StAR: Segment Anything Reasoner

Seokju Yun[1,2], Dongheon Lee[2], Noori Bae[2], Jaesung Jun[2], Chanseul Cho[2], and Youngmin Ro[2†]

[1] KAIST AI
[2] University of Seoul
https://github.com/ysj9909/StAR

Abstract. As AI systems are being integrated more rapidly into diverse and complex real-world environments, the ability to perform holistic reasoning over an implicit query and an image to localize a target is becoming increasingly important. However, recent reasoning segmentation methods fail to sufficiently elicit the visual reasoning capabilities of the base model. In this work, we present **S**egmen**t** **A**nything **R**easoner (**StAR**), a comprehensive framework that refines the design space from multiple perspectives—including parameter-tuning scheme, reward functions, learning strategies and answer format—and achieves substantial improvements over recent baselines. In addition, for the first time, we successfully introduce parallel test-time scaling to the segmentation task, pushing the performance boundary even further. To eXtend the scope and depth of reasoning covered by existing benchmark, we also construct the ReasonSeg-X, which compactly defines reasoning types and includes samples that require deeper reasoning. Leveraging this dataset, we train StAR with a rollout-expanded selective-tuning approach to activate the base model's latent reasoning capabilities, and establish a rigorous benchmark for systematic, fine-grained evaluation of advanced methods. With only 5k training samples, StAR achieves significant gains over its base counterparts across extensive benchmarks, demonstrating that our method effectively brings dormant reasoning competence to the surface.

Keywords: Reasoning Segmentation · Reinforcement Learning · Test-time Scaling

1 Introduction

Embodied AI agents are increasingly expected to operate beyond bounded lab setups, in diverse daily scenes where user intent is often implicit and incomplete. In this regime, recognizing object categories is not sufficient; an agent must reason about the user's goal in context and localize the relevant regions that support safe and effective action. Reasoning segmentation [18] formalizes this requirement by demanding adaptive cognitive reasoning that interprets an implicit query in accordance with the context of a given image. In particular,

† Corresponding author

beyond a superficial interpretation of the scene, the ability to (i) flexibly apply information given in the query or world knowledge, (ii) comprehensively understand the background context and relations among entities, or (iii) reason step-by-step over multiple components has become an important challenge toward general artificial intelligence (see Fig. 1).

Following the groundwork laid by LISA [18], several reasoning segmentation methods [3,5,6,11,14,18,24–27,29,32,42,53] have demonstrated promising results by integrating the versatile reasoning capabilities of multimodal large language models (MLLMs) [1, 2, 22, 23, 40] with the robust mask generation ability of the SAM series [16, 30]. Furthermore, inspired by the expanded reasoning capacity boundary of leading MLLMs and the remarkable effectiveness of *Reinforcement Learning with Verifiable Rewards* (RLVR) [8,19], recent methods [11,24,25] adopt Group Relative Policy Optimization (GRPO) [34] for MLLMs, effectively enhancing the ability to localize target regions through visual reasoning. For example, VisionReasoner [25] introduces a decoupled "reasoning module-mask generator" framework and successfully applies GRPO using a compact reward design with a multi-object matching algorithm. However, despite their empirical success, the bottleneck of current RLVR frameworks remains underexamined. This raises a fundamental question: ***Does the current RLVR framework maximally elicit visual reasoning potential that does not readily manifest? If not, which component is responsible for the bottleneck?***

To systematically answer this question, we investigate all components that constitute the pillars of RLVR and try to identify design decisions that fully leverage the base model's pre-trained capacity. We start from VisionReasoner employing Qwen2.5-VL 7B [2] as the MLLM backbone, and gradually "retrofit" the framework to resolve the reasoning bottleneck in each component.
Specifically, our exploration centers on the following pivotal dimensions:

- **Parameter-tuning scheme**: full-parameter tuning requiring careful hyperparameter adjustment → LoRA [10] training with an appropriate configurations (high-rank and large learning rate) for efficiently enhancing reasoning.
- **Reward design**: MLLM-level + SAM-level fine-grained reward function for mask-aware gradients and rollout scalability.
- **Learning strategy**: vanilla GRPO + rollout-expanded selective-tuning for propagating more informative training signals through broadened exploration.
- **Answer format**: bbox/point prediction + label prediction for promoting reasoning faithfulness and visually grounded reasoning.

Through this process, we uncover several key bottlenecks that severely compromise performance. We then eliminate these bottlenecks in a computationally efficient manner, simply yielding large performance improvements. As a result, we propose a comprehensive framework, **S**egmen**t** **A**nything **R**easoner (**StAR**).

Enabling LLMs to improve their outputs by allocating additional test-time compute [36] is a critical step in post-training. A simple yet effective approach is *majority voting* [43], which generates N responses in parallel and selects the most frequent one as the final answer. However, while this strategy is naturally applicable to tasks such as mathematical reasoning—where the equality

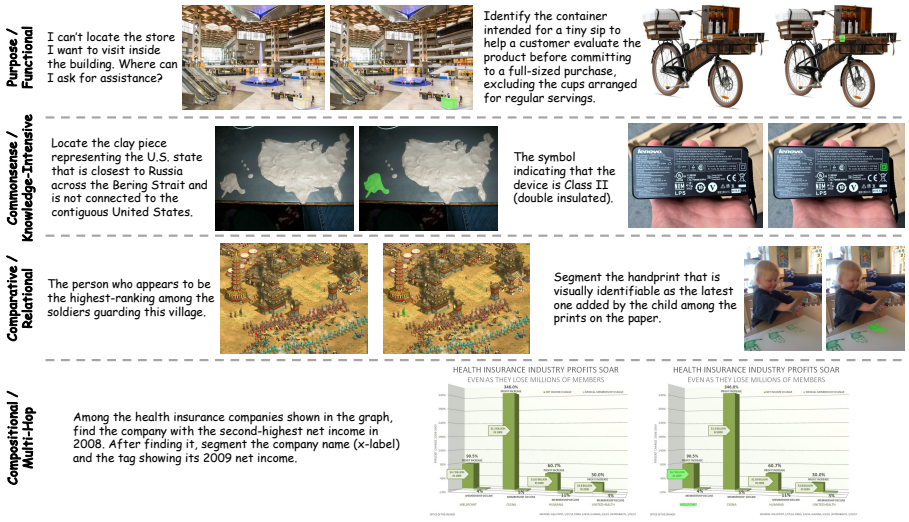

Fig. 1: We establish a reasoning segmentation benchmark, *ReasonSeg-X*, that demands a broad range of reasoning skills. Our dataset addresses goal-oriented reasoning and the ability to flexibly invoke world knowledge, while also covering complex relational reasoning and step-by-step reasoning capabilities. Our model, StAR (equipped with Qwen3-VL 32B [1]), demonstrates remarkable performance across all these aspects (*right* image). Additional results are provided in the supplementary material.

of two answers can be easily determined—it is not well-suited to pixel-level prediction tasks. Therefore, instead of a per-pixel voting scheme, we adopt a *mask-level* strategy that selects a representative mask set via IoU-based clustering. By leveraging the diverse and vast reasoning pattern space of StAR, our test-time scaling (TTS) approach consistently improves performance across extensive benchmarks, enabling StAR to tackle increasingly complex problems through more inference-time compute. Moreover, our *parallel sampling* approach opens a new research direction that lies on the opposite axis of the recent SAM 3 agentic pipeline [5] centered on *sequential refinement*.

