Zhang et al., 2024b), ScienceQA (Lu et al., 2022), and LLaVA-Bench In-the-Wild (LLaVA-W) (Liu et al., 2023). M3CoT focuses on multi-domain, multi-step multimodal reasoning, ScienceQA serves as a standard benchmark for general VLM reasoning, and LLaVA-W evaluates fine-grained visual understanding via long-form answers with GPT-4V references. For mathematical visual reasoning, we adopt MathVista (Lu et al., 2024a) and MathVision (Wang et al., 2024a), which require both fine-grained visual perception and multi-step mathematical reasoning.

For training data, we use the M3CoT training set and the LLaVA-CoT-100k (Xu et al., 2025) training set for commonsense visual reasoning. For mathematical visual reasoning, we adopt MathV360K (Shi et al., 2024), from which we carefully select over 50K diverse samples and generate textual chain-of-thought annotations using GPT-4V. All the above datasets contain textual chain-of-thought annotations and do not include any visual grounding or visual token-level annotations.

## 4.2 Compared Methods

For **commonsense visual reasoning**, we compare our method with several representative reasoning strategies, all instantiated on the same Qwen2-VL-7B backbone for fairness. **Multimodal CoT** (Zhang et al., 2024b) generates text-only intermediate reasoning steps before producing the final answer. **CCoT** (Mitra et al., 2024) constructs a scene graph to capture compositional object relations and uses it to guide answer generation. **DDCoT** (Zheng et al., 2023) decomposes the question into sub-questions and answers them using a VLM. **SCAFFOLD** (Lei et al., 2024) overlays coordinate grids onto images to expose fine-grained spatial information and guides reasoning with these coordinates. **ICoT** (Gao et al., 2025) repeatedly re-injects visual tokens during CoT while treating visual patches uniformly.

For **mathematical visual reasoning**, we compare our method against a set of strong recent vision-language models, including MiniCPM-V-2.6 (Yao et al., 2024b), VITA-1.5 (Fu et al., 2025), LLaVA-CoT (Xu et al., 2025), Qwen2-VL-7B (Wang et al., 2024b), InternVL2.5 (Chen et al., 2025c), POINTS (Liu et al., 2024b), Ovis (Lu et al., 2024b), and TVC-Qwen2-VL-7B (Sun et al., 2025). TVC-Qwen2-VL-7B mitigates visual forgetting in multimodal long-chain reasoning by periodically reintroducing compressed visual inputs during reasoning.

## 4.3 Model Implementation

We use Qwen2-VL-7B as the base model in our experiments. Both projectors are implemented as

| Methods | M3CoT ACC.↑ | ScienceQA ACC.↑ | LLaVA-W ROUGE-L↑ |
|---|---|---|---|
| Qwen2-VL-7B | 43.6 | 56.3 | 32.7 |
| MultimodalCoT | 40.1 | 51.3 | 30.7 |
| CCoT | 43.3 | 56.4 | 29.4 |
| DDCoT | 42.6 | 55.2 | 31.2 |
| SCAFFOLD | 41.7 | 53.7 | 31.8 |
| ICoT | 44.1 | 56.8 | 34.2 |
| **SSV-CoT (Ours)** | **44.9** | **57.3** | **35.7** |

Table 1: **Commonsense visual reasoning results.** Comparison of SSV-CoT applied to Qwen2-VL-7B with CoT-based methods (including No-CoT, MultimodalCoT, CCoT, DDCoT, SCAFFOLD, and ICoT) on M3CoT (Accuracy), ScienceQA (Accuracy), and LLaVA-W (ROUGE-L). The best result for each benchmark is highlighted in bold.

| Model | MathVista Acc.↑ | MathVision Acc.↑ | Average Acc.↑ |
|---|---|---|---|
| MiniCPM-V-2.6 | 60.8 | 18.4 | 39.6 |
| VITA-1.5 | 66.2 | 19.5 | 42.9 |
| LLaVA-CoT | 52.5 | 19.9 | 36.2 |
| Qwen2-VL-7B | 60.9 | 16.3 | 38.6 |
| InternVL2.5 | 64.5 | 17.0 | 40.8 |
| POINTS 1.5 | 66.4 | 22.0 | 44.2 |
| Ovis1.6-Gemma2 | 70.2 | 20.6 | 45.4 |
| TVC-Qwen2-VL-7B | 68.1 | 22.7 | 45.4 |
| **SSV-CoT (Ours)** | **72.2** | **23.5** | **47.9** |

Table 2: **Mathematical visual reasoning results.** Comparison of SSV-CoT applied to Qwen2-VL-7B with vision-language models on MathVista (Accuracy) and MathVision (Accuracy). The best results are highlighted in bold.

single-layer linear modules, and we uniformly set the similarity filtering threshold to $\theta = 0.7$. See the appendix for detailed settings.

### 4.4 Overall Performance

Overall, the experimental results indicate that SSV-CoT leads to consistent improvements in long-context multimodal reasoning across both commonsense and mathematical tasks.

