CoPRS: Learning Positional Prior from Chain-of-Thought for Reasoning Segmentation

Anonymous authors
Paper under double-blind review

Abstract

Existing works on reasoning segmentation either connect hidden features from a language model directly to a mask decoder or represent positions in text, which limits interpretability and semantic detail. To solve this, we present CoPRS, a Multi-modal Chain-of-Thought (MCoT)–based positional perception model that bridges language reasoning to segmentation through a differentiable and interpretable positional prior instantiated as a heatmap. By making the reasoning process clear via MCoT and expressing it as a dense, differentiable heatmap, this interface enhances interpretability and diagnostic analysis and yields more concentrated evidence on the target. A learnable concentration token aggregates features of the image and reasoning text to generate this positional prior, which is decoded to precise masks through a lightweight decoder, providing a direct connection between reasoning and segmentation. Across the RefCOCO series and ReasonSeg, CoPRS matches or surpasses the best reported metrics on each standard split under comparable protocols, with performance at or above the prior state of the art across both validation and test partitions. Extensive experiments demonstrate a strong positive correlation among the CoT trajectory, the generated heatmap, and the decoded mask, supporting an interpretable alignment between the reasoning output and downstream mask generation. Collectively, these findings support the utility of this paradigm in bridging reasoning and segmentation and show advantages in concentration driven by reasoning and in more precise mask prediction. Code, checkpoints and logs will be released.

1 Introduction

Visual perception is increasingly expected to not only assign labels to pixels but also follow natural-language instructions with compositional constraints, such as "Segment the UAV that is trailing the quadcopter and partially occluded by trees." This demand advances the long arc of visual understanding, starting from semantic segmentation (category labels) (Guo et al., 2018), to instance segmentation (object masks) (Hafiz & Bhat, 2020), and further to open-vocabulary segmentation (open-set text categories) (Ren et al., 2024a), and most recently, toward ***reasoning segmentation*** (free-form instructions) Lai et al. (2024). Meeting this goal requires coupling language reasoning with spatial grounding by converting textual instructions into perceptual decisions.

Existing attempts to bridge language reasoning with segmentation fall into two distinct camps. ***Latent reasoning*** methods (Pi et al., 2024; Lai et al., 2024) predict the masks by directly decoding hidden features from the language models, which keep intermediate decisions non-transparent and uncontrollable. ***Text-based reasoning*** methods (Lan et al., 2025; Liu et al., 2025), on the other hand, readout positions in text and generate discrete coordinates. While explicit, such an interface is inflexible to capture and reflect fine-grained visual semantics, and also fragile to practical issues like formatting errors or out-of-image coordinates. In essence, limitations in the two polarized paradigms highlight the need for a better trade-off between interpretability and representational fidelity.

To close this gap, we introduce **CoPRS**, a **Co**T-based **P**ositional perception model for **R**easoning **S**egmentation. CoPRS is one-stage and end-to-end: given an image–instruction input, it first reasons before producing a perception heatmap concentrating the target region, which provides a ***positional prior*** to enhance the segmentation mask decoding. As compared in Figure 1, the positional prior serves as a differentiable and interpretable connection between MCoT (Wang et al., 2025b) and

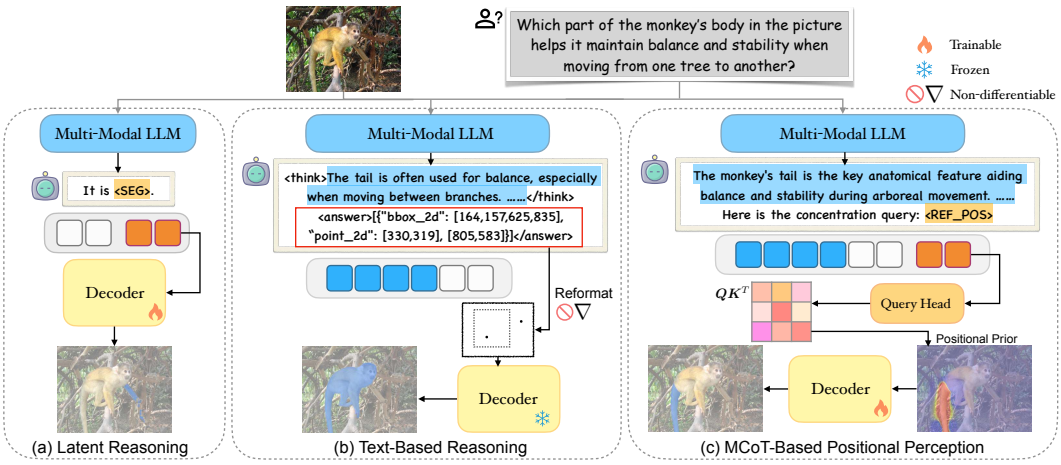


Figure 1: **Illustration of paradigms for reasoning segmentation.** (a) is exemplified by LISA (Lai et al., 2024), and (b) by Seg-Zero (Liu et al., 2025). Our CoPRS (c) bridges MCoT reasoning to segmentation through a differentiable and interpretable positional prior.

segmentation, which is direct and effective to enhance visual perception of a Multi-modal Large Language Model (MLLM) and align instruction semantics with mask decoding.

Specifically, we first introduce a learnable concentration token to aggregate image–instruction context and generate a concentration query. Next, we convert this query to a heatmap used as the positional prior to concentrate the target for mask prediction. This dense, differentiable heatmap is more interpretable than purely hidden features, and provides finer detail than discrete textual coordinates. Concurrently, we establish a unified training framework by adopting the Group Relative Policy Optimization (GRPO) (Shao et al., 2024) strategy jointly with segmentation supervision. This framework enhances reasoning capability through GRPO, jointly supervising the MLLM and segmentation model via a differentiable positional prior and offering an effective solution to the limitations of prior paradigms.

CoPRS matches or exceeds the best reported cIoU/gIoU on each split under comparable protocols across RefCOCO, RefCOCO+ (Kazemzadeh et al., 2014), RefCOCOg (Mao et al., 2016), and ReasonSeg (Lai et al., 2024). We further find a strong positive correlation among the quality of the CoT trajectory, the generated heatmap, and the decoded mask, indicating strong concentration driven by reasoning and precise mask generation. Beyond reasoning segmentation, the unified framework and its positional prior naturally extend to region concentration tasks such as referring tracking and trajectory prediction.

To summarize, we make the following contributions in this paper.

- **CoPRS Formulation.** We present an end-to-end MCoT-driven positional perception model for reasoning segmentation, where a language-conditioned positional prior serves as an interpretable intermediate aligning instruction understanding with mask prediction.
- **Unified Framework.** We establish a unified training framework by combining a GRPO strategy with a supervised objective, enhancing reasoning and segmentation in a single loop and overcoming the limitations of prior paradigms.
- **Positional Prior Interface.** A learnable concentration query produces a heatmap as a dense positional prior, and a lightweight decoder refines it into a precise mask. Our design provides both interpretable concentration and strong boundary quality.
- **Strong Results.** CoPRS performs strongly on each split across the RefCOCO series and ReasonSeg, and further analysis clarifies how reasoning output aligns with segmentation performance.

2 Related Work

Referring and Reasoning Segmentation. Referring segmentation requires a model to produce a mask for the entity described in a short instruction. Prior methods such as VLT (Ding et al., 2021),

CRIS (Wang et al., 2022), LAVT (Yang et al., 2022), ReLA (Liu et al., 2023a), X-Decoder (Zou et al., 2023a), SEEM (Zou et al., 2023b), Grounded-SAM (Ren et al., 2024a), typically rely on specific text encoders rather than large language models (LLMs) to parse the text and predict the mask. Reasoning segmentation extends this setting to longer, compositional instructions with stricter grounding requirements, motivating the two method families outlined next.

Latent Reasoning Methods. Advances in multimodal large language models (MLLMs) (Liu et al., 2023b; Bai et al., 2023) have substantially improved the reasoning capability of vision–language perception. LISA (Lai et al., 2024) bridges the gap between MLLMs and reasoning segmentation by introducing a special token. Subsequent works, including PerceptionGPT (Pi et al., 2024), PixelLM (Ren et al., 2024b), SegLLM (Ren et al., 2024b), LaSagnA (Wei et al., 2024), OMG-LLaVA (Zhang et al., 2024a), GroundHog (Zhang et al., 2024b), GLaMM (Rasheed et al., 2024), RAS (Cao et al., 2025), leverage LLM latent features and decode them into segmentation masks. However, they neither reveal intermediate reasoning before the final prediction nor expose it through a transparent interface. In contrast, our approach makes the reasoning process clear via MCoT and visualizes the intermediate as a heatmap, improving interpretability and diagnostic analysis.