LISA [18] also introduced ReasonSeg benchmark, where images are annotated with implicit text instructions and target masks. Unfortunately, ReasonSeg has the following limitations: (i) some samples exhibit low mask quality or contain reasoning errors; (ii) its limited reasoning depth makes it difficult to effectively evaluate methods leveraging frontier MLLMs; and (iii) because the benchmark does not clearly define the reasoning scope and type it covers, it cannot systematically assess recent methods across diverse aspects. To mitigate these, we first introduce a Refined version of ReasonSeg (ReasonSeg-R) that addresses issue (i), aiming to prevent currently proposed methods from being misled by incomplete evaluations and thereby support proper research progress (see Fig. 2). In addition, we construct an eXtended version of ReasonSeg, termed ***ReasonSeg-X***, which deepens the reasoning depth and systematically expands

the coverage of reasoning, complementing issues (ii)–(iii). A distinguishing characteristic of the ReasonSeg-X is its balanced coverage of samples across four reasoning types (purpose/functional, commonsense/knowledge-intensive, comparative/relational, and compositional/multi-hop reasoning—see Fig. 1 and Sec. 2). We utilize this dataset to support StAR in developing its reasoning segmentation capabilities across various dimensions.

We conduct experiments on several reasoning segmentation benchmarks, including ReasonSeg [18], MMR [14], MUSE [32], and our proposed ReasonSeg-X/R, and observe that our StAR (i) showcases substantial performance gains over their base counterparts by preserving the base model's inherent capacity and effectively adapting it to the segmentation task, and (ii) maximizes the efficacy of scaling test-time compute and model size by surfacing "latent, hard-to-elicit" reasoning capabilities through a train-time exploration scaling strategy that increases exposure to diverse reasoning patterns.

2 ReasonSeg-X Benchmark

LISA [18] introduces ReasonSeg, which handles implicit query requiring intricate reasoning, rather than explicit query such as category names. Building on this, PixelLM [32] proposes MUSE dataset to facilitate multi-target reasoning, while MMR [14] further extends the scope by curating samples that incorporate part-level reasoning. However, while these studies primarily consider segmentation-level formulation (mask prediction format), we focus on the largely underexplored aspects of **reasoning depth and reasoning type design**. Through this, we define the following four compact reasoning taxonomy:

- **Purpose/Functional (P/F)**: a reasoning type that requires the ability to locate a target object needed to achieve a specific purpose, or intended for a particular use.
- **Commonsense/Knowledge-Intensive (C/KI)**: a reasoning type that particularly requires commonsense or domain-specific knowledge to localize the target region. Moreover, this covers everyday world knowledge as well as human-like visual priors accumulated through experience.
- **Comparative/Relational (C/R)**: a reasoning type in which the target must be localized through comparisons or relationships between entities in the image. This type includes comparative reasoning over diverse attributes as well as holistic reasoning about inter-entity relations.
- **Compositional/Multi-Hop (C/MH)**: a reasoning type that requires step-by-step composition of multiple evidences, involving challenging cases that integrate complex constraints or apply various reasoning skills sequentially.

When designing this reasoning taxonomy, we aim to make each type as mutually exclusive as possible; however, in some cases the boundary inevitably becomes blurred. For example, "finding an object for a particular use" almost always involves commonsense/knowledge (P/F $\leftrightarrow$ C/KI). Nevertheless, to foster an agent's goal-directed reasoning ability, we construct P/F by distinguishing it from C/KI in a strict and careful manner. Moreover, C/R and C/MH samples

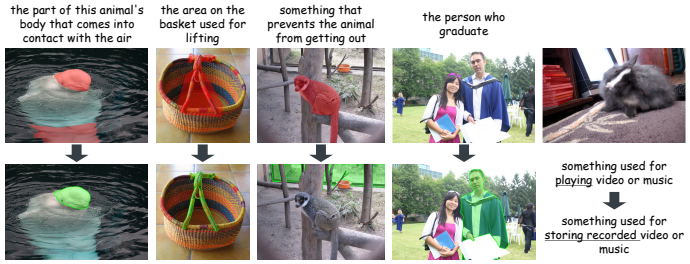

Fig. 2: ReasonSeg-R examples. We correct masks for cases that are mismatched to the query and also refine mask quality. In addition, we modify query expressions that may inadvertently include regions outside the mask (*e.g.*, the video player monitor).

exhibit reasoning depth that has been rarely covered in existing datasets, thereby posing new challenges to current methods. Examples are shown in Fig. 1.

To generate our dataset, we curate diverse and context-rich images from OpenImages [17] and, based on the reasoning type design above, meticulously annotate them with implicit query and high-quality target mask. For each generated sample, we perform cross-validation among annotators and employ GPT-5 [35] and Gemini 3 [7] for additional verification to further establish reasoning validity. The resulting dataset, ***ReasonSeg-X***, comprises 1,169 data samples, split into 240/156/773 samples for `train`/`val`/`test`. Sample counts by reasoning type (in the aforementioned type order) are 53/33/88/66 for `train`, 37/36/48/35 for `val`, and 201/181/253/138 for `test`.

ReasonSeg-R. ReasonSeg is the first benchmark specifically designed for reasoning segmentation, providing an important foundation for follow-up studies. However, our inspection reveals that some data samples suffer from poor mask quality or contain reasoning errors (see Fig. 2). Therefore, to facilitate a more reliable assessment in this research area, we provide a refined version of ReasonSeg, termed *ReasonSeg-R*. Specifically, we rigorously verify query-reasoning-mask correspondence and further check whether each mask accurately covers the target region along its boundary. As a result, we refine the masks of 113 samples and the queries of 3 samples, while excluding 13 unsuitable samples. In addition, since many recent methods do not use the ReasonSeg `train` set, we merge the `val` and `test` sets for a streamlined evaluation process. Additional details on our presented datasets can be found in the supplementary material.