For commonsense visual reasoning (Table 1), SSV-CoT shows moderate gains over the Qwen2-VL-7B backbone, with improvements of +1.3 Accuracy on M3CoT, +1.0 Accuracy on ScienceQA, and +3.0 ROUGE-L on LLaVA-W. Compared with ICoT, which interleaves visual tokens while treating image patches uniformly, SSV-CoT better preserves task-relevant visual information through goal-conditioned and sequential visual selection, resulting in more stable performance across benchmarks.

For mathematical visual reasoning (Table 2), SSV-CoT also improves upon the Qwen2-VL-7B baseline, increasing the average score from 38.6 to 47.9. In comparison to TVC, which alleviates visual forgetting by periodically reintroducing compressed visual inputs, SSV-CoT further aligns visual selection with reasoning objectives, supporting more effective use of visual information in multi-step mathematical reasoning.

Overall, these results suggest that SSV-CoT helps mitigate visual forgetting and supports task-aware visual grounding, contributing to improved performance across diverse visual reasoning scenarios.

| Methods | M3CoT ACC.↑ | ScienceQA ACC.↑ | LLaVA-W ROUGE-L↑ |
|---|---|---|---|
| SSV-CoT | 44.9 | 57.3 | 35.7 |
| w/o SR | 43.5 (-1.4) | 56.1 (-1.2) | 33.9 (-1.8) |
| w/o SS | 43.8 (-1.1) | 56.4 (-0.9) | 34.2 (-1.5) |
| w/o AS | 44.3 (-0.6) | 56.9 (-0.4) | 35.0 (-0.7) |

Table 3: Ablation study of SSV-CoT on commonsense visual reasoning benchmarks. SR denotes Structured Regions, SS denotes Sequential Selection, and AS denotes Adaptive Stopping. Numbers in parentheses denote performance degradation compared with the full model.

## 5 Discussion

### 5.1 Ablation Experiment

We ablate SSV-CoT to assess the contribution of its core components across three commonsense visual reasoning benchmarks (Table 3), with the following settings: (1) *w/o Structured Regions*: the model does not construct question-conditioned structured regions and directly injects raw visual tokens, resembling patch-level interleaved reasoning; (2) *w/o Sequential Selection*: all region embeddings are injected at once, removing the sequential visual access order; (3) *w/o Adaptive Stopping*: the model is forced to attend to a fixed number of regions without dynamically deciding when to stop visual querying.

The results show that all components contribute to SSV-CoT. Removing structured regions causes the largest performance drop, while disabling sequential selection also consistently degrades performance. In contrast, removing adaptive stopping leads to smaller but stable declines. Performance drops are more pronounced on M3CoT and LLaVA-W than on ScienceQA, reflecting their stronger reliance on fine-grained visual information.

### 5.2 In-depth Analysis

**Structure and order are mutually reinforcing.** Table 4 jointly ablates visual structure and cog-

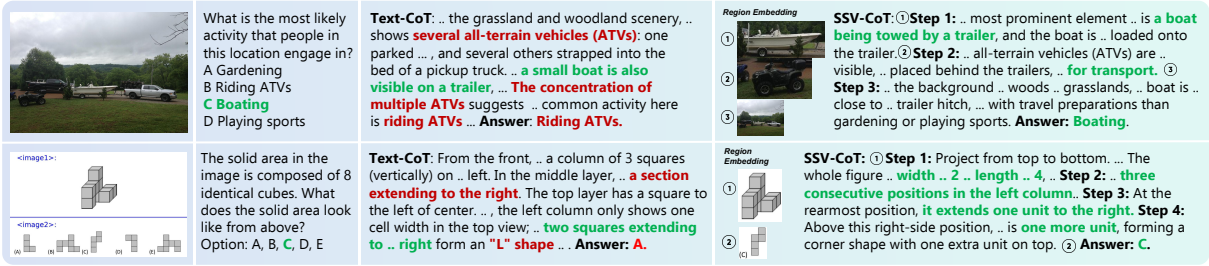



Figure 3: Comparison of Text-CoT and SSV-CoT on Qwen2-VL-7B for visual reasoning. Errors are shown in red. Only selected regions are displayed, and numbers indicate the token injection order.

| | Order (Acc@1 %) | |
|---|---|---|
| Visual repr. | Random | Cognition |
| Patch subset | 63.1 | 65.4 |
| Saliency regions | 68.0 | 72.2 |

Table 4: 2×2 ablation of visual structure and cognition order on MathVista. Each cell reports Acc@1 (%). All variants share the same visual token budget and maximum steps $K$.

| Strategy | Acc@1 (%) | Avg #regions |
|---|---|---|
| Fixed-$K$=2 (Seq) | 68.5 | 2.0 |
| Fixed-$K$=3 (Seq) | 71.0 | 3.0 |
| Fixed-$K$=4 (Seq) | 71.3 | 4.0 |
| Adaptive-$K$ (Ours) | 72.2 | 2.7 |

Table 5: Visual budget analysis on MathVista. Fixed-$K$ variants always query exactly $K$ regions. Our adaptive-$K$ policy learns when to stop via a <STOP> action, achieving comparable or better accuracy with fewer region queries on average.

nition order on MathVista. Using unstructured patch subsets with a random schedule (*Patch subset, Random*) gives the weakest performance. Introducing a learned access order on top of the same patches (*Patch subset, Cognition*) yields a modest gain (63.1→65.4), suggesting that sequential, goal-conditioned access helps even with weak visual units. Replacing patches with saliency-based regions under a random order (*Saliency regions, Random*) brings a larger improvement (63.1→68.0), showing that structured, semantically coherent regions are crucial on their own. Our full model (*Saliency regions, Cognition*) combines both ingredients and achieves the best accuracy (72.2), with gains that exceed either modification in isolation. This pattern indicates that region structure and cognition order are complementary: the policy needs good regions to reason over, and good regions are used most effectively when queried in a learned sequence.