Text-based Reasoning Methods. Since SAM (Kirillov et al., 2023) achieves strong segmentation quality when prompted with boxes or points, it is feasible to prompt SAM using textual coordinates after a simple format conversion. Recent works, such as SAM4MLLM (Chen et al., 2024), Seg-Zero (Liu et al., 2025) and Seg-R1 (You & Wu, 2025), use MLLMs to generate textual coordinates of boxes and points via chain-of-thought, and then feed them to SAM for mask prediction. In a similar vein, Text4Seg (Lan et al., 2025) generates textual patch indices and applies CRF (Krähenbühl & Koltun, 2011) or SAM for mask refinement. Such sparse, discrete outputs provide limited semantic detail and are sensitive to formatting errors and out-of-image coordinates. To address these issues, our model introduces a dense, differentiable positional prior that captures richer semantic detail.

Additional related work on GRPO and multimodal chain-of-thought are introduced in Section A.2.

3 METHOD

We first present the model design and data flow in Section 3.1. We then formalize the learning objectives, unifying policy optimization via GRPO on the language path with segmentation supervision on the vision path in Section 3.2. Finally, we detail the training and inference procedures in Section 3.3, including data preparation, tokenization, group rollouts, and deterministic inference.

3.1 MODEL ARCHITECTURE

Overall Architecture. As shown in Figure 2, CoPRS is built upon a multimodal LLM (MLLM), a vision backbone, a query head and a mask decoder. Given image and text inputs $(\boldsymbol{x}_{\text{img}}, \boldsymbol{x}_{\text{txt}})$, a policy model $\pi_{\boldsymbol{\theta}}(\cdot)$ generates a token sequence that includes the chain-of-thought (CoT) and a concentration token, and we read the MLLM's hidden states to obtain the concentration token embedding. Then the query head $\mathcal{F}_{\text{head}}(\cdot)$ maps this embedding to a concentration query. The vision encoder $\mathcal{F}_{\text{enc}}(\cdot)$ extracts image features as image keys. Subsequently, the query attends to the image keys with multi-head attention, yielding a heatmap that serves as a positional prior. Finally, the mask decoder $\mathcal{F}_{\text{dec}}(\cdot)$ decodes this prior to the predicted mask $\hat{\boldsymbol{M}}$.

MLLM Backbone. We use Qwen2.5-VL (Bai et al., 2025) as our MLLM backbone. Following DeepSeek-R1 (Guo et al., 2025), we adopt multimodal chain-of-thought (MCoT) to leverage the reasoning capabilities of MLLM on compositional instructions. Specifically, we use an instruction prompt to elicit both the CoT and a concentration token: given $(\boldsymbol{x}_{\text{img}}, \boldsymbol{x}_{\text{txt}})$, the model is asked to (i) reason in a `<think>...</think>` block and then (ii) output the concentration token `<REF_POS>`. We obtain the concentration token's embedding $\boldsymbol{e}_{\text{conc}}$ via $\mathcal{F}_{\text{conc}}$ which finds its occurrence and reads the hidden states of LLM. Under this setup, the policy $\pi_{\boldsymbol{\theta}}$ generates the token sequence $\boldsymbol{y}_{1:T}$ via next token prediction. Formally, the process is given in

$$\begin{aligned} y_t &\sim \pi_{\boldsymbol{\theta}}(\cdot \mid \boldsymbol{y}_{0:t-1}, \boldsymbol{x}_{\text{img}}, \boldsymbol{x}_{\text{txt}}),\ t = 1, \ldots, T, \\ \boldsymbol{e}_{\text{conc}} &= \mathcal{F}_{\text{conc}}(\boldsymbol{y}_{1:T}), \end{aligned} \tag{1}$$

where $\boldsymbol{y}_{1:T}$ includes both the CoT and the concentration token.

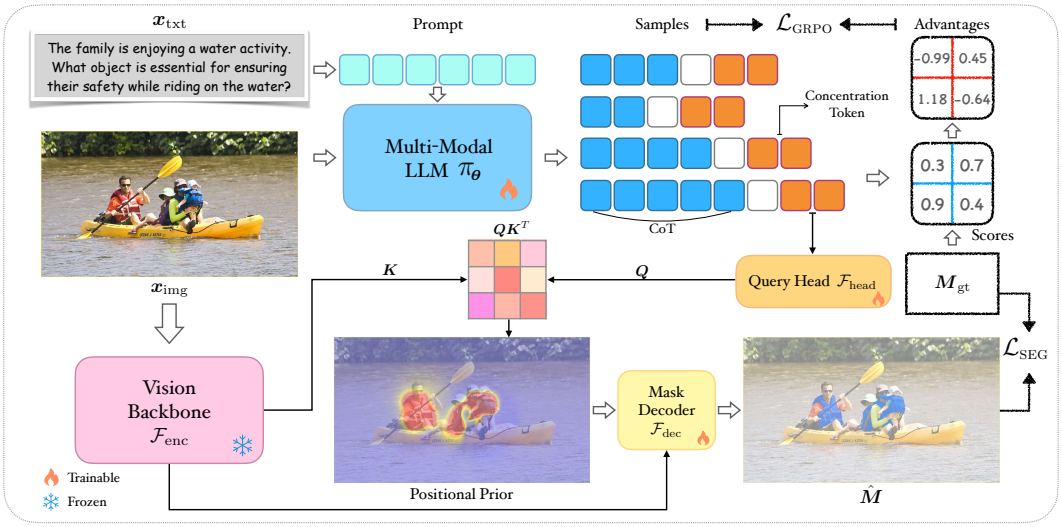


Figure 2: **Overall architecture.** Given image and text inputs, the policy generates CoT and concentration tokens, which query image keys to generate a positional prior, that is then decoded to masks. The policy and segmentation modules are jointly optimized.

From Keys and a Query to Positional Prior. The vision backbone encodes $\boldsymbol{x}_{\text{img}}$ into image features, which we map to vision keys $\boldsymbol{K}$ via a multilayer perceptron (MLP) applied to the backbone output. In practice, we choose ViT-H — an image encoder from SAM (Kirillov et al., 2023) as the vision backbone and an MLP query head projects $\boldsymbol{e}_{\text{conc}}$ into the concentration query $\boldsymbol{Q}$. Subsequently, we compute scaled dot product multi-head attention scores (Vaswani et al., 2017) between $\boldsymbol{Q}$ and $\boldsymbol{K}$, and we use two stacked 2D convolutional layers denoted $\mathcal{F}_{\text{fuse}}(\cdot)$ to aggregate features across heads. Formally, the computation is defined in the following equations.

$$\begin{aligned} \boldsymbol{K} &= \mathcal{F}_{\text{enc}}(\boldsymbol{x}_{\text{img}}), \quad \boldsymbol{Q} = \mathcal{F}_{\text{head}}(\boldsymbol{e}_{\text{conc}}), \\ \boldsymbol{H}_{\text{prior}} &= \mathcal{F}_{\text{fuse}}\Big(\big[(\boldsymbol{Q}\boldsymbol{W}_i^Q)(\boldsymbol{K}\boldsymbol{W}_i^K)^\top / \sqrt{d_c} \big]_{i=1}^{n_{\text{head}}} \Big), \end{aligned} \tag{2}$$

where $\boldsymbol{Q} \in \mathbb{R}^{d_q}$, $\boldsymbol{K} \in \mathbb{R}^{H \times W \times d_k}$, $\boldsymbol{W}_i^Q \in \mathbb{R}^{d_q \times d_h}$, $\boldsymbol{W}_i^K \in \mathbb{R}^{d_k \times d_h}$, d_h is the head dimension, n_{head} is the number of heads, and $\mathcal{F}_{\text{fuse}} : \mathbb{R}^{n_{\text{head}} \times H \times W} \to \mathbb{R}^{H \times W}$. Details are provided in Algorithm 1.

Lightweight Decoder. Our mask decoder comprises two submodules. First, three stacked 2D convolutional blocks resample the fused positional prior, producing a feature map at the decoder resolution. Second, we choose a Two-Way Transformer following the SAM decoder design (Kirillov et al., 2023), which performs bidirectional cross attention between the image features and the positional prior. This lightweight design has 4.7M parameters and enables the prior to guide dense segmentation. Formally, we formulate the process as

$$\hat{\boldsymbol{M}} = \mathcal{F}_{\text{dec}}(\boldsymbol{K}, \boldsymbol{H}_{\text{prior}}). \tag{3}$$

3.2 Learning Objectives

Unified Objective. We train the whole system end-to-end with a single objective that couples reinforcement learning on the language path with segmentation supervision on the vision path. For each $(\boldsymbol{x}_{\text{img}}, \boldsymbol{x}_{\text{txt}})$, the policy π_θ rolls out a group of responses $\{\boldsymbol{y}_{1:T_i}^{(i)}\}_{i=1}^{G}$ with the group size G, and we compute a GRPO loss $\mathcal{L}_{\text{GRPO}}$ from the advantages. In parallel, the positional prior $\boldsymbol{H}_{\text{prior}}$ and the predicted mask $\hat{\boldsymbol{M}}$ are supervised against the ground truth mask $\boldsymbol{M}_{\text{gt}}$ to yield the segmentation loss $\mathcal{L}_{\text{SEG}}$. The overall objective is

$$\mathcal{L} = \mathcal{L}_{\text{GRPO}}\Big(\{\boldsymbol{y}_{1:T_i}^{(i)}\}_{i=1}^{G}\Big) + \lambda_{\text{SEG}}\mathcal{L}_{\text{SEG}}\Big(\boldsymbol{H}_{\text{prior}}, \hat{\boldsymbol{M}}, \boldsymbol{M}_{\text{gt}}\Big). \tag{4}$$

We compute both terms for each batch and take a single backward pass through all trainable modules.