3 Methodology

3.1 Pipeline Overview and Preliminaries

In this section, we detail the components that constitute the foundation of the StAR framework. Following prior works [24, 25], StAR adopts a reasoning-segmentation decoupled design: an MLLM performs reasoning over the input query-image pair to predict bbox/point coordinates, which are subsequently used

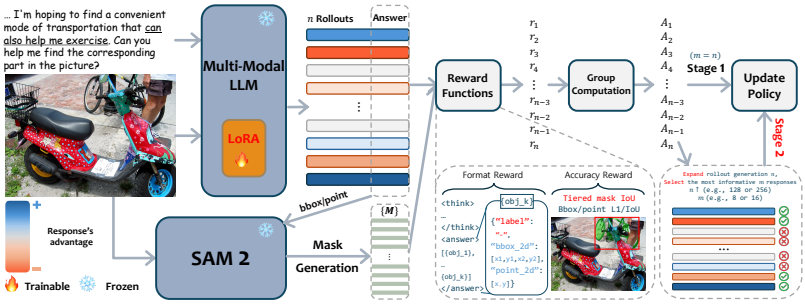

Fig. 3: Overview of the StAR framework pipeline. We pinpoint and resolve reasoning bottlenecks from all perspectives of the existing reasoning segmentation framework (highlighted in red text). See text for details.

as visual prompts for SAM 2 [30] to extract the final masks. The pipeline of our framework is illustrated in Fig. 3. Specifically, given a text query $\mathbf{T}$ and the input image $\mathbf{I}$, the reasoning module $\mathcal{F}_{MLLM}$ generates a chain-of-thought (CoT) [44] reasoning trace and then predicts the bounding boxes $\{\mathbf{B}_i\}_{i=1}^{N_{pred}}$ and representative points $\{\mathbf{P}_i\}_{i=1}^{N_{pred}}$ for the target regions associated with the query. Lastly, these predictions are passed to the segmentation module $\mathcal{F}_{SAM}$ to produce the final binary masks $\{\mathbf{M}_i\}_{i=1}^{N_{pred}}$. This procedure can be formulated as:

$$(\{\mathbf{B}_i, \mathbf{P}_i\})_{i=1}^{N_{pred}} = \mathcal{F}_{MLLM}(\mathbf{I}, \mathbf{T}), \tag{1}$$

$$\{\mathbf{M}_i\}_{i=1}^{N_{pred}} = \mathcal{F}_{SAM}(\mathbf{I}, (\{\mathbf{B}_i, \mathbf{P}_i\})_{i=1}^{N_{pred}}). \tag{2}$$

With this forward pipeline, we keep SAM 2 frozen and apply RLVR directly on the MLLM to enhance its visual reasoning capability.

Group Relative Policy Optimization (GRPO). We adopt GRPO [34], a variant of Proximal Policy Optimization [33], as our core RLVR algorithm. Instead of using a value function, it estimates the advantage using a *normalized reward* within a group of responses to the same prompt. Specifically, for each question q, GRPO first samples a group of n rollouts $\{o_i\}_{i=1}^{n}$ from the old policy $\pi_{\theta_{old}}$; each rollout is then evaluated with reward functions, producing a reward vector $\{r_i\}_{i=1}^{n}$. Finally, GRPO optimizes the policy model by maximizing the following objective:

$$\begin{aligned}\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^{n} \sim \pi_{\theta_{old}}(O|q)] \\ \frac{1}{n}\sum_{i=1}^{n}\left(\min\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}A_i, \text{clip}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}, 1-\epsilon, 1+\epsilon\right)A_i\right) - \beta\mathbb{D}_{KL}\left(\pi_\theta||\pi_{ref}\right)\right),\end{aligned} \tag{3}$$

where ϵ and β are hyperparameters, and A_i is the advantage, calculated using a group of rewards $\{r_i\}_{i=1}^{n}$ corresponding to the outputs inside each group:

$$A_i = \frac{r_i - \text{mean}(\{r_i\}_{i=1}^{n})}{\text{std}(\{r_i\}_{i=1}^{n})}. \tag{4}$$

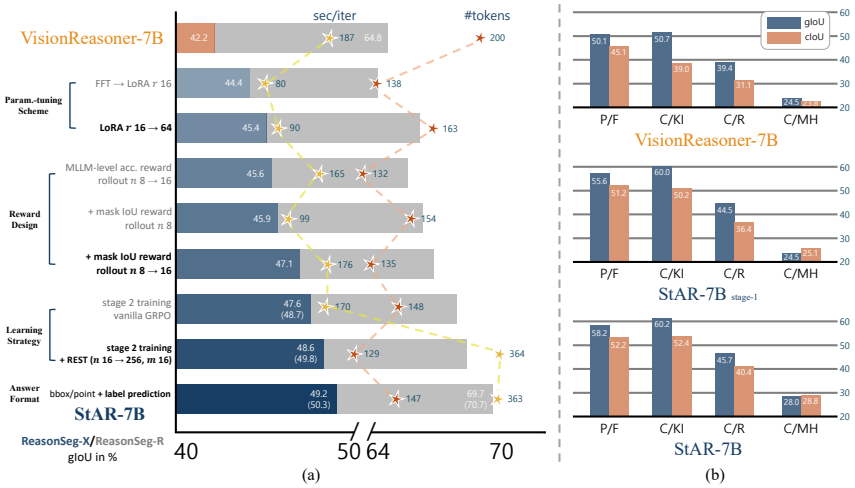

Fig. 4: (a) We progressively retrofit all pillars of the standard RLVR framework (VisionReasoner) toward our performant and robust model (StAR), without introducing significant computational overhead. The foreground bars show performance on ReasonSeg-X `test`, while results on ReasonSeg-R are shown with gray bars. The performance values in parentheses are obtained with our majority voting strategy (Sec. 3.3). Gray-marked design decisions (y-axis label texts) indicate comparison-only variants that are not adopted. **(b)** The proposed RLVR design space better preserves and exploits the base MLLM's diverse reasoning capabilities (VisionReasoner → StAR stage-1). Moreover, Stage 2 training on the ReasonSeg-X `train` set with rollout-expanded selective-tuning (REST) effectively elicits the MLLM's hidden reasoning potential, improving performance particularly on complex reasoning (*e.g.*, C/MH). The resulting model, StAR, outperforms the baseline by a large margin across all reasoning types.

Training Pipeline. We train our model via *two stages*, employing GRPO throughout both phases. Stage 1 is primarily designed to elicit flexible localization capabilities while maximally preserving pre-trained visual perception abilities. Accordingly, we utilize the training set collected by VisionReasoner [25], *excluding the reasoning data samples* (LISA++ [48]). Thus, the Stage-1 training set is sourced from LVIS [9], RefCOCOg [49], and gRefCOCO [21], and consists of **5k** samples in total. Stage 2 aims to further strengthen the model's reasoning segmentation capabilities by effectively bringing out the base model's hidden reasoning potential. To this end, we continue finetuning the Stage-1 model on our proposed *ReasonSeg-X* `train` set (240 samples), resulting in **StAR**, the final model tailored for complex and versatile visual reasoning.

3.2 Roadmap to Segment Anything Reasoner (StAR)

In this section, we present the design evolution that "retrofits" VisionReasoner [25] by alleviating key reasoning bottlenecks. Our explorations focus on investigating performance-degrading factors along each design axis and mitigating them in a

computationally efficient manner. We start from VisionReasoner-7B and study a series of design decisions summarized as (1) parameter-tuning scheme, (2) reward design, (3) RL strategy, and (4) answer format. As illustrated in Fig. 4 (a), our design trajectory toward StAR addresses bottlenecks across all axes and yields non-trivial performance gains (+7.0/4.9% gIoU on ReasonSeg-X/R).