**Adaptive visual budget.** Table 5 analyzes how the visual budget interacts with the sequential policy. Increasing the fixed number of region queries from $K$=2 to $K$=3 substantially improves performance (68.5→71.0), but further raising the budget to $K$=4 yields only marginal gains (71.3) while forcing the model to inspect more regions. In contrast, our adaptive-$K$ variant, equipped with a learned <STOP> action, reaches the highest accuracy (72.2) while using 2.7 regions on average. This shows that the model not only learns *where* and *in which order* to look, but also *how much* visual evidence is needed for a given question, yielding more accurate reasoning with a tighter and automatically controlled visual budget.

### 5.3 Case Study

This Fig. 3 compares the performance of Text-CoT and Structured Visual Reasoning (SSV-CoT) in two typical cases: common-sense visual reasoning (top) and mathematical visual reasoning (bottom). In the common-sense case, Text-CoT focuses on salient but incomplete clues (e.g., multiple ATVs) while ignoring finer contextual details, leading to incorrect conclusions. In contrast, SSV-CoT sequentially focuses on task-relevant areas (boat, trailer, background), progressively integrating evidence to arrive at the correct answer. In the mathematical case, Text-CoT relies on a holistic description of the structure, misinterpreting the spatial layout. However, SSV-CoT explicitly selects and infers key areas (projection, column continuity, extension, and height), enabling accurate spatial reasoning.

## 6 Conclusion

We introduce SSV-CoT, which frames visual cognition in multimodal reasoning as a structured, sequential process instead of static token injection. Trained with chain-of-thought supervision and answer-level rewards, SSV-CoT learns visual selection and stopping without region annotations or external tools. Empirical results across common-sense and math benchmarks demonstrate consistent improvements and highlight the importance of structured regions and adaptive visual access.

## Limitations

SSV-CoT introduces additional computation due to sequential visual access compared with one-shot visual token injection. Although the vision encoder is executed only once, the model still needs extra region scoring and policy decisions during decoding. Moreover, its effectiveness depends on the quality of the initially constructed question-conditioned region bank. If the saliency map misses small but decisive evidence, or if the connected components fail to cover the correct fine-grained region, the sequential policy cannot recover a new region outside the precomputed candidate bank. This limitation is especially relevant for dense charts, diagrams, and spatial reasoning tasks. Extending SSV-CoT with multi-scale region proposals, iterative region refinement, or dynamic region generation during reasoning remains future work.

## Ethical Considerations

SSV-CoT is trained on existing vision-language datasets and does not require additional annotations or external tools. Like other vision-language models, it may inherit biases from the training data. Care should be taken when applying the method in real-world or high-stakes scenarios.

## References


Mohamed Salim Aissi, Clemence Grislain, Mohamed Chetouani, et al. 2025. Viper: Visual perception and explainable reasoning for sequential decision-making. *CoRR*, abs/2503.15108.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, et al. 2022. Flamingo: a visual language model for few-shot learning. In *Advances in Neural Information Processing Systems*.

Jimmy Ba, Volodymyr Mnih, and Koray Kavukcuoglu. 2015. Multiple object recognition with visual attention. In *International Conference on Learning Representations*. Poster.

Chun-Fu Chen, Rameswar Panda, and Quanfu Fan. 2022. Regionvit: Regional-to-local attention for vision transformers. In *International Conference on Learning Representations*.

Keqin Chen, Zhao Zhang, Weili Zeng, et al. 2023. Shikra: Unleashing multimodal llm's referential dialogue magic. *CoRR*, abs/2306.15195.

Xinyan Chen, Renrui Zhang, Dongzhi Jiang, Aojun Zhou, Shilin Yan, Weifeng Lin, and Hongsheng Li. 2025a. MINT-CoT: Enabling interleaved visual tokens in mathematical chain-of-thought reasoning.

Xinyan Chen, Renrui Zhang, Dongzhi Jiang, et al. 2025b. Mint-cot: Enabling interleaved visual tokens in mathematical chain-of-thought reasoning. *CoRR*, abs/2506.05331.

Zhe Chen, Weiyun Wang, Yue Cao, et al. 2025c. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *CoRR*, abs/2412.05271.

Wenliang Dai, Junnan Li, Dongxu Li, et al. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. In *Advances in Neural Information Processing Systems*.

Ming Ding, Wendi Zheng, Wenyi Hong, and Jie Tang. 2022. Cogview2: Faster and better text-to-image generation via hierarchical transformers. *CoRR*, abs/2204.14217.

Patrick Esser, Robin Rombach, and Björn Ommer. 2021. Taming transformers for high-resolution image synthesis. *CoRR*, abs/2012.09841.