GRPO Objective. Following Shao et al. (2024), we optimize $\pi_{\boldsymbol{\theta}}$ with the GRPO objective. The update ratio $r_{i,t}$ is the likelihood ratio between the current policy $\pi_{\boldsymbol{\theta}}$ and the old policy $\pi_{\boldsymbol{\theta}_{\text{old}}}$ at token $o_{i,t}$, which is clipped with ε introduced in PPO (Schulman et al., 2017) for stability. The advantage $\widehat{A}_{i,t}$ is computed relative rewards within each group only; details are given in Section A.1. Formally, the policy loss is

$$\mathcal{L}_{\pi} \;=\; \mathbb{E}_{i,t}\Big[\min\Big(r_{i,t}\,\widehat{A}_{i,t},\; \text{clip}\big(r_{i,t},\,1-\varepsilon,\,1+\varepsilon\big)\,\widehat{A}_{i,t}\Big)\Big], \quad t=1:T_i,\; i=1:G, \tag{5}$$

where the update ratio

$$r_{i,t} = \frac{\pi_{\boldsymbol{\theta}}\big(o_{i,t} \,\big|\, \boldsymbol{o}_{i,1:t-1},\, \boldsymbol{x}_{\text{img}},\, \boldsymbol{x}_{\text{txt}}\big)}{\pi_{\boldsymbol{\theta}_{\text{old}}}\big(o_{i,t} \,\big|\, \boldsymbol{o}_{i,1:t-1},\, \boldsymbol{x}_{\text{img}},\, \boldsymbol{x}_{\text{txt}}\big)}, \tag{6}$$

and the token $o_{i,t} = y_t^{(i)}$. GRPO further regularizes with a KL divergence term between the trained policy and the reference policy:

$$\mathcal{L}_{\text{GRPO}} = \mathcal{L}_{\pi} - \beta\, \mathbb{D}_{\text{KL}}[\,\pi_{\boldsymbol{\theta}}\,||\,\pi_{\text{ref}}\,], \tag{7}$$

where β is the coefficient of the KL penalty (See Section A.1).

For each sampled response in the group, we design a reward function that combines mask quality and CoT format compliance. Specifically, the mask reward score aggregates soft IoU, soft dice, and hard IoU, while the CoT format reward score is computed via multiple regular expressions for the string matching. We then normalize both rewards to the range $[0, 1]$ using fixed coefficients. Further implementation details are provided in Section 4.1.

Supervised Segmentation Objective. The segmentation loss comprises three complementary terms. (i) A binary cross-entropy (BCE) loss applied to $\boldsymbol{H}_{\text{prior}}$ encourages positional evidence and accurate concentration. (ii) A dice loss (Milletari et al., 2016) on the predicted mask $\hat{\boldsymbol{M}}$ directly supervises mask quality. (iii) A focal loss (Lin et al., 2017) on the mask logits emphasizes hard pixels and fine-grained structures. All losses are computed only over the original image region and averaged per image over the batch, with the dice loss coefficient λ_d and focal loss coefficient λ_f being reported in Section 4.1. Formally, the segmentation loss is

$$\mathcal{L}_{\text{SEG}} \;=\; \mathcal{L}_{\text{BCE}}\big(\boldsymbol{H}_{\text{prior}},\, \boldsymbol{M}_{\text{gt}}\big) + \lambda_d \mathcal{L}_{\text{DICE}}\big(\hat{\boldsymbol{M}},\, \boldsymbol{M}_{\text{gt}}\big) + \lambda_f \mathcal{L}_{\text{FOCAL}}\big(\hat{\boldsymbol{M}},\, \boldsymbol{M}_{\text{gt}}\big). \tag{8}$$

3.3 Training and Inference

Data Preparation. Before entering the $\mathcal{F}_{\text{enc}}$, we resize each image so that its longer side is 1024 pixels while preserving aspect ratio, then we pad it to 1024×1024. We apply the same transforms to the masks to maintain coordinate alignment during loss computation. For the policy $\pi_{\boldsymbol{\theta}}$, we cap the input at 705,600 pixels (900 vision tokens). If an image exceeds this cap, we downsample it while preserving aspect ratio for the policy input.

Training Procedure. As shown in Figure 2, during training we tokenize $(\boldsymbol{x}_{\text{img}},\, \boldsymbol{x}_{\text{txt}})$, replicate each pair for G times, and feed these copies to the $\pi_{\boldsymbol{\theta}}$ to generate G responses. For each response in the group, the reward function assigns a scalar score, and the scores are converted into advantages for computing $\mathcal{L}_{\text{GRPO}}$, which updates only the MLLM parameters. In the same batch, $\boldsymbol{x}_{\text{img}}$ is resized and padded, then encoded by the vision backbone, and decoded to $\hat{\boldsymbol{M}}$ for computing $\mathcal{L}_{\text{SEG}}$, which updates all trainable modules. We optimize both losses jointly in each iteration.

Inference Procedure. At inference, $(\boldsymbol{x}_{\text{img}}, \boldsymbol{x}_{\text{txt}})$ is used without replication. $\pi_{\boldsymbol{\theta}}$ runs with deterministic next token prediction to produce a single response that includes the concentration token. We then apply the same forward path as in training to produce mask logits. Finally, we remove padding, resize to the original image size, and threshold the logits at zero to obtain the binary mask.

4 Experiments

Research Questions. In this section, we aim to answer the following research questions:

RQ1: Does CoPRS achieve higher accuracy in reasoning segmentation and state-of-the-art results on standard benchmarks compared to prior methods?

Table 1: Comparison of methods on RefCOCO, RefCOCO+, and RefCOCOg datasets.

Model Type	Method	RefCOCO			RefCOCO+			RefCOCOg	
		val	testA	testB	val	testA	testB	val	test
Methods without LLMs	VLT	67.5	70.5	65.2	56.3	61.0	50.1	55.0	57.7
	CRIS	70.5	73.2	66.1	62.3	68.1	53.7	59.9	60.4
	LAVT	72.7	75.8	68.8	62.1	68.4	55.1	61.2	62.1
	ReLA	73.8	76.5	70.2	66.0	71.0	57.7	65.0	66.0
	X-Decoder	–	–	–	–	–	–	64.6	–
	SEEM	–	–	–	–	–	–	65.7	–
Latent Reasoning	LISA-7B	74.9	79.1	72.3	65.1	70.8	58.1	67.9	70.6
	LISA-13B	76.0	78.8	72.9	65.0	70.2	58.1	69.5	70.5
	PerceptionGPT-7B	75.1	78.6	71.7	68.5	73.9	61.3	70.3	71.7
	PerceptionGPT-13B	75.3	79.1	72.1	68.9	74.0	61.9	70.7	71.9
	PixelLM-7B	73.0	76.5	68.2	66.3	71.7	58.3	69.3	70.5
	LaSagnA-7B	76.8	78.7	73.8	66.4	70.6	60.1	70.6	71.9
	SegLLM-7B	80.2	81.5	75.4	70.3	73.0	62.5	72.6	73.6
	OMG-LLaVA-7B	78.0	80.3	74.1	69.1	73.1	63.0	72.9	72.9
	GroundHog-7B	78.5	79.9	75.7	70.5	75.0	64.9	74.1	74.6
	GLaMM-7B	79.5	83.2	76.9	72.6	78.7	64.6	74.2	74.9
	RAS-13B	81.0	*83.5*	79.0	75.1	80.0	**70.3**	76.0	**77.5**
Text-based Reasoning	SAM4MLLM-7B	79.6	82.8	76.1	73.5	77.8	65.8	74.5	75.6
	Seg-R1-3B	69.9	76.0	64.9	59.1	66.8	50.9	67.9	67.3
	Seg-R1-7B	74.3	78.7	67.6	62.6	70.9	57.9	71.0	71.4
	Seg-Zero-3B	–	79.3	–	–	73.7	–	–	71.5
	Seg-Zero-7B	–	80.3	–	–	76.2	–	–	72.6
	Text4Seg-7B	79.3	81.9	76.2	72.1	77.6	66.1	72.1	73.9
	Text4Seg-13B	80.2	82.7	*77.3*	*73.7*	78.6	*67.6*	74.0	*75.1*
Positional Prior	CoPRS-3B	*80.4*	83.9	75.6	71.8	*78.9*	66.5	*74.8*	73.7
	CoPRS-7B	**81.6**	**85.3**	**79.5**	**75.9**	**80.3**	69.7	**76.2**	76.2

RQ2: How are the CoT, the positional prior $\boldsymbol{H}_{\text{prior}}$, and the predicted mask $\hat{\boldsymbol{M}}$ mutually correlated, i.e., does higher CoT quality align with stronger positional priors and better segmentation accuracy?