Parameter-tuning scheme. Unlike supervised fine-tuning scenarios for dense knowledge transfer and for acquiring new reasoning patterns, applying RLVR to the reasoning segmentation (RS) task requires following a principle: "***preserve*** *knowledge and reasoning patterns, and* ***learn to leverage*** *them for segmentation*". Yet RLVR-based RS methods [11,25], as well as most existing open-source reasoning models [28,39,45,50], typically rely on expensive *full-parameter tuning*. This approach is notoriously resource-intensive and often requires extra model copies—such as a reference model for KL penalty—greatly increasing memory usage; moreover, adjusting all weights risks substantially damaging the base model's underlying knowledge. We therefore investigate *parameter-efficient tuning* to maximally maintain the base model's inherent capacity while efficiently enhancing reasoning capabilities.

Low-rank adaptation (LoRA) [10] uses a low-rank decomposition to model sparse updates [20], thereby largely preserving the base model's world knowledge and demonstrating robust performance [4, 38, 41, 51]. Building on these observations, we attempt to employ LoRA in the RLVR setting. However, when using the default hyperparameter configuration, we observe a substantial performance drop compared to the full-parameter tuning baseline. To better promote task adaptation, we increase the learning rate from 1×10^{-6} to 1×10^{-5} and exclude the KL penalty term. With this configuration and a LoRA rank of 16, we achieve comparable or even superior performance while reducing training costs by more than 2× compared to VisionReasoner. We then increase the LoRA rank from 16 to 64 to expand the expressiveness of the update matrices, allowing the model to tackle more complex reasoning [12]. Overall, our derived tuning scheme achieves surprisingly strong performance (42.2% → 45.4%/64.8% → 66.1%) *by following the aforementioned principle.*

Reward design. Reward functions are central to RLVR. However, the accuracy reward functions used in VisionReasoner are not tightly aligned with the task's ultimate goal—mask generation—increasing the risk of reward hacking (Fig. 5, *left*). In addition, the prevailing "hard" reward (1 if IoU exceeds a fixed threshold, otherwise 0) neither reflects incremental improvements throughout training nor enables fine-grained evaluation. To remedy these issues, inspired by [11], we incorporate SAM into the RLVR training loop and introduce a piecewise mask-IoU reward to provide mask-aware feedback. This design provides graded signals that directly encourage incremental improvements in mask quality (Fig. 5, *right*).

We employ a tiered mask reward function based on IoU to discretize segmentation quality into multiple levels. The reward is computed as follows (additions

Fig. 5: Motivation for the mask IoU reward. *Left*: Misalignment between the MLLM-level reward function and the task's final objective potentially induces confusing learning signals. *Right*: Distinguishing between diverse mask defects via a fine-grained tiered mask reward promotes stable and stepwise optimization.

relative to SAM-R1 [11] are marked in red):

$$\text{mask reward} = \begin{cases} 5, & \text{IoU} > 0.90, \\ 4, & 0.80 < \text{IoU} \leq 0.90, \\ 3, & 0.70 < \text{IoU} \leq 0.80, \\ 2, & 0.50 < \text{IoU} \leq 0.70, \\ 1, & 0.30 < \text{IoU} \leq 0.50, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

Along with this SAM-level reward, we also retain the reward functions from VisionReasoner that evaluate the accuracy of the MLLM's predictions. Specifically, we use (i) a bbox IoU reward that assigns 1 when the predicted bbox exceeds 0.5 IoU with the GT bbox, and (ii) a bbox/point L1 reward that assigns 1 when the L1 distance between predicted and GT bbox/point is below 10/30 pixels. We then solve the multi-object matching problem over N_{pred} predictions and N_{GT} instance annotations using a batched Hungarian algorithm [25]. Based on the resulting assignment, the final accuracy reward is calculated and then divided by $\max\{N_{pred}, N_{GT}\}$ to suppress over- and under-segmentation errors. Additionally, VisionReasoner's Thinking/Answer/Non-repeat reward functions are applied to ensure a structured and reliable output format.

Maintaining MLLM-level accuracy rewards bolsters coarse localization capabilities, particularly by imposing stronger penalties on reasoning failures. Moreover, as illustrated in Fig. 5 *right*, our fine-grained mask reward delivers a *curriculum-like training signal*: early training emphasizes locating the target via high-quality reasoning, while later stages focus on accurately segmenting target region boundaries. With this comprehensive reward design, increasing rollout utilization ($8 \rightarrow 16$) brings meaningful performance gains—unlike VisionReasoner—suggesting that our rewards discriminate rollouts more finely and offer the richest feedback. Notably, by reallocating the resources saved by LoRA training to rollout scaling, we improve performance ($45.4\% \rightarrow 47.1\%/66.1\% \rightarrow 66.6\%$) while keeping overall training costs modest.

Reinforcement learning (RL) strategy. While Stage 1 training primarily leverages referring segmentation datasets and focuses on utilizing the base

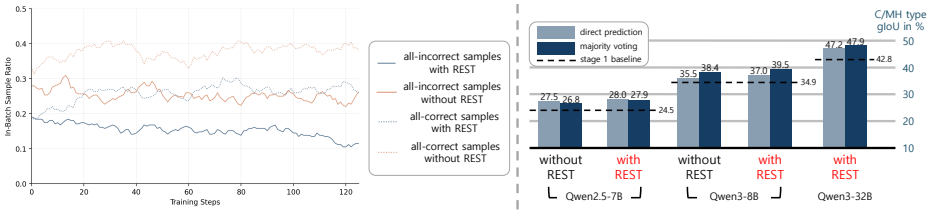

Fig. 6: *Left*: Rollout-expanded selective-tuning (REST) substantially improves sample utilization. In particular, for hard samples, it develops the model's intricate reasoning by exploring a vast space of reasoning paths. "Correct" is defined as IoU $>$ 0.5, and results are reported with the Qwen3-VL 8B model. *Right*: REST effectively boosts performance on problems that require multi-step reasoning; moreover, **its gains synergistically scale with the base model's reasoning capability boundaries.**

model's perceptual capacity for localization, Stage 2 mainly aims to cultivate the ability to solve complex reasoning problems using our proposed dataset. However, Stage 2 GRPO training often suffers from *advantage vanishing*: for samples that are too easy or too hard, the model generates group responses with consistent rewards, severely reducing learning efficiency. As depicted in Fig. 6 *left*, during vanilla GRPO training, on average, more than half of the batch samples exhibit a near-zero-variance reward distribution. Moreover, the sparse positive feedback on extremely challenging problems prevents fully drawing out the base model's hidden reasoning potential.