Chaoyou Fu, Haojia Lin, Xiong Wang, et al. 2025. Vita-1.5: Towards gpt-4o level real-time vision and speech interaction. *CoRR*, abs/2501.01957.

Jun Gao, Yongqi Li, Ziqiang Cao, and Wenjie Li. 2025. Interleaved-modal chain-of-thought. *CoRR*, abs/2411.19488.

L. Itti, C. Koch, and E. Niebur. 1998. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259.

Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, et al. 2022. Perceiver io: A general architecture for structured inputs & outputs. In *International Conference on Learning Representations*.

Aishwarya Kamath, Mannat Singh, Yann LeCun, et al. 2021. Mdetr – modulated detection for end-to-end multi-modal understanding. *CoRR*, abs/2104.12763.

Aisha Urooj Khan, Hilde Kuehne, Chuang Gan, et al. 2022. Weakly supervised grounding for vqa in vision-language transformers. *CoRR*, abs/2207.02334.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, et al. 2023. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3992–4003.

Xuanyu Lei, Zonghan Yang, Xinrui Chen, et al. 2024. Scaffolding coordinates to promote vision-language coordination in large multi-modal models. *CoRR*, abs/2402.12058.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. 2023. 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*, pages 19730–19742.

Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, et al. 2022. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10955–10965.

Xiping Li and Jianghong Ma. 2025. Aimcot: Active information-driven multimodal chain-of-thought for vision-language reasoning. *CoRR*, abs/2509.25699.

Yuanze Lin, Yujia Xie, Dongdong Chen, et al. 2022. Revive: Regional visual representation matters in knowledge-based visual question answering. *CoRR*, abs/2206.01201.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *CoRR*, abs/2304.08485.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, et al. 2024a. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, pages 38–55.

Yuan Liu, Le Tian, Xiao Zhou, et al. 2024b. Points1.5: Building a vision-language model towards real world applications. *CoRR*, abs/2412.08443.

Pan Lu, Hritik Bansal, Tony Xia, et al. 2024a. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations*.

Pan Lu, Swaroop Mishra, Tony Xia, et al. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *Advances in Neural Information Processing Systems*.

Shiyin Lu, Yang Li, Qing-Guo Chen, et al. 2024b. Ovis: Structural embedding alignment for multimodal large language model. *CoRR*, abs/2405.20797.

Chancharik Mitra, Brandon Huang, Trevor Darrell, and Roei Herzig. 2024. Compositional chain-of-thought prompting for large multimodal models. *CoRR*, abs/2311.17076.

Volodymyr Mnih, Nicolas Heess, Alex Graves, et al. 2014. Recurrent models of visual attention. In *Advances in Neural Information Processing Systems*, pages 2204–2212.

Alec Radford, Jong Wook Kim, Chris Hallacy, et al. 2021. Learning transferable visual models from natural language supervision. *CoRR*, abs/2103.00020.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, et al. 2024. Sam 2: Segment anything in images and videos. *CoRR*, abs/2408.00714.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, et al. 2023. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems*.

Hao Shao, Shengju Qian, Han Xiao, et al. 2024. Visual cot: Advancing multi-modal language models with a comprehensive dataset and benchmark for chain-of-thought reasoning. *CoRR*, abs/2403.16999.

Wenhao Shi, Zhiqiang Hu, Yi Bin, et al. 2024. Math-llava: Bootstrapping mathematical reasoning for multimodal large language models. *CoRR*, abs/2406.17294.

Hai-Long Sun, Zhun Sun, Houwen Peng, and Han-Jia Ye. 2025. Mitigating visual forgetting via take-along visual conditioning for multi-modal long cot reasoning. *CoRR*, abs/2503.13360.

Ke Wang, Junting Pan, Weikang Shi, et al. 2024a. Measuring multimodal mathematical reasoning with math-vision dataset. In *Advances in Neural Information Processing Systems*.

Peng Wang, Shuai Bai, Sinan Tan, et al. 2024b. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *CoRR*, abs/2409.12191.

Jason Wei, Xuezhi Wang, Dale Schuurmans, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Chenfei Wu, Shengming Yin, Weizhen Qi, et al. 2023. Visual chatgpt: Talking, drawing and editing with visual foundation models. *CoRR*, abs/2303.04671.

Guowei Xu, Peng Jin, Ziang Wu, et al. 2025. Llava-cot: Let vision language models reason step-by-step. *CoRR*, abs/2411.10440.

Zhengyuan Yang, Zhe Gan, Jianfeng Wang, et al. 2022. Unitab: Unifying text and box outputs for grounded vision-language modeling. In *European Conference on Computer Vision*, pages 521–539.

Zhengyuan Yang, Linjie Li, Jianfeng Wang, et al. 2023. Mm-react: Prompting chatgpt for multimodal reasoning and action. *CoRR*, abs/2303.11381.

Huanjin Yao, Jiaxing Huang, Wenhao Wu, Jingyi Zhang, Yibo Wang, Shunyu Liu, Yingjie Wang, Yuxin Song, Haocheng Feng, Li Shen, and Dacheng Tao. 2024a. Mulberry: Empowering MLLM with o1-like reasoning and reflection via collective monte carlo tree search.