RQ3: Do the GRPO settings, supervised segmentation losses, and MLLM/vision backbone choices each contribute to performance, and does our unified objective with the default backbones outperform these alternatives?

4.1 Experimental Setup

Datasets and Metrics. We evaluate CoPRS by conducting experiments on four datasets. We train CoPRS-3B and CoPRS-7B separately on the training sets of RefCOCO, RefCOCO+ and RefCOCOg. To prevent data leakage, we remove from the training data all COCO images that appear in the validation or test splits of RefCOCO(+/g). We evaluate on the official validation and test splits of RefCOCO(+/g). We further assess zero-shot reasoning segmentation by evaluating on ReasonSeg (validation and test) without training on its images. Consistent with common practice in prior work (e.g., Lai et al. (2024)), we adopt intersection over union (IoU) metrics. Specifically, we report cIoU (the cumulative intersection over the cumulative union) on RefCOCO(+/g), and both cIoU and gIoU (mean of per-image IoU) on ReasonSeg.

Baselines. We compare our method with 20 prior works grouped into three categories. Methods without LLMs, including VLT (Ding et al., 2021), CRIS (Wang et al., 2022), LAVT (Yang et al., 2022), ReLA (Liu et al., 2023a), X-Decoder (Zou et al., 2023a), SEEM (Zou et al., 2023b), Grounded-SAM (Ren et al., 2024a), do not rely on LLM to encode instruction texts for generating masks. Latent reasoning methods, including LISA (Lai et al., 2024), PerceptionGPT (Pi et al., 2024), PixelLM (Ren et al., 2024b), LaSagnA (Wei et al., 2024), SegLLM (Wang et al., 2025a), OMG-LLaVA (Zhang et al., 2024a), GroundHog (Zhang et al., 2024b), GLaMM (Rasheed et al., 2024), RAS (Cao et al., 2025), take hidden features from a large language model and decode them into segmentation masks. Text-based reasoning methods, including SAM4MLLM (Chen et al., 2024), Seg-Zero (Liu et al., 2025), Seg-R1 (You & Wu, 2025), Text4Seg (Lan et al., 2025), use an MLLM to emit discrete location tokens—box/point coordinates or patch indices, and then convert them to masks. For approaches available in multiple parameter scales, we report results for all the variants. RAS provides only a version with 13B parameters.

Implementation Details. We train on 8 NVIDIA A100 (80 GB) GPUs. Our implementation builds on the VERL codebase. Concretely, we weight the two components of reward function as 0.7 for mask and 0.3 for CoT format. Within the mask score, the coefficients for soft IoU, soft Dice, and hard IoU are set to 0.5, 0.2, and 0.3, respectively, and the format score is computed under specific regular expression rules for five conditions (see Section B.1). For GRPO, we use sampling numbers of 2, 4, and 8. Loss coefficients λ_{SEG}, λ_d and λ_f are set to 0.3, 3.0 and 10, respectively, for most batches. The base learning rate for the MLLM backbone is set to 2e-6; we apply multipliers of $25\times$ for the concentration query head, and $10\times/5\times$ for two submodules of mask decoder. We use the AdamW (Loshchilov & Hutter, 2019) optimizer with weight decay 0.01. We adopt OneCycleLR (Smith & Topin, 2019) as the learning rate scheduler, applying cosine decay to each parameter group down to one tenth of its peak learning rate. Full configurations are provided in Section B.3.

4.2 Overall Performance (RQ1)

We compare CoPRS with prior state-of-the-art reasoning segmentation methods on two standard benchmarks: the RefCOCO series and ReasonSeg.

Results on RefCOCO(+/g). We follow standard evaluation protocols (Lai et al., 2024) and evaluate on the RefCOCO series. At matched model sizes, CoPRS-3B and CoPRS-7B achieve the best performance across all RefCOCO, RefCOCO+, and RefCOCOg splits (Table 1). Specifically, CoPRS-7B outperforms the latest reasoning methods on all the splits, trailing RAS-13B on only 2 of 8 splits. This advantage stems from our learning objectives, strengthening the CoT reasoning capability of CoPRS, which is crucial in reasoning segmentation.

Moreover, compared to Seg-R1 and Seg-Zero trained via GRPO, CoPRS achieves significant improvements at both model scales, with the 3B model surpassing their 7B counterparts. This fully demonstrates the effectiveness of our designed learnable concentration query in connecting reasoning and segmentation.

Results on ReasonSeg. We evaluate on ReasonSeg in a zero-shot setting to validate the generalization ability of CoPRS on complex reasoning segmentation scenarios. From Table 2, our CoPRS also demonstrates superior results on the complex reasoning segmentation task. Meanwhile, we find that methods trained with reinforcement learning, such as Seg-R1, Seg-Zero and our CoPRS, consistently outperform other methods, demonstrating the generalization benefits of reinforcement learning for segmentation models.

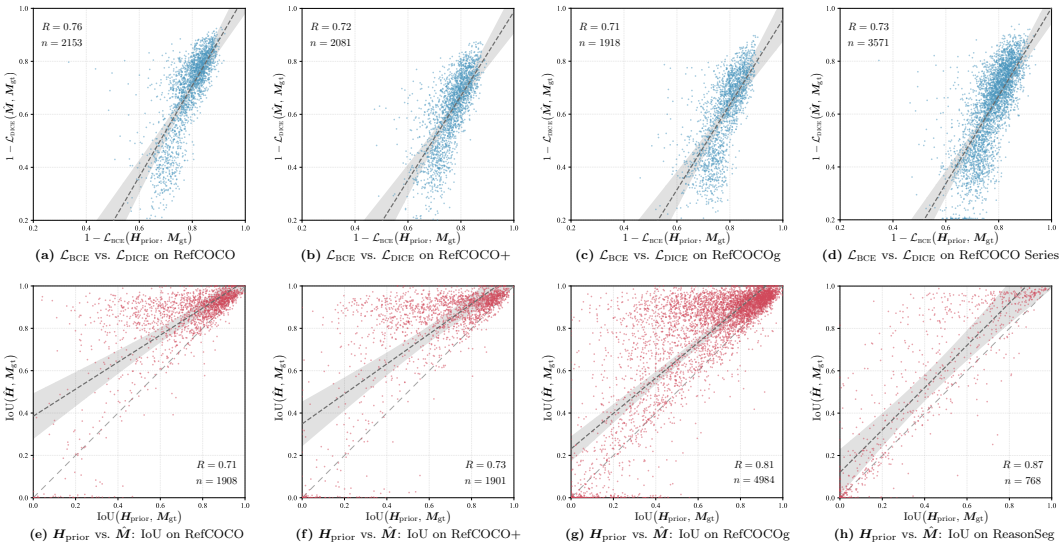
Table 2: Zero-shot comparison of methods on ReasonSeg dataset.