To mitigate this, we propose Rollout-Expanded Selective-Tuning (REST) as a drop-in modification to GRPO. REST **decouples exploration from learning**: we first *expand* the rollout pool from the conventional small-n regime (*e.g.,* 8-16) to a large-N regime (*e.g.,* 128-256) to explore a much wider set of reasoning trajectories. Instead of updating on all N rollouts, REST then performs *advantage-extremal selection*, updating only on an informative subset of size m: the $m/2$ rollouts with the largest advantages and the $m/2$ rollouts with the smallest advantages. Selecting both tails aligns with the reward variance-maximization strategies [31, 46], yielding stronger contrastive signals for policy improvement.

Furthermore, REST cleverly exploits the compute/memory asymmetry in LLM RL. The rollout generation is highly parallel, requires minimal activation storage, and is typically high-throughput when batched—so generating many rollouts can be amortized efficiently. By contrast, policy updates are memory- and communication-intensive because of optimizer states and cross-device synchronization. Consequently, exploring a 16$\times$ larger reasoning pattern space (n 16 $\rightarrow$ 256) increases the end-to-end training wall-clock time by only about 2$\times$ (Fig. 4 (a)). In other words, REST keeps the expensive part nearly constant while scaling the cheaper, batched inference part.

Importantly, for hard prompts, REST increases the chance of observing rare successful reasoning paths, efficiently eliciting and internalizing "latent, dormant" intricate reasoning skills (Fig. 6). Leveraging our dataset with REST for Stage-2 training boosts performance on ReasonSeg-X/R by 1.5/1.6%, respectively.

Answer format. Reasoning segmentation is inherently *semantic-first*: the model must interpret the query, perform compositional visual-language inference, and identify target regions consistent with that semantics. However, the endpoint of the task is *geometric*, which can implicitly bias generation toward a "geometry-only" mode where the model focuses on producing plausible coordinates without committing to a grounded semantic identification. Meanwhile, recent studies [37, 47] reveal that as CoT reasoning progresses, visual attention tends to markedly diminish, which in turn exacerbates hallucination. We therefore introduce Label Prediction (LP) as a simple and lightweight answer-format augmentation. Rather than predicting only geometry for each target, the model is additionally required to emit a short semantic label that describes the localized entity (*e.g.*, {"label": "the owner of the largest dog", "bbox_2d": ..., "point_2d": ...}). Crucially, while this modification merely adjusts the prompt/response schema, it systematically reshapes generation dynamics by introducing an explicit semantic "naming" step immediately before coordinate prediction.

Interestingly, predicted labels often paraphrase expressions in the query or take the form of answering the query, thereby semantically anchoring the predicted region to the query and ultimately enhancing reasoning faithfulness. LP also increases StAR-7B's attention mass over visual tokens during coordinate prediction from 5.3% to 6.3%. This suggests that, in contrast to methods that explicitly strengthen visual dependency during training/inference [13,15], a simple form of strategic prompting—requiring label prediction—can efficiently and effortlessly facilitate visually-grounded reasoning. As a result, this remarkably simple solution improves performance from 48.6/68.2% to 49.2/69.7%. *This design trajectory brings us to our final model,* ***StAR****.*

3.3 Majority Voting for Segmentation

Majority voting [43] has proven effective for tasks with a discrete answer space, where "voting" is naturally well-defined. *Here, we introduce a mask-level clustering strategy that adapts majority voting to pixel-level, multi-object prediction.* Given a pool of mask candidates collected from all parallelly generated responses, we group all candidates into clusters using a greedy procedure: for each candidate, we compute its IoU with the representative mask of existing clusters; if the IoU exceeds a certain threshold ($= 0.85$), we assign it to that cluster, otherwise we create a new cluster. We then decide the number of target instances $\hat{K}$ by the mode of the per-response predicted target counts, and if *no-target* prediction is in the majority, we also output "no target object". Subsequently, we rank clusters by vote count and select the top $\min(\hat{K}, \#\text{clusters})$ clusters. For each selected cluster, we choose the single most confident instance mask, leveraging SAM's mask quality score [30] for robust aggregation. Finally, we compose the overall prediction by taking the union of the selected instance masks. **This strategy is generally applicable to any MLLM-based segmentation framework and brings substantial performance gains when paired with our StAR's vast reasoning capacity boundary (see Tab. 1).**

Table 1: Performance results on the ReasonSeg (RS) series. "MV" denotes our majority voting strategy adapted for segmentation. We compare our StAR against state-of-the-art baselines: LISA [18], SegLLM [42], READ [29], CoReS [3], RSVP [26], CoPRS [27], SAM-R1 [11], SAM-Veteran [6], VisionReasoner [25] and SAM 3 Agent [5].

Method	Base model (Size)	RS		RS-R		RS-X		RS-X test									
		test		–		val		overall		P/F		C/KI		C/R		C/MH	
		gIoU	cIoU	gIoU	cIoU	gIoU	cIoU	gIoU	cIoU	gIoU	cIoU	gIoU	cIoU	gIoU	cIoU	gIoU	cIoU
Supervised/Reinforcement Finetuning Methods																	
LISA (ft)	Llama2 13B	51.5	51.3	52.5	53.3	23.8	24.3	25.1	26.0	27.9	30.6	28.2	29.2	26.8	25.8	13.8	16.8
SegLLM	LLaVA1.5 7B	52.4	48.4	–	–	–	–	–	–	–	–	–	–	–	–	–	–
READ (ft)	LLaVA1.5 13B	62.2	62.8	–	–	–	–	–	–	–	–	–	–	–	–	–	–
CoReS (ft)	LLaVA1.5 13B	65.5	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RSVP	GPT-4o	60.3	60.0	–	–	–	–	–	–	–	–	–	–	–	–	–	–
CoPRS	Qwen2.5-VL 7B	59.8	55.1	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SAM-R1	Qwen2.5-VL 7B	60.2	54.3	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SAM-Veteran	Qwen2.5-VL 7B	62.6	56.1	–	–	–	–	–	–	–	–	–	–	–	–	–	–
VisionReasoner	Qwen2.5-VL 7B	63.6	55.7	64.8	56.8	44.1	37.6	42.2	33.8	50.1	45.1	50.7	39.0	39.4	31.1	24.5	23.8
$\text{StAR}_{\text{stage-1}}$	Qwen2.5-VL 7B	66.7	60.8	69.0	65.6	48.5	42.7	47.4	40.6	55.6	51.2	60.0	50.2	44.5	36.4	24.5	25.1
Training-free Inference-time Scaling Methods																	
SAM 3 Agent	Qwen2.5-VL 7B	62.6	56.2	63.1	58.0	34.1	28.0	34.4	29.5	41.7	37.9	37.7	31.8	35.3	26.0	17.9	21.9
SAM 3 Agent	Qwen2.5-VL 72B	71.8	65.2	72.4	65.3	52.0	43.0	49.8	40.3	57.8	50.5	55.5	45.4	49.6	34.0	30.8	35.1
StAR	Qwen2.5-VL 7B	67.5	61.3	69.7	66.2	50.5	48.0	49.2	43.6	58.2	52.2	60.2	52.4	45.7	40.4	28.0	28.8
StAR + MV	Qwen2.5-VL 7B	68.5	65.0	70.7	67.2	51.3	48.4	50.3	44.9	60.6	56.9	61.8	54.4	46.2	40.1	27.9	30.5
StAR	Qwen3-VL 8B	70.6	63.8	73.8	69.2	60.8	57.2	57.9	50.1	69.1	55.9	66.2	60.3	54.4	45.0	37.0	41.4
StAR + MV	Qwen3-VL 8B	71.8	66.5	74.9	69.7	62.6	**60.1**	59.6	53.1	69.0	58.6	68.5	66.4	56.7	47.9	39.5	41.7
StAR	Qwen3-VL 32B	72.1	67.3	73.8	68.0	61.5	54.2	61.7	59.2	71.7	**71.6**	66.7	66.9	58.2	50.8	47.2	**51.5**
StAR + MV	Qwen3-VL 32B	**72.7**	**68.1**	**75.0**	**70.6**	**64.7**	57.5	**64.1**	**60.8**	**74.0**	69.6	**70.2**	**71.0**	**60.7**	**54.2**	**47.9**	50.3