Yuan Yao, Tianyu Yu, Ao Zhang, et al. 2024b. Minicpm-v: A gpt-4v level mllm on your phone. *CoRR*, abs/2408.01800.

Alfred L. Yarbus. 1967. *Eye Movements and Vision*. Plenum Press, New York.

Daoan Zhang, Junming Yang, Hanjia Lyu, et al. 2024a. Cocot: Contrastive chain-of-thought prompting for large multimodal models with multiple image inputs. *CoRR*, abs/2401.02582.

Jingyi Zhang, Jiaxing Huang, Huanjin Yao, Shunyu Liu, Xikun Zhang, Shijian Lu, and Dacheng Tao. 2025. R1-VL: Learning to reason with multimodal large language models via step-wise group relative policy optimization.

Zhuosheng Zhang, Aston Zhang, Mu Li, et al. 2024b. Multimodal chain-of-thought reasoning in language models. *Transactions on Machine Learning Research*.

Ge Zheng, Bin Yang, Jiajin Tang, et al. 2023. Dd-cot: Duty-distinct chain-of-thought prompting for multimodal reasoning in language models. *CoRR*, abs/2310.16436.

# A Algorithm

**Algorithm 1** Question-aware Discriminative Region Selection with Adaptive Token Compression

**Require:** Image $x$, text query $q$, multimodal model $\mathcal{M}$, number of regions $N$, token budget $n$
**Ensure:** Region set $\mathcal{R} = \{R_1, \ldots, R_N\}$ and region tokens $\{E_k\}_{k=1}^{N}$

$V \leftarrow \text{VisionEncoder}_{\mathcal{M}}(x),\ t \leftarrow \text{TextEncoder}_{\mathcal{M}}(q)$

$A(i,j) \leftarrow \text{sim}(v_{i,j}, t),\ \tilde{A} \leftarrow \text{Normalize}(A)$

$M \leftarrow \tilde{A} \geq \text{OtsuThreshold}(\tilde{A})$
$\{R_k\} \leftarrow \text{ConnectedComponents}(M)$

Compute $\rho_k = \frac{1}{|R_k|} \sum_{(i,j)\in R_k} \tilde{A}(i,j)$
Select top $(N-1)$ regions by $\rho_k$
$R_N \leftarrow \Omega \setminus \bigcup_{k=1}^{N-1} R_k$

**for** *each region* $R_k$ **do**
  Collect region tokens $\{v_j\}_{j=1}^{m}$
  **if** $m \leq n$ **then**
    $E_k \leftarrow \{\text{Proj}(v_j)\}_{j=1}^{m}$
  **else**
    Select top-$n$ tokens by saliency
    Cluster selected tokens with $k$-means
    $E_k \leftarrow \{\text{Proj}(\bar{v}_i)\}_{i=1}^{n}$

**return** $\{\mathcal{R}, E_k\}_{k=1}^{N}$

Algorithm 1 describes the question-aware region selection process. We compute a query-conditioned saliency map over visual patches, apply adaptive thresholding and connected-component analysis to obtain candidate regions, and select the top $N-1$ regions by average saliency, with the remaining area treated as background.

# B Key Implementation Details

**Region Bank Construction and Step-wise Injection.** Given an input image $x$ and question $q$, SSV-CoT first constructs a fixed-size candidate region bank $\mathcal{E} = \{e_1, \ldots, e_N\}$ before chain-of-thought decoding. This stage is performed once per image–question pair: the vision encoder is executed once, and the resulting region embeddings are cached for subsequent reasoning steps. Importantly, the fixed-size bank does not mean that all region embeddings are injected at the beginning of inference.

During chain-of-thought decoding, whenever a valid reasoning-step boundary is detected, the step-wise visual policy selects one entry from the cached region bank according to the current reasoning state. The selected region embedding is projected into the LLM hidden space and inserted as additional visual

evidence for the next reasoning segment:

$$\hat{e}_{a_t} = \mathrm{Proj}(e_{a_t}) \in \mathbb{R}^{d_{\mathrm{LM}}}.$$

If the selected action is STOP, no further visual region is injected and the model proceeds to generate the final answer. Therefore, what remains fixed during inference is the candidate region bank and its number of slots, while the actually accessed visual evidence is dynamically selected at each reasoning step. This design neither replaces the original visual tokens nor requires repeatedly running the vision encoder during decoding.

**Region Selection and Hyperparameters.** We set the default number of regions to $N = 5$, consisting of $N-1$ discriminative regions and one background region. Regions are generated via Otsu-based adaptive thresholding over a similarity heatmap, followed by connected-component analysis. Components with area smaller than $\alpha = 0.01 \times HW$ (where $H, W$ denote the feature map resolution) are discarded. The remaining regions are ranked by their average activation score $\rho_k$, and the top $N-1$ regions are selected.