Model Type	Method	val		test	
		gIoU	cIoU	gIoU	cIoU
Methods without LLMs	ReLA	22.4	19.9	21.3	22.0
	X-Decoder	22.6	17.9	21.7	16.3
	SEEM	25.5	21.2	24.3	18.7
	Grounded-SAM	26.0	14.5	21.3	16.4
Latent Reasoning	LISA-7B	53.6	52.3	48.7	48.8
	LISA-13B	57.7	60.3	53.8	50.8
	LaSagnA-7B	–	47.2	–	–
	SegLLM-7B	57.2	54.3	52.4	48.4
	GroundHog-7B	56.2	–	–	–
Text-based Reasoning	SAM4MLLM-7B	46.7	48.1	–	–
	Seg-R1-3B	60.8	56.2	55.3	46.6
	Seg-R1-7B	58.6	41.2	56.7	53.7
	Seg-Zero-3B	58.2	53.1	56.1	48.6
	Seg-Zero-7B	62.6	62.0	*57.5*	52.0
Positional Prior	CoPRS-3B	*61.3*	*60.6*	57.8	*52.7*
	CoPRS-7B	**65.2**	**64.5**	**59.8**	**55.1**

4.3 Correlation Analysis and Visualization (RQ2)

Correlation Analysis Methodology. We first analyze the correlation between the positional prior $\boldsymbol{H}_{\text{prior}}$ and the predicted mask $\hat{\boldsymbol{M}}$ during both training and inference. We then analyze how the quality of CoT correlates with both $\boldsymbol{H}_{\text{prior}}$ and $\hat{\boldsymbol{M}}$, thereby linking the linguistic reasoning to the visual outputs. We plot the corresponding training losses and evaluation metrics as scatter points to make the relationship clear. Additionally, we use ordinary least square regression to plot the regression line $y = \hat{\alpha} + \hat{\beta}x$ and the mean confidence bands $\hat{y}(x) \pm \eta\,\text{s.e.}(\hat{y}(x))$, where $\text{s.e.}(\hat{y}) = \hat{\sigma}\sqrt{\frac{1}{n} + \frac{(x-\bar{x})^2}{\sum_i (x_i-\bar{x})^2}}$ with $\eta = 10$ for visual clarity and $\hat{\sigma}$ being residual standard error.

Correlation between Heatmap and Mask. During training, panels (a)–(d) in Figure 3 show blue points, each representing one training batch. The x-axis is $1 - \mathcal{L}_{\text{BCE}}(\boldsymbol{H}_{\text{prior}}, \boldsymbol{M}_{\text{gt}})$, which increases as the prior better matches $\boldsymbol{M}_{\text{gt}}$. The y-axis is $1 - \mathcal{L}_{\text{DICE}}(\hat{\boldsymbol{M}}, \boldsymbol{M}_{\text{gt}})$, which is higher when $\hat{\boldsymbol{M}}$ converges to $\boldsymbol{M}_{\text{gt}}$. The points exhibit low dispersion, reflecting stable loss with batch size of 128.



(a) $\mathcal{L}_{\text{BCE}}$ vs. $\mathcal{L}_{\text{DICE}}$ on RefCOCO (b) $\mathcal{L}_{\text{BCE}}$ vs. $\mathcal{L}_{\text{DICE}}$ on RefCOCO+ (c) $\mathcal{L}_{\text{BCE}}$ vs. $\mathcal{L}_{\text{DICE}}$ on RefCOCOg (d) $\mathcal{L}_{\text{BCE}}$ vs. $\mathcal{L}_{\text{DICE}}$ on RefCOCO Series

(e) $\boldsymbol{H}_{\text{prior}}$ vs. $\hat{\boldsymbol{M}}$: IoU on RefCOCO (f) $\boldsymbol{H}_{\text{prior}}$ vs. $\hat{\boldsymbol{M}}$: IoU on RefCOCO+ (g) $\boldsymbol{H}_{\text{prior}}$ vs. $\hat{\boldsymbol{M}}$: IoU on RefCOCOg (h) $\boldsymbol{H}_{\text{prior}}$ vs. $\hat{\boldsymbol{M}}$: IoU on ReasonSeg

Figure 3: **Correlation analysis** between the positional prior $\boldsymbol{H}_{\text{prior}}$ and the predicted mask $\hat{\boldsymbol{M}}$ during training and inference on RefCOCO(+/g) and ReasonSeg. Each blue point represents one training batch, while each red point represents one inference instance. Ordinary least squares (OLS) regression lines and mean confidence bands are overlaid.

Across all datasets, the scatter patterns and correlation coefficients $R > 0.7$ indicate a strong positive association between $\boldsymbol{H}_{\text{prior}}$ and $\hat{\boldsymbol{M}}$.

During inference, panels (e)–(h) in Figure 3 show red points, each representing one inference instance. The x-axis is IoU between $\boldsymbol{H}_{\text{prior}}$ and $\boldsymbol{M}_{\text{gt}}$, i.e., the mask quality if the prior were used directly with no decoding. The y-axis is the IoU between $\hat{\boldsymbol{M}}$ and $\boldsymbol{M}_{\text{gt}}$, a standard segmentation metric. As in training, the scatter pattern and correlations $R > 0.7$ reveal a strong positive relationship across test splits. It is observed that the regression lines, confidence bands and most points lie above $y = x$. This trend indicates that the positional prior already concentrates well, while the decoder further refines it to a precise mask.

Correlation between CoT and Segmentation Quality. While Figure 3 already confirms the alignment between the heatmap and the final masks, it does not yet quantify how well the CoT reasoning itself aligns with these visual outputs. To make this link more explicit, we additionally use Gemini-2.5-Flash (Comanici et al., 2025) as an independent automatic evaluator. Inspired by Yin et al. (2025), we compute a consistency score in $[0, 1]$ (weighted average over four dimensions: logical correctness 0.3, task relevance 0.2, visual consistency 0.3, localization accuracy 0.2) between the image–instruction pair and the generated CoT on the RefCOCO+ testA split. The scatter plots in Figure 4 show a clear positive correlation between CoT consistency scores and both Heatmap IoU and Mask IoU. Moreover, Table 3 groups samples by consistency score range and reports the number of samples and heatmap/mask mIoU in each range. Higher consistency bins consistently achieve higher segmentation quality. This quantitative evidence directly supports that better CoT reasoning quality leads to better segmentation performance in CoPRS.

Table 3: **CoT consistency.** Consistency score ranges with sample counts, mean heatmap IoU, and mean mask IoU on RefCOCO+.

Consistency Score	#Samples	Heatmap mIoU	Mask mIoU
$[0, 0.25)$	225	0.25	0.55
$[0.25, 0.5)$	568	0.51	0.78
$[0.5, 0.75)$	749	0.69	0.90
$[0.75, 1.0]$	359	0.82	0.94

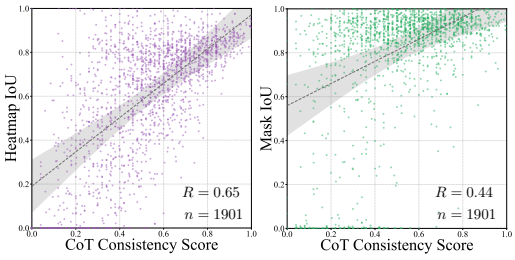


Figure 4: **Correlation** between CoT quality and segmentation quality (Heatmap/Mask IoU) on RefCOCO+. OLS results are overlaid.

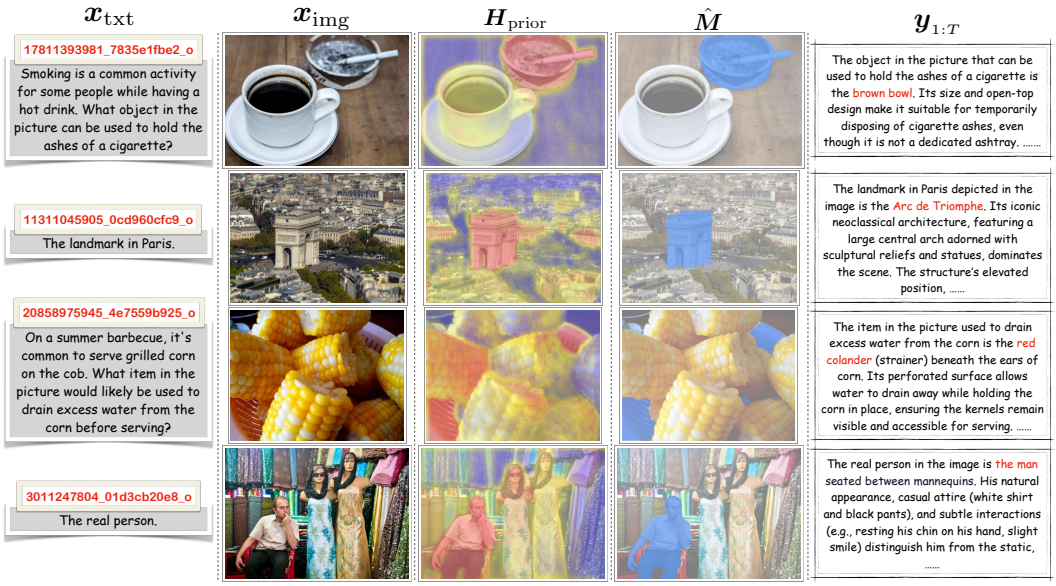


Figure 5: **Sample visualizations.** With sample ID exposed, all samples are from the ReasonSeg test split. From left to right: image-text pair, positional prior, predicted mask, and chain of thought.