Table 2: Comparison on the Multi-target and Multi-granularity Reasoning (MMR) dataset. * indicates models trained on a training set that includes the MMR dataset.

Method	Base model (Size)	val		test					
		Obj & Part		Obj & Part		Obj		Part	
		gIoU	cIoU	gIoU	cIoU	gIoU	cIoU	gIoU	cIoU
LISA [18]	Llama2 13B	15.4	20.0	16.1	19.8	26.1	27.9	7.4	8.4
LISA* [18]	Llama2 13B	22.3	33.4	23.0	29.2	40.2	45.2	10.7	16.4
M^2SA^* [14]	LLaVA 7B	27.8	48.6	30.9	46.8	41.0	55.6	13.5	27.0
M^2SA^* [14]	Llama2 13B	28.4	**49.1**	31.6	**47.6**	42.3	**57.6**	13.6	**27.2**
VisionReasoner [25]	Qwen2.5-VL 7B	26.7	21.4	28.4	21.7	37.2	26.5	11.5	8.7
StAR	Qwen2.5-VL 7B	29.8	26.3	32.4	27.2	44.4	33.3	14.3	11.7
StAR + MV	Qwen2.5-VL 7B	30.6	27.1	33.0	27.8	45.6	34.7	14.9	12.0
StAR	Qwen3-VL 8B	31.5	27.0	33.9	26.8	45.0	33.6	15.3	11.8
StAR + MV	Qwen3-VL 8B	**32.4**	27.6	**34.6**	27.7	**46.2**	34.0	**15.8**	11.9

Further details and a discussion of the methodology can be found in the supplementary material.

4 Experiments

4.1 Experimental Setup

Datasets. We follow VisionReasoner [25] for stage 1 training, using 5k explicit/referring expression segmentation (RES) [9, 21, 49] samples, and use our proposed *ReasonSeg-X* `train` set (240 samples) for stage 2. Extending the widely used ReasonSeg [18], we introduce *ReasonSeg-R/X* to evaluate model performance from multiple perspectives and across varying difficulty levels. We further

Table 3: Robotics grounding performance on RefSpatial-Bench [52].

Success rate (%)	RefSpatial-Bench	
Method	**Location**	**Unseen**
Gemini-2.5-Pro	46.96	27.14
RoboRefer-8B [52]	52.00	37.66
StAR-8B	**62.00**	**38.96**

adopt the MMR [14] dataset to assess multi-target and multi-granularity reasoning segmentation capabilities. For out-of-domain evaluation, results on the RES datasets [49] and MUSE [32] are presented in the supplementary material.

Implementation details. We use Qwen2.5-VL 7B [2] and Qwen3-VL 8/32B [1] as our base models, with SAM 2 Large as the mask generator. We set the batch size to 16 and the learning rate to 1×10^{-5} (5×10^{-6} for stage 2). We use LoRA rank 64 for all base models, and apply REST only during stage 2 finetuning. By default, the sampling number of reasoning paths in our majority voting is set to 32. Detailed training settings are provided in the supplementary material.

4.2 Comparison with State-of-the-Arts

Following previous works [18, 25], we adopt gIoU and cIoU as our evaluation metrics: gIoU is the average of per-sample IoUs, while cIoU is computed as the cumulative intersection over the cumulative union.

Results on the ReasonSeg(-R/X) datasets. In Tab. 1, we compare StAR against previous SOTA baselines. Notably, our Stage-1 model shows that it can leverage inherent reasoning capabilities **without using reasoning data**, surpassing methods using the same base model by a large margin. Finetuning on our dataset with REST further strengthens performance especially on complex reasoning. Remarkably, StAR-7B with our majority voting attains overall comparable performance on the ReasonSeg-X `test` set to the SAM 3 Agent (72B), suggesting that *parallel* test-time scaling is also a promising direction. Moreover, by fully exploiting the base model's reasoning pattern space, our approach benefits increasingly from larger base model size, and our test-time scaling strategy improves performance in nearly all scenarios. Fig. 7 illustrates that StAR models are capable of intricate reasoning through step-by-step thinking interleaved with visual inspection. Meanwhile, it is worth noting that while the performance gap between StAR-8B and StAR-32B is not evident on ReasonSeg-R, it becomes pronounced on ReasonSeg-X, which involves more complex reasoning (*e.g.*, C/R and C/MH). *This indicates that our benchmark presents a necessary direction for the advancement of future methods.*

Results on the MMR dataset. In Tab. 2, we evaluate our models in multi-target and multi-granularity setting [14]. StAR models demonstrate strong **zero-shot** performance, outperforming VisionReasoner as well as models [14] trained on the MMR dataset (primarily on gIoU). These results further validate StAR's flexible segmentation capabilities in addition to its reasoning strengths.

Fig. 7: Qualitative results on our ReasonSeg-X `test` set demonstrate that StAR models, via multi-step reasoning, can effectively tackle complex segmentation scenarios necessitating various capabilities, such as comparative/relational reasoning, physics-based 3D awareness, comprehensive pattern analysis, and multi-hop algebraic expansion.

Results on the RefSpatial-Bench dataset. In Tab. 3, we evaluate StAR's spatial referring performance, which is crucial for embodied robots to interact with the 3D physical world. RefSpatial-Bench [52] is a challenging robotics grounding benchmark for evaluating spatial referring with multi-step reasoning. StAR shows strong **zero-shot** performance: it is competitive not only on examples requiring target-object localization (location set), but also on more challenging examples involving complex spatial relations, multiple anchors, hierarchical references, or implied placements (unseen set), thereby demonstrating its generalized robotics grounding capability.

Model efficiency comparisons can be found in the supplementary material.