**GRPO Reward Design.** We train the model using GRPO, where the reward is defined as a weighted sum:

$$r = r_{\text{task}} + \lambda_1 r_{\text{format}} - \lambda_2 r_{\text{length}} - \lambda_3 r_{\text{vision}}.$$

Here, $r_{\text{task}}$ measures answer correctness, $r_{\text{format}}$ enforces valid output structure, $r_{\text{length}}$ penalizes overly long generations, and $r_{\text{vision}}$ penalizes excessive visual access. We set the weights to

$$\lambda_1 = 0.2, \quad \lambda_2 = 0.01, \quad \lambda_3 = 0.05.$$

**Training Configuration.** We use Qwen2-VL-7B as the base multimodal large language model (MLLM) in our experiments. Both projectors are implemented as single-layer linear modules. The training procedure consists of two stages: (1) Supervised Fine-Tuning (SFT), where we train for 3 epochs with a learning rate of $1 \times 10^{-6}$ and a batch size of 64; and (2) Reinforcement Learning (RL), where we train for 700 steps with group size $G = 4$, a weighting factor $\lambda = 0.02$, a learning rate of $1 \times 10^{-6}$, and a batch size of 16. During training, the vision encoder is kept frozen, while all remaining model parameters (including the projector layers) are unfrozen and optimized.

# C Additional Implementation Clarifications and Analyses

## C.1 Similarity Function and Step Boundary

**Cosine similarity for saliency estimation.** In the main text, the function $\mathrm{sim}(\cdot,\cdot)$ is implemented as cosine similarity. Specifically, given the MLLM-contextualized question representation $q$ and the MLLM-contextualized visual token $\tilde{z}_{ij}$ at spatial position $(i, j)$, the question-conditioned saliency score is computed as

$$S_{ij} = \cos(q, \tilde{z}_{ij}) = \frac{q^\top \tilde{z}_{ij}}{\|q\|_2 \|\tilde{z}_{ij}\|_2}. \tag{22}$$

The resulting similarity scores are then normalized and used for Otsu thresholding, connected-component analysis, and Top-$N$ region selection.

**Reasoning-step boundary.** We use the newline token \n as an implicit reasoning-step boundary because the chain-of-thought training data follows a step-wise reasoning template in which each newline typically corresponds to a logical reasoning transition. To reduce meaningless triggers caused by formatting, visual access is activated only when the generated newline matches the expected step-transition pattern in the prescribed output format. In addition, the format reward encourages valid reasoning–answer structure, while the visual-access penalty discourages unnecessary region queries. Nevertheless, this trigger is still a heuristic design. A more robust alternative is to introduce an explicit special step token for visual access control, which we leave for future work.

## C.2 Controlled Same-Training Ablation

Public mathematical reasoning baselines may differ in backbone, data scale, supervision signals, and optimization pipelines. Therefore, public benchmark comparisons mainly demonstrate overall competitiveness rather than strict mechanism-level attribution. To better isolate the contribution of sequential visual access, we conduct a controlled same-training ablation under the same LoRA-GRPO setting. All variants use the same Qwen2-VL backbone, the same 4K MathV360K SFT data, the same GRPO-LoRA configuration, batch size 16, group size 4, LoRA rank 4, and 300 training steps. The only difference is whether SSV visual access is introduced and whether the region selection policy is conditioned on the current reasoning state.

As shown in Table 6, SSV-GRPO-LoRA improves over vanilla Qwen2-VL GRPO-LoRA by

| Method | MathVista Acc. | Delta |
|---|---|---|
| Vanilla Qwen2-VL GRPO-LoRA | 59.5 | – |
| SSV-GRPO-LoRA w/o step state | 62.0 | +2.5 |
| SSV-GRPO-LoRA | 63.5 | +4.0 |

Table 6: Controlled same-training ablation on MathVista under the same 4K LoRA-GRPO setting. The results show that SSV visual access improves over vanilla GRPO, and step-aware sequential selection further improves over non-step-aware visual access.

4.0 points. Moreover, removing the step-aware reasoning state reduces the gain from +4.0 to +2.5, indicating that the improvement is not merely due to applying GRPO, but is also related to reasoning-conditioned sequential visual access.

### C.3 Additional Public Mathematical Reasoning Baselines

We further compare SSV-CoT with recent strong mathematical visual reasoning baselines. These methods differ in training data, supervision type, and optimization recipe, so the comparison should be interpreted as an overall competitiveness comparison rather than a fully controlled attribution experiment.

### C.4 Training and Inference Efficiency

SSV-CoT introduces additional region scoring and step-wise visual access, but it does not repeatedly run the vision encoder during decoding. The image is encoded once before reasoning, and the candidate region bank is cached. During chain-of-thought generation, the additional computation mainly comes from lightweight similarity scoring over a small region bank and policy selection.

**Training overhead.** We compare Vanilla GRPO-LoRA and SSV-GRPO-LoRA under the same configuration: batch size 16, group size 4, LoRA rank 4, and 300 training updates on NVIDIA H100 GPUs.

As shown in Table 8, SSV-GRPO-LoRA introduces a moderate training overhead. The mean GPU memory increases from 44.832 GiB to 46.165 GiB, and the train-loop wall time increases from 4.59 hours to 5.18 hours. This overhead mainly comes from region scoring and the step-wise visual access policy, rather than repeated execution of the vision encoder.

**Inference latency.** We also evaluate inference efficiency on 200 MathVista samples using vLLM on a single NVIDIA H100 GPU.