Visualization Results. We present zero-shot visualizations on ReasonSeg, as shown in Figure 5. After MCoT reasoning, the positional prior indicates all instances relevant to the instruction (yellow), with the target instance most strongly concentrated (deep red). Figure 8 in Appendix presents additional visualizations. Additional failure cases in Figure 7 in the Appendix show that CoPRS mainly struggles with very small objects that disappear at our current input resolution, and dense groups of similar instances where text alone cannot reliably disambiguate the target.

4.4 Ablation Study (RQ3)

To gain a deeper understanding of the contributing factors, we perform ablation studies on RefCOCO+ with different MLLM backbones and varied vision backbones, and further ablations of CoPRS-7B on RefCOCO+, RefCOCOg, and ReasonSeg. We systematically examine MLLM backbone choice, vision backbone choice, GRPO group size, training mode, reward coefficients, and segmentation loss combinations.

MLLM Backbone. For ablating the MLLM backbone, we additionally train CoPRS with LLaVA-1.5-7B/13B on RefCOCO+. Table 4 reports cIoU metrics of CoPRS versions with both LLaVA-1.5 and Qwen2.5-VL series. As expected, performance increases with backbone capacity, but the gains across different MLLM backbones are relatively modest. This indicates that CoPRS is not sensitive to the specific MLLM architecture and that our improvements largely transfer across different backbone choices. Together with the comparisons to prior work under the same LLaVA-1.5 backbone (Table 1), this suggests that our gains are complementary to backbone strength rather than being tied to a particular MLLM.

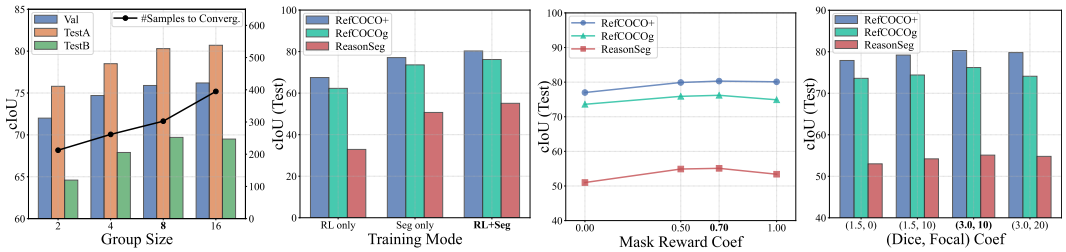
Table 4: Effect of MLLM Backbone Choice. **Gray row** denotes the default backbone.

Method	Backbone	val	testA	testB
CoPRS-3B	Qwen2.5-VL	71.8	78.9	66.5
CoPRS-7B	Qwen2.5-VL	**75.9**	**80.3**	69.7
CoPRS-7B	LLaVA-1.5	73.1	79.0	66.4
CoPRS-13B	LLaVA-1.5	75.5	**80.3**	**70.7**

Vision Backbone. As shown in Table 5, we ablate SAM backbones (ViT-B/L/H) on RefCOCO+ with a fixed Qwen2.5-VL-7B MLLM and report the total parameters of the full pipeline. Larger vision backbones bring slightly better segmentation performance, but the improvement is modest and the overall trend remains stable across sizes.

Table 5: Effect of Vision Backbone Choice. **Gray row** denotes the default backbone.

Backbone	#Params(B)	val	testA	testB
ViT-B	8.38	73.2	77.3	67.0
ViT-L	8.60	74.8	78.9	68.5
ViT-H	8.93	**75.9**	**80.3**	**69.7**



(a) Effect of GRPO group size (b) Effect of training mode choice (c) Effect of mask reward coefficient. (d) Effect of segmentation loss coefficients.

Figure 6: **Ablation studies** on GRPO group size, training mode, mask reward coefficient, and segmentation loss coefficients. (a) is evaluated on all splits of RefCOCO+, while (b)–(d) are evaluated on the test split of each dataset. **Bold x-axis labels** mark the default settings.

Additionally, vision backbones constitute only a small portion of the total parameters, so scaling them up only marginally increases overall computational cost.

GRPO Group Size. We study the effects of GRPO group size during training. The group size G denotes the number of responses sampled per question during rollout. As shown in Figure 6a, increasing G improves performance across splits of RefCOCO+. To quantify efficiency, we also report the total number of GRPO samples required to reach convergence (loss fluctuation $< 10\%$ over 300 steps) for $G \in \{2, 4, 8, 16\}$. Particularly, the number of samples for convergence does not grow linearly with G, because larger groups offer more diverse candidates per step, improving exploration and the contrast between positive and negative samples. Empirically, we find that $G = 8$ strikes a good trade-off between efficiency and performance.

Training Modes. We compare reinforcement learning, segmentation supervision, and a combined objective for CoPRS-7B. As shown in Figure 6b, the combined objective achieves the best performance. This suggests that reinforcement learning strengthens reasoning, while supervised signals sharpen mask generation. Together they are more effective for complex reasoning segmentation.

Reward Coefficients. We evaluate the impact of reward mixing ratio between mask reward score and format score. Figure 6c compares their combinations, where the format score is one minus the mask score. As the coefficient on the mask reward increases from 0 to 0.7, cIoU improves across all three datasets, but pushing it further to 1.0 slightly degrades performance. This pattern suggests that the segmentation term is the main driver of segmentation quality, while keeping a small contribution from the format score helps regularize the policy and improves generalization, especially on out of distribution data (ReasonSeg). We set the 0.7/0.3 weighting by default, with the segmentation reward dominant and the format score acting as a regularizer, and Figure 6c supports this choice.

Segmentation Loss Combinations. We compare segmentation loss configurations with varying coefficients (see Figure 6d) to assess the contribution of each component, with BCE weight fixed at 1. To avoid the prohibitive cost of LLM experiments, we only probe a few representative weight settings, which already show trends consistent with our expectations. Adding a focal loss term, which emphasizes hard pixels and fine-grained structures, improves segmentation performance. The relative weight between focal and dice loss also affects the balance between global and local mask quality.

5 Conclusions

In this work, we propose CoPRS, connecting language reasoning with segmentation via an interpretable and differentiable interface. CoPRS implements this idea with a learnable concentration query to produce a positional prior instantiated as a heatmap, from which precise masks are decoded, within a unified framework combining reinforcement learning and segmentation supervision. This interface avoids feeding hidden features to the decoder or representing positions in text, instead providing a direct, interpretable alignment between reasoning and mask generation. Empirically, CoPRS attains strong performance across datasets. Further analysis shows that CoT trajectory and heatmap quality strongly correlate with final mask accuracy, and sample visualizations show the same pattern. Overall, CoPRS delivers strong concentration from reasoning and predicts precise masks in a unified formulation, providing a starting point for perception aligned with instructions.

Reproducibility Statement

Reproducibility Statement. We point readers to the fundamental setup in Experimental Setup (Section 4.1), and to the appendix Implementation Details (Section B), which concisely summarizes the pipeline implementation (Section B.1), the design details (Section B.2) and the training configuration (Section B.3). These sections contain the information needed to reproduce our results. We will release code, configurations, and checkpoints upon acceptance.

LLM Usage Statement. Consistent with policies on LLM usage, we used an LLM only for language polishing (see Section B.5 for details). All ideas, experiments, and analyses were produced and verified by the authors, who take full responsibility.

References

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond, 2023.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Shengcao Cao, Zijun Wei, Jason Kuen, Kangning Liu, Lingzhi Zhang, Jiuxiang Gu, HyunJoon Jung, Liang-Yan Gui, and Yu-Xiong Wang. Refer to anything with vision-language prompts. *arXiv preprint arXiv:2506.05342*, 2025.

Yi-Chia Chen, Wei-Hua Li, Cheng Sun, Yu-Chiang Frank Wang, and Chu-Song Chen. Sam4mllm: Enhance multi-modal large language model for referring expression segmentation. In *European Conference on Computer Vision*, pp. 323–340. Springer, 2024.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Henghui Ding, Chang Liu, Suchen Wang, and Xudong Jiang. Vision-language transformer and query generation for referring segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 16321–16330, 2021.

Hanze Dong, Wei Xiong, Bo Pang, Haoxiang Wang, Han Zhao, Yingbo Zhou, Nan Jiang, Doyen Sahoo, Caiming Xiong, and Tong Zhang. Rlhf workflow: From reward modeling to online rlhf. *arXiv preprint arXiv:2405.07863*, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z.

Yanming Guo, Yu Liu, Theodoros Georgiou, and Michael S Lew. A review of semantic segmentation using deep neural networks. *International journal of multimedia information retrieval*, 7(2):87–93, 2018.

Abdul Mueed Hafiz and Ghulam Mohiuddin Bhat. A survey on instance segmentation: state of the art. *International journal of multimedia information retrieval*, 9(3):171–189, 2020.

Jian Hu, Zixu Cheng, Chenyang Si, Wei Li, and Shaogang Gong. Cos: Chain-of-shot prompting for long video understanding. *arXiv preprint arXiv:2502.06428*, 2025.

Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models. *arXiv preprint arXiv:2503.06749*, 2025.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 787–798, 2014.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026, 2023.

Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. *Advances in neural information processing systems*, 24, 2011.

Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9579–9589, 2024.

Mengcheng Lan, Chaofeng Chen, Yue Zhou, Jiaxing Xu, Yiping Ke, Xinjiang Wang, Litong Feng, and Wayne Zhang. Text4seg: Reimagining image segmentation as text generation. In *The Thirteenth International Conference on Learning Representations*, 2025.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.

Chang Liu, Henghui Ding, and Xudong Jiang. Gres: Generalized referring expression segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 23592–23601, 2023a.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023b.

Yuqi Liu, Bohao Peng, Zhisheng Zhong, Zihao Yue, Fanbin Lu, Bei Yu, and Jiaya Jia. Seg-zero: Reasoning-chain guided segmentation via cognitive reinforcement. *arXiv preprint arXiv:2503.06520*, 2025.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 11–20, 2016.

Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*, pp. 565–571. Ieee, 2016.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

Renjie Pi, Lewei Yao, Jiahui Gao, Jipeng Zhang, and Tong Zhang. Perceptiongpt: Effectively fusing visual perception into llm. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 27124–27133, 2024.

Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M Anwer, Eric Xing, Ming-Hsuan Yang, and Fahad S Khan. Glamm: Pixel grounding large multimodal model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13009–13018, 2024.

Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024a.

Zhongwei Ren, Zhicheng Huang, Yunchao Wei, Yao Zhao, Dongmei Fu, Jiashi Feng, and Xiaojie Jin. Pixellm: Pixel reasoning with large multimodal model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26374–26383, 2024b.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning for multi-domain operations applications*, volume 11006, pp. 369–386. SPIE, 2019.

Cheng Tan, Jingxuan Wei, Zhangyang Gao, Linzhuang Sun, Siyuan Li, Ruifeng Guo, Bihui Yu, and Stan Z Li. Boosting the power of small multimodal reasoning models to match larger models with self-consistency training. In *European Conference on Computer Vision*, pp. 305–322. Springer, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

XuDong Wang, Shaolun Zhang, Shufan Li, Kehan Li, Konstantinos Kallidromitis, Yusuke Kato, Kazuki Kozuka, and Trevor Darrell. SegLLM: Multi-round reasoning segmentation with large language models. In *The Thirteenth International Conference on Learning Representations*, 2025a.

Yaoting Wang, Shengqiong Wu, Yuecheng Zhang, Shuicheng Yan, Ziwei Liu, Jiebo Luo, and Hao Fei. Multimodal chain-of-thought reasoning: A comprehensive survey. *arXiv preprint arXiv:2503.12605*, 2025b.

Zhaoqing Wang, Yu Lu, Qiang Li, Xunqiang Tao, Yandong Guo, Mingming Gong, and Tongliang Liu. Cris: Clip-driven referring image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11686–11695, 2022.

Cong Wei, Haoxian Tan, Yujie Zhong, Yujiu Yang, and Lin Ma. Lasagna: Language-based segmentation assistant for complex queries. *arXiv preprint arXiv:2404.08506*, 2024.

Zhao Yang, Jiaqi Wang, Yansong Tang, Kai Chen, Hengshuang Zhao, and Philip HS Torr. Lavt: Language-aware vision transformer for referring image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18155–18165, 2022.

Zhangyue Yin, Qiushi Sun, Zhiyuan Zeng, Qinyuan Cheng, Xipeng Qiu, and Xuan-Jing Huang. Dynamic and generalizable process reward modeling. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4203–4233, 2025.

Zuyao You and Zuxuan Wu. Seg-r1: Segmentation can be surprisingly simple with reinforcement learning. *arXiv preprint arXiv:2506.22624*, 2025.

En Yu, Kangheng Lin, Liang Zhao, Jisheng Yin, Yana Wei, Yuang Peng, Haoran Wei, Jianjian Sun, Chunrui Han, Zheng Ge, et al. Perception-r1: Pioneering perception policy with reinforcement learning. *arXiv preprint arXiv:2504.07954*, 2025a.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025b.

Tao Zhang, Xiangtai Li, Hao Fei, Haobo Yuan, Shengqiong Wu, Shunping Ji, Chen Change Loy, and Shuicheng Yan. Omg-llava: Bridging image-level, object-level, pixel-level reasoning and understanding. *Advances in neural information processing systems*, 37:71737–71767, 2024a.

Yichi Zhang, Ziqiao Ma, Xiaofeng Gao, Suhaila Shakiah, Qiaozi Gao, and Joyce Chai. Groundhog: Grounding large language models to holistic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14227–14238, 2024b.

Zhuosheng Zhang, Aston Zhang, Mu Li, hai zhao, George Karypis, and Alex Smola. Multimodal chain-of-thought reasoning in language models. *Transactions on Machine Learning Research*, 2024c. ISSN 2835-8856.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, et al. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025.

Xueyan Zou, Zi-Yi Dou, Jianwei Yang, Zhe Gan, Linjie Li, Chunyuan Li, Xiyang Dai, Harkirat Behl, Jianfeng Wang, Lu Yuan, et al. Generalized decoding for pixel, image, and language. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15116–15127, 2023a.

Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Wang, Lijuan Wang, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once. *Advances in neural information processing systems*, 36:19769–19782, 2023b.

A Appendix: GRPO Theory and Additional Related Work

A.1 Group Relative Policy Optimization

The reasoning ability of MLLMs is a key factor that influences the reasoning segmentation performance. Since Reinforcement Learning (RL) is an effective way to improve the reasoning ability of LLMs and MLLMs, we employ it to enhance the reasoning segmentation capability of our method.

Proximal Policy Optimization (PPO) (Schulman et al., 2017) is widely used in the RL fine-tuning stage of LLMs. PPO is an actor-critic RL algorithm, which optimizes LLMs by maximizing the following surrogate objective:

$$\mathcal{L}_{\text{PPO}} = \mathbb{E}\left[q \sim P(Q), o \sim \pi_{\boldsymbol{\theta}_{\text{old}}}(O|q)\right] \frac{1}{|o|} \sum_{t=1}^{|o|} \min\left[\frac{\pi_{\boldsymbol{\theta}}(o_t|q,\boldsymbol{o}_{<t})}{\pi_{\boldsymbol{\theta}_{\text{old}}}(o_t|q,\boldsymbol{o}_{<t})} A_t, \text{clip}\left(\frac{\pi_{\boldsymbol{\theta}}(o_t|q,\boldsymbol{o}_{<t})}{\pi_{\boldsymbol{\theta}_{\text{old}}}(o_t|q,\boldsymbol{o}_{<t})}, 1-\varepsilon, 1+\varepsilon\right) A_t\right] \quad (9)$$

where $\pi_{\boldsymbol{\theta}}$ and $\pi_{\boldsymbol{\theta}_{old}}$ are the current and old policy models, and q, o are questions and outputs sampled from the question dataset and the old policy $\pi_{\boldsymbol{\theta}_{old}}$, respectively. ε is a clipping-related hyper-parameter introduced in PPO for stabilizing training. The advantage, A_t, is based on the reward $\{r_{\geq t}\}$ and a learned value function V_ψ, computed by applying Generalized Advantage Estimation (GAE) (Schulman et al., 2015). Furthermore, a per-token KL penalty from a reference model is added to the reward at each token to mitigate over-optimization of the reward model (Ouyang et al., 2022), denoted as:

$$r_t = r_\varphi(q, \boldsymbol{o}_{\leq t}) - \beta \log \frac{\pi_{\boldsymbol{\theta}}(o_t|q, \boldsymbol{o}_{<t})}{\pi_{\text{ref}}(o_t|q, \boldsymbol{o}_{<t})} \quad (10)$$

where r_φ is the reward model, π_{ref} is the reference model, which is usually the initial policy model, and β is the coefficient of the KL penalty.