4.3 Additional Analysis

Inference-aware finetuning for majority voting. Current finetuning algorithms are agnostic to, and thus decoupled from, the test-time strategy. However, such a decoupling between training and test-time inference can be suboptimal; when majority voting is employed at test time, training should not optimize only the model's first attempt, but should also consider multiple attempts under a high-sampling regime. Accordingly, in Tab. 4 *left*, we explore how the number of rollout generations in REST influences performance, with particular emphasis on MV performance. The results show that REST enables the model to surface high-quality reasoning patterns that express its dormant capability to solve

Table 4: *Left*: We evaluate performance under varying degrees of REST **exploration** (the number of rollout generations n), focusing on the efficacy of the *inference-time strategy*; the per-step training time for StAR-7B is reported below. *Right*: We find that reversing the prediction order at test time—after training the model to predict the label before bbox/point coordinates—**causes a large performance degradation.** Default settings are marked in blue.

RS-X test gIoU (%)	REST n			
	16	64	128	256
StAR-7B	48.4	49.2	**49.4**	49.2
+ MV	49.3	49.5	49.8	**50.3**
StAR-8B	57.1	56.9	57.3	**57.9**
+ MV	58.1	58.3	59.0	**59.6**
Sec/iter	**170**	214	260	364

Model	Answer Format / Prediction Order	RS-R		RS-X test	
		gIoU	cIoU	gIoU	cIoU
StAR-7B	label first	**69.7**	**66.2**	**49.2**	**43.6**
	label last	68.2	63.0	46.3	40.1
StAR-8B	label first	**73.8**	69.2	**57.9**	**50.1**
	label last	73.1	**70.8**	55.8	47.7

complex reasoning problems, thereby improving performance—especially under majority voting. The more pronounced improvements observed in MV with increasing rollout generations can be attributed to closer alignment between the sampling distributions used during training and test time. Moreover, these gains increase synergistically with both the exploration degree and the base model's reasoning potential (*i.e.*, model size, pre-trained capacity).

Answer prediction order. As shown in Tab. 4 *right*, we note that label prediction per se is not the crucial factor. Instead, the benefit comes from reformulating localization as a *semantically-conditioned* coordinate prediction step, in which the initially predicted label text serves as an anchor for subsequent grounding. Conversely, we also observe that for a model trained to predict the label last, prompting label-first prediction at test time substantially improves performance.

5 Conclusion

In this work, we present StAR, a comprehensive framework that effectively surfaces the base model's latent reasoning capabilities. We further, for the first time, explore parallel test-time scaling in the reasoning segmentation task and tightly couple training methodologies with inference-time compute strategies. Finally, we construct ReasonSeg-X, a highly challenging evaluation benchmark, establishing a clear and systematic target for future works.

6 Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) [RS-2025-00562400, RS-2022-NR068754], and also supported by Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government [26CS1100, Development of Proprietary Physical AI-based Small-scale Computers and Integrated Soft Suits]

References

1. Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., Ge, W., Guo, Z., Huang, Q., Huang, J., Huang, F., Hui, B., Jiang, S., Li, Z., Li, M., Li, M., Li, K., Lin, Z., Lin, J., Liu, X., Liu, J., Liu, C., Liu, Y., Liu, D., Liu, S., Lu, D., Luo, R., Lv, C., Men, R., Meng, L., Ren, X., Ren, X., Song, S., Sun, Y., Tang, J., Tu, J., Wan, J., Wang, P., Wang, P., Wang, Q., Wang, Y., Xie, T., Xu, Y., Xu, H., Xu, J., Yang, Z., Yang, M., Yang, J., Yang, A., Yu, B., Zhang, F., Zhang, H., Zhang, X., Zheng, B., Zhong, H., Zhou, J., Zhou, F., Zhou, J., Zhu, Y., Zhu, K.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025)
2. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., et al.: Qwen2. 5-vl technical report. arXiv preprint arXiv:2502.13923 (2025)
3. Bao, X., Sun, S., Ma, S., Zheng, K., Guo, Y., Zhao, G., Zheng, Y., Wang, X.: Cores: Orchestrating the dance of reasoning and segmentation. In: ECCV (2024)
4. Biderman, D., Ortiz, J.G., Portes, J., Paul, M., Greengard, P., Jennings, C., King, D., Havens, S., Chiley, V., Frankle, J., et al.: Lora learns less and forgets less. TMLR (2024)
5. Carion, N., Gustafson, L., Hu, Y.T., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K.V., Khedr, H., Huang, A., et al.: Sam 3: Segment anything with concepts. In: ICLR (2026)
6. Du, T., Li, H., Fan, Z., Zhang, J., Pan, P., Zhang, Y.: SAM-veteran: An MLLM-based human-like SAM agent for reasoning segmentation. In: ICLR (2026)
7. Gemini Team, Google DeepMind: Gemini 3: A new era of intelligence with gemini 3. Tech. rep. (2025), `https://blog.google/products-and-platforms/products/gemini/gemini-3/`, technical Report
8. Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al.: Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. arXiv preprint arXiv:2501.12948 (2025)
9. Gupta, A., Dollar, P., Girshick, R.: Lvis: A dataset for large vocabulary instance segmentation. In: CVPR (2019)
10. Hu, E.J., yelong shen, Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: LoRA: Low-rank adaptation of large language models. In: ICLR (2022)
11. Huang, J., Xu, Z., Zhou, J., Liu, T., Xiao, Y., Ou, M., Ji, B., Li, X., Yuan, K.: SAM-r1: Leveraging SAM for reward feedback in multimodal segmentation via reinforcement learning. In: NeurIPS (2025)
12. Huang, Q., Ko, T., Zhuang, Z., Tang, L., Zhang, Y.: HiRA: Parameter-efficient hadamard high-rank adaptation for large language models. In: ICLR (2025)
13. Huang, S., Qu, X., Li, Y., Luo, Y., He, Z., Liu, D., Cheng, Y.: Spotlight on token perception for multimodal reinforcement learning. In: ICLR (2026)
14. Jang, D., Cho, Y., Lee, S., Kim, T., Kim, D.: MMR: A large-scale benchmark dataset for multi-target and multi-granularity reasoning segmentation. In: ICLR (2025)
15. Jung, M., Lee, S., Kim, E., Yoon, S.: Visual attention never fades: Selective progressive attention recalibration for detailed image captioning in multimodal large language models. In: ICML (2025)
16. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: ICCV (2023)
17. Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Malloci, M., Kolesnikov, A., Duerig, T., Ferrari, V.: The open