Table 9(a) shows that SSV-CoT increases end-to-end latency from 757.4 ms to 899.0 ms per sample, while maintaining nearly the same token throughput. The additional latency is mainly due to region bank construction, region selection, and slightly longer reasoning outputs.

**Single-pass visual encoding.** Let $C_{\text{vis}}$ denote the cost of one vision-encoder forward pass, $C_{\text{dec}}$ denote the language decoding cost, $K$ denote the number of candidate regions, and $T$ denote the number of visual-access decisions. SSV-CoT performs visual encoding only once and caches the resulting region bank. Its inference cost can be summarized as

$$C_{\text{SSV}} = C_{\text{vis}} + C_{\text{dec}} + O(TKd),$$

where $O(TKd)$ corresponds to similarity scoring between the reasoning-state query and the cached region embeddings. Since $K$ is small in our implementation, this additional term is lightweight compared with repeatedly processing the image. In contrast, methods that repeatedly crop, re-encode, or re-inject dense visual inputs may require multiple visual encoding or dense visual-token processing operations during reasoning. Therefore, the main efficiency advantage of SSV-CoT comes from decoupling one-time region-bank construction from step-wise visual access.

### C.5 Adaptive Stopping Behavior

To verify whether the `STOP` action learns adaptive stopping rather than converging to a fixed average number of visual queries, we group 1000 evaluation samples by question complexity and report the number of queried regions.

As shown in Table 9(b), the average number of queried regions increases from 2.16 for simple questions to 3.05 for complex questions. Meanwhile, the early-stop ratio decreases from 66.0% to 28.0%, while the ratio of querying at least three regions increases from 34.0% to 72.0%. This trend suggests that SSV-CoT dynamically adjusts its visual-access budget according to question complexity.

| Method | Training / Supervision | MathVista Acc. |
|---|---|---|
| TVC-Qwen2-VL-7B | Large-scale SFT, no RL | 68.1 |
| R1-VL-7B (Zhang et al., 2025) | Mulberry-260K (Yao et al., 2024a) + StepGRPO | 69.63 |
| MINT-CoT-7B (Chen et al., 2025a) | 54K step-level visual-token annotations + SFT + RL | 73.70 |
| SSV-CoT | 50K+ SFT + GRPO, no region-level annotations | 72.2 |

Table 7: Additional public mathematical visual reasoning comparison on MathVista. Unlike methods relying on step-level visual-token annotations, SSV-CoT does not use region-level or visual-token-level annotations.

| Method | Updates | Peak GPU (GiB) | Mean GPU (GiB) | Mean Update Wall (s) | Train-loop Wall (h) |
|---|---|---|---|---|---|
| Vanilla GRPO-LoRA | 300 | 52.276 | 44.832 | 55.086 | 4.59 |
| SSV-GRPO-LoRA | 300 | 56.481 | 46.165 | 62.086 | 5.18 |

Table 8: Training efficiency comparison under the same GRPO-LoRA configuration. SSV-GRPO-LoRA increases mean GPU memory by about 3.0% and train-loop wall time by about 1.13×.

### C.6 Effect of the Global Complement Region

The global complement region is introduced as an auxiliary robustness mechanism. It aggregates patches not covered by the selected discriminative regions, providing coarse contextual information from the remaining image area. It is not intended to fully recover fine-grained evidence missed by the saliency map.

As shown in Table 9(c), removing the global complement region decreases accuracy from 68.0% to 66.5%. This indicates that the complement region provides useful auxiliary context. However, because it is constructed by mean-pooling all unselected patches, it may over-smooth background information, especially when the remaining image contains multiple secondary objects or fine-grained cues. Therefore, the complement region should be viewed as a coarse robustness component rather than a primary source of fine-grained visual reasoning.

### C.7 Motivation and Limitation of K-means Token Compression

When a candidate region contains no more than $n = 48$ visual tokens, SSV-CoT preserves all tokens in that region. When the number of tokens exceeds the budget, we apply K-means clustering in the embedding space and use the resulting centroids as representative region tokens. The purpose of this design is to reduce redundant tokens and preserve semantic diversity under a fixed token budget.

Compared with top-saliency selection, K-means avoids over-concentrating on only the strongest local responses and provides broader coverage of the region. Compared with random sampling, it is more stable and less likely to miss important visual patterns. However, K-means does not explicitly preserve the original 2D spatial order of visual patches. Therefore, it is mainly designed for semantic coverage rather than precise geometry preservation. This limitation may affect tasks requiring highly fine-grained spatial or chart reasoning, and future work may explore spatially constrained clustering or multi-scale region refinement.

### C.8 Failure Case Analysis

Although SSV-CoT mitigates visual forgetting by selecting from a structured region bank, the one-shot construction of this bank can still become a bottleneck. If the initial saliency-based region proposals miss critical evidence, the sequential policy can only select from the precomputed candidates and cannot recover a new high-resolution region during later reasoning.

For example, in a MathVista document-chart case, the question asks which subject has the highest pulse rate during the baseline period. The correct answer is subject 1, but SSV-CoT predicts 84. The selected regions include a coarse page-level region, a narrow strip covering only part of the pulse-rate panel, and an off-target lower-panel crop. Although the page-level region provides global context, it is too diffuse for fine-grained chart reading. Meanwhile, the localized regions do not jointly cover the full baseline segment and the corresponding subject-index context. As a result, the model confuses the subject index with a plotted pulse-rate value.

This failure suggests that region-bank construction is a critical component of SSV-CoT. When the initially constructed region bank lacks the neces-

| (a) Inference efficiency on 200 MathVista samples | | | | | |
|---|---|---|---|---|---|
| **Method** | **Acc.** | **Latency / Sample (ms)** | **Extra Latency** | **Output Tokens** | **Throughput (tok/s)** |
| Qwen2-VL vLLM | 57.0 | 757.4 | – | 121.58 | 160.5 |
| SSV-CoT vLLM | 68.0 | 899.0 | +18.7% | 143.85 | 160.0 |

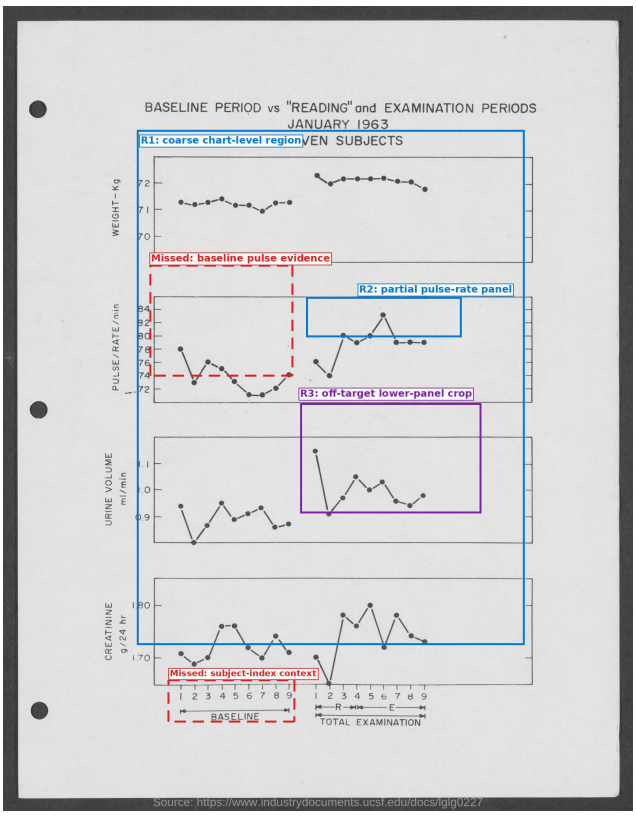
| (b) Adaptive stopping behavior grouped by question complexity | | | | |
|---|---|---|---|---|
| **Complexity** | **Samples** | **Avg. #Regions** | **Stop Early $\leq$2** | **Query More $\geq$3** |
| Simple | 179 | 2.16 | 66.0% | 34.0% |
| Medium | 427 | 2.67 | 43.0% | 57.0% |
| Complex | 394 | 3.05 | 28.0% | 72.0% |
| Overall | 1000 | 2.72 | 41.0% | 59.0% |

| (c) Ablation of the global complement region on 200 MathVista samples | | | |
|---|---|---|---|
| **Method** | **MathVista Acc.** | **Avg. #Regions** | **Stop Rate** |
| SSV-CoT w/ global complement | 68.0 | 2.63 | 36.5% |
| SSV-CoT w/o global complement | 66.5 | 2.58 | 42.0% |

Table 9: Additional analysis of SSV-CoT. (a) SSV-CoT improves accuracy with a moderate inference-latency increase while maintaining nearly the same token-level throughput. (b) The learned `STOP` action adapts the visual budget according to question complexity. (c) The global complement region provides modest auxiliary contextual benefit.

sary fine-grained evidence, sequential visual access alone cannot fully recover it. Future extensions may incorporate multi-scale region proposals, iterative region refinement, or dynamic region generation during reasoning.

## Failure Case: Region-bank construction misses fine-grained chart evidence




Solid boxes: selected regions
Dashed red boxes: missed fine-grained evidence

**Input Question**

Which subject has the highest pulse rate during the baseline period?

**Ground truth:** Subject 1

**Selected Region Bank**

**Region 1:** Coarse page-level region, providing global context but too diffuse for precise chart reading.

**Region 2:** Partial pulse-rate panel, covering only part of the baseline segment and missing subject-index context.

**Region 3:** Off-target lower-panel crop, containing weakly related visual evidence.

**Reasoning Outcome**

**Model prediction:** 84

**Error:** The model confuses a plotted pulse-rate value with the subject index.

**Cause:** The policy can only select from the precomputed region bank; missing fine-grained evidence cannot be recovered during later reasoning.

**Takeaway.** This case shows that one-shot region-bank construction can become a bottleneck for dense chart reasoning. When critical fine-grained evidence is not included in the initial candidate regions, the sequential visual policy cannot generate a new high-resolution region during later reasoning. This motivates future extensions such as multi-scale region proposals, iterative region refinement, or dynamic region generation during CoT reasoning.

Figure 4: A representative failure case caused by imperfect region-bank construction. Although SSV-CoT performs sequential visual access, the policy can only choose from the precomputed candidate region bank. If the bank misses critical fine-grained chart evidence, the model may still produce an incorrect answer.