images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. IJCV (2020)
18. Lai, X., Tian, Z., Chen, Y., Li, Y., Yuan, Y., Liu, S., Jia, J.: Lisa: Reasoning segmentation via large language model. In: CVPR (2024)
19. Lambert, N., Morrison, J., Pyatkin, V., Huang, S., Ivison, H., Brahman, F., Miranda, L.J.V., Liu, A., Dziri, N., Lyu, S., et al.: Tulu 3: Pushing frontiers in open language model post-training. arXiv preprint arXiv:2411.15124 (2024)
20. Liang, J., Huang, W., Wan, G., Yang, Q., Ye, M.: Lorasculpt: Sculpting lora for harmonizing general and specialized knowledge in multimodal large language models. In: CVPR (2025)
21. Liu, C., Ding, H., Jiang, X.: Gres: Generalized referring expression segmentation. In: CVPR (2023)
22. Liu, H., Li, C., Li, Y., Li, B., Zhang, Y., Shen, S., Lee, Y.J.: Llava-next: Improved reasoning, ocr, and world knowledge (2024), https://llava-vl.github.io/blog/2024-01-30-llava-next/
23. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. In: NeurIPS (2023)
24. Liu, Y., Peng, B., Zhong, Z., Yue, Z., Lu, F., Yu, B., Jia, J.: Seg-zero: Reasoning-chain guided segmentation via cognitive reinforcement. arXiv preprint arXiv:2503.06520 (2025)
25. Liu, Y., Qu, T., Zhong, Z., Peng, B., Liu, S., Yu, B., Jia, J.: Visionreasoner: Unified reasoning-integrated visual perception via reinforcement learning. In: ICLR (2026)
26. Lu, Y., Cao, J., Wu, Y., Li, B., Tang, L., Ji, Y., Wu, C., Wu, J., Zhu, W.: RSVP: Reasoning segmentation via visual prompting and multi-modal chain-of-thought. In: ACL (2025)
27. Lu, Z., Li, L., Wang, J., Feng, Y., Chen, B., Chen, K., Wang, Y.: Coprs: Learning positional prior from chain-of-thought for reasoning segmentation. In: ICLR (2026)
28. Muennighoff, N., Yang, Z., Shi, W., Li, X.L., Fei-Fei, L., Hajishirzi, H., Zettlemoyer, L., Liang, P., Candès, E., Hashimoto, T.B.: s1: Simple test-time scaling. In: EMNLP (2025)
29. Qian, R., Yin, X., Dou, D.: Reasoning to attend: Try to understand how< seg> token works. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 24722–24731 (2025)
30. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., Mintun, E., Pan, J., Alwala, K.V., Carion, N., Wu, C.Y., Girshick, R., Dollar, P., Feichtenhofer, C.: SAM 2: Segment anything in images and videos. In: ICLR (2025)
31. Razin, N., Wang, Z., Strauss, H., Wei, S., Lee, J.D., Arora, S.: What makes a reward model a good teacher? an optimization perspective. In: NeurIPS (2025)
32. Ren, Z., Huang, Z., Wei, Y., Zhao, Y., Fu, D., Feng, J., Jin, X.: Pixellm: Pixel reasoning with large multimodal model. In: CVPR (2024)
33. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347 (2017)
34. Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y., Wu, Y., et al.: Deepseekmath: Pushing the limits of mathematical reasoning in open language models. arXiv preprint arXiv:2402.03300 (2024)
35. Singh, A., Fry, A., Perelman, A., Tart, A., Ganesh, A., El-Kishky, A., McLaughlin, A., Low, A., Ostrow, A., Ananthram, A., et al.: Openai gpt-5 system card. arXiv preprint arXiv:2601.03267 (2025)
36. Snell, C.V., Lee, J., Xu, K., Kumar, A.: Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In: ICLR (2025)

37. Tian, X., Zou, S., Yang, Z., He, M., Waschkowski, F., Wesemann, L., Tu, P.H., Zhang, J.: More thought, less accuracy? on the dual nature of reasoning in vision-language models. In: ICLR (2026)
38. Wang, H., Li, Y., Wang, S., Chen, G., Chen, Y.: Milora: Harnessing minor singular components for parameter-efficient llm finetuning. In: NAACL (2025)
39. Wang, H., Qu, C., Huang, Z., Chu, W., Lin, F., Chen, W.: VL-rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. In: NeurIPS (2025)
40. Wang, P., Bai, S., Tan, S., Wang, S., Fan, Z., Bai, J., Chen, K., Liu, X., Wang, J., Ge, W., et al.: Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. arXiv preprint arXiv:2409.12191 (2024)
41. Wang, S., Asilis, J., Akgül, Ö.F., Bilgin, E.B., Liu, O., Neiswanger, W.: Tina: Tiny reasoning models via loRA. In: ICLR (2026)
42. Wang, X., Zhang, S., Li, S., Li, K., Kallidromitis, K., Kato, Y., Kozuka, K., Darrell, T.: SegLLM: Multi-round reasoning segmentation with large language models. In: ICLR (2025)
43. Wang, X., Wei, J., Schuurmans, D., Le, Q.V., Chi, E.H., Narang, S., Chowdhery, A., Zhou, D.: Self-consistency improves chain of thought reasoning in language models. In: ICLR (2023)
44. Wei, J., Wang, X., Schuurmans, D., Bosma, M., brian ichter, Xia, F., Chi, E.H., Le, Q.V., Zhou, D.: Chain of thought prompting elicits reasoning in large language models. In: NeurIPS (2022)
45. Xu, G., Jin, P., Wu, Z., Li, H., Song, Y., Sun, L., Yuan, L.: Llava-cot: Let vision language models reason step-by-step. In: ICCV (2025)
46. Xu, Y.E., Savani, Y., Fang, F., Kolter, J.Z.: Not all rollouts are useful: Down-sampling rollouts in llm reinforcement learning. arXiv preprint arXiv:2504.13818 (2025)
47. Xu, Z., Liu, C., Wei, Q., Wu, J., Zou, J., Wang, X.E., Zhou, Y., Liu, S.: More thinking, less seeing? assessing amplified hallucination in multimodal reasoning models. In: NeurIPS (2025)
48. Yang, S., Qu, T., Lai, X., Tian, Z., Peng, B., Liu, S., Jia, J.: Lisa++: An improved baseline for reasoning segmentation with large language model. arXiv preprint arXiv:2312.17240 (2023)
49. Yu, L., Poirson, P., Yang, S., Berg, A.C., Berg, T.L.: Modeling context in referring expressions. In: ECCV (2016)
50. Yu, Q., Zhang, Z., Zhu, R., Yuan, Y., Zuo, X., Yue, Y., Dai, W., Fan, T., Liu, G., Liu, L., et al.: Dapo: An open-source llm reinforcement learning system at scale. arXiv preprint arXiv:2503.14476 (2025)
51. Yun, S., Chae, S., Lee, D., Ro, Y.: Soma: Singular value decomposed minor components adaptation for domain generalizable representation learning. In: CVPR (2025)
52. Zhou, E., An, J., Chi, C., Han, Y., Rong, S., Zhang, C., Wang, P., Wang, Z., Huang, T., Sheng, L., Zhang, S.: Roborefer: Towards spatial referring with reasoning in vision-language models for robotics. In: NeurIPS (2026)
53. Zhu, L., Chen, T., Xu, Q., Liu, X., Ji, D., Wu, H., Soh, D.W., Liu, J.: Popen: Preference-based optimization and ensemble for lvlm-based reasoning segmentation. In: CVPR (2025)