PPO relies on a separate value function that is typically another model of comparable size to the policy model, imposing heavy memory and computational costs. Additionally, the value function is treated as a baseline in the calculation of the advantage for variance reduction. Moreover, in the LLM context, usually only the last token is assigned a reward score by the reward model, which may complicate the training of a value function that is accurate at each token. Group Relative Policy Optimization (GRPO) (Shao et al., 2024) is proposed to address these drawbacks by obviating the need for additional value function approximation as in PPO, and using the average reward of multiple sampled outputs, produced in response to the same question, as the baseline. Specifically, for each question q, GRPO samples a group of outputs $\{o_1, o_2, \cdots, o_G\}$ from the old policy $\pi_{\boldsymbol{\theta}_{old}}$ and then optimizes the policy model by maximizing the following objective:

$$\begin{aligned}&\mathcal{L}_{\text{GRPO}} = \mathbb{E}_{q\sim P(Q),\{o_i\}_{i=1}^{G}\sim\pi_{\boldsymbol{\theta}_{\text{old}}}(O|q)} \\ &\left\{\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\min\left[\frac{\pi_{\boldsymbol{\theta}}(o_{i,t}|q,\boldsymbol{o}_{i,<t})}{\pi_{\boldsymbol{\theta}_{\text{old}}}(o_{i,t}|q,\boldsymbol{o}_{i,<t})}\hat{A}_{i,t}, \text{clip}\left(\frac{\pi_{\boldsymbol{\theta}}(o_{i,t}|q,\boldsymbol{o}_{i,<t})}{\pi_{\boldsymbol{\theta}_{\text{old}}}(o_{i,t}|q,\boldsymbol{o}_{i,<t})}, 1-\varepsilon, 1+\varepsilon\right)\hat{A}_{i,t}\right] - \beta\mathbb{D}_{\text{KL}}\left[\pi_{\boldsymbol{\theta}}\|\pi_{\text{ref}}\right]\right\}\end{aligned} \quad (11)$$

where ε and β are hyper-parameters, and $\hat{A}_{i,t}$ is the advantage calculated based on relative rewards of the outputs inside each group only. For each question q, a group of outputs $\{o_1, o_2, \cdots, o_G\}$ are sampled from the old policy model $\pi_{\boldsymbol{\theta}_{old}}$. The score of the outputs is obtained through a reward model, yielding G rewards $\{r_1, r_2, \cdots, r_G\}$ correspondingly. The advantages $\hat{A}_{i,t}$ for all tokens in an output are defined as the normalized reward, i.e., $\hat{A}_{i,t} = \tilde{r}_i = \frac{r_i - \text{mean}(\boldsymbol{r})}{\text{std}(\boldsymbol{r})}$. In addition, GRPO directly adds the KL divergence between the trained policy and the reference policy to the loss, avoiding complicating the calculation of $\hat{A}_{i,t}$. The KL divergence is estimated by the following unbiased estimator:

$$\mathbb{D}_{\text{KL}}\left[\pi_{\boldsymbol{\theta}}\|\pi_{\text{ref}}\right] = \frac{\pi_{\text{ref}}(o_{i,t}|q,\boldsymbol{o}_{i,<t})}{\pi_{\boldsymbol{\theta}}(o_{i,t}|q,\boldsymbol{o}_{i,<t})} - \log\frac{\pi_{\text{ref}}(o_{i,t}|q,\boldsymbol{o}_{i,<t})}{\pi_{\boldsymbol{\theta}}(o_{i,t}|q,\boldsymbol{o}_{i,<t})} - 1 \quad (12)$$

A.2 Additional Related Work

GRPO Guided Reinforcement Learning. The GRPO (Shao et al., 2024) strategy addresses reward hacking in RLHF (Dong et al., 2024) by penalizing deviation from a reference policy. However, its reliance on a static reference limits adaptability. This spurred key optimizations: Dynamic Advantage-based Policy Optimization (DAPO) (Yu et al., 2025b) introduces a moving trust region by dynamically updating the reference policy via an exponential moving average, enabling more stable, long-term improvement. Another significant limitation of the original GRPO is its token-level

optimization, which can be computationally intensive and may lead to training instability. Addressing this, Sequence-wise Policy Optimization (GSPO) (Zheng et al., 2025) was proposed to shift the optimization granularity from the token level to the sequence level. By defining a sequence-level importance ratio and advantage, GSPO significantly reduces computational overhead and improves training stability, especially for large-scale models.

Multimodal Chain-of-Thought. Multimodal chain-of-thought (MCoT) (Wang et al., 2025b) reasoning has recently attracted substantial attention, particularly in its integration with MLLMs. Early implementations, such as Multimodal-CoT (Zhang et al., 2024c), have established a basic MCoT pattern by generating intermediate rationales before predictions. MC-CoT (Tan et al., 2024) further refines this paradigm by employing word-level majority during training to enhance the quality of generated rationales. The dependence on high-quality MCoT training data hinders the further improvement of the inference ability of traditional methods. Most recently, the great success of Deepseek-R1 (Guo et al., 2025) has provided a way (i.e., GRPO) to enhance LLM inference capabilities through model autonomous exploration without the need for expensive CoT annotation data. Inspired by this, subsequent works utilize the GRPO strategy to efficiently enhance the reasoning ability of MLLMs. For example, Vision-R1 (Huang et al., 2025) first utilizes existing MLLM and DeepSeek-R1, as well as data filtering, through modal bridging to generate multimodal cold start CoT data, and then applies GRPO to further enhance the model's inference capability. Perception-R1 (Yu et al., 2025a) explores the effects of RL on different perception tasks and optimizes the reward modeling to support perception policy learning. In addition, Chain-of-Shot (Hu et al., 2025) further extends GRPO strategy to optimize frame sampling via binary video summaries. In this work, we study a heatmap-based positional prior that couples MCoT with precise positional perception in a unified training framework for GRPO strategy and segmentation supervision, addressing the gap between high-level reasoning and pixel-level segmentation.

B Appendix: Implementation Details

B.1 Pipeline Implementation

We build on the VERL codebase, which was originally designed for PPO and extended with GRPO functionality.

Sharding Strategy. We shard the VLLM/policy component using Fully Sharded Data Parallel (FSDP), partitioning parameters across devices during training. The lightweight segmentation modules (query head, Q–V attention, and mask decoder) are left unsharded to avoid FSDP overhead and keep their compute/memory costs low. We apply tensor parallelism across attention heads during autoregressive decoding.

FSDP Workers. We precompute image features offline to reduce compute, so the frozen vision backbone is excluded from the training loop. Our framework uses three FSDP workers. (i) The **actor** contains all trainable modules (the MLLM and the segmentation components) and is responsible for parameter updates. (ii) The **rollout worker** runs the MLLM only, taking image and text inputs to generate responses via next token prediction. (iii) The frozen **reference worker** runs an MLLM as the reference policy to compute the KL term in $\mathcal{L}_{\text{GRPO}}$ (eq. (7)) and includes the segmentation modules to decode masks used for computation of mask reward scores and group advantages.

Training Pipeline Implementation. For each annotation, the rollout worker generates G responses for the image–text pair with the current policy by next token prediction, caching the tokens and their log probabilities. The frozen reference worker then runs forward without gradients on the same inputs to compute reference log probabilities for those sampled responses and to decode a mask used in the mask based reward. From each response and its mask signal we compute a scalar reward and convert rewards to group advantages. Next, the actor worker runs forward to obtain the policy log probabilities for the sampled responses and the predicted mask. We form the GRPO objective from the actor log probabilities, the stored old log probabilities from rollout, the reference log probabilities, and the advantages, and we form the segmentation objective from the predicted mask and the ground truth mask. The two objectives are summed and optimized jointly in a single backward pass, updating all trainable modules.

Algorithm 1 Generation of positional prior $\boldsymbol{H}_{\text{prior}}$

Require: Image $\boldsymbol{x}_{\text{img}}$; concentration token embedding $\boldsymbol{e}_{\text{conc}}$; image encoder $\mathcal{F}_{\text{enc}}$; query head $\mathcal{F}_{\text{head}}$; fusion network $\mathcal{F}_{\text{fuse}}$; projection matrices $\{\boldsymbol{W}_i^Q, \boldsymbol{W}_i^K\}_{i=1}^{n_{\text{head}}}$

Ensure: Positional prior $\boldsymbol{H}_{\text{prior}} \in \mathbb{R}^{H \times W}$

$\boldsymbol{K} \leftarrow \mathcal{F}_{\text{enc}}(\boldsymbol{x}_{\text{img}})$ ▷ $\boldsymbol{K} \in \mathbb{R}^{H \times W \times d_k}$

$\boldsymbol{Q} \leftarrow \mathcal{F}_{\text{head}}(\boldsymbol{e}_{\text{conc}})$ ▷ $\boldsymbol{Q} \in \mathbb{R}^{d_q}$

for $i = 1$ to n_{head} **do**

  $\boldsymbol{K}_i \leftarrow \boldsymbol{K}\boldsymbol{W}_i^K$ ▷ $\boldsymbol{K}_i \in \mathbb{R}^{H \times W \times d_h}$

  $\boldsymbol{q}_i \leftarrow \boldsymbol{Q}\boldsymbol{W}_i^Q$ ▷ $\boldsymbol{q}_i \in \mathbb{R}^{d_h}$

  for $(u, v) \in \{1, \ldots, H\} \times \{1, \ldots, W\}$ **do**

    $S_i(u, v) \leftarrow \frac{1}{\sqrt{d_h}} \boldsymbol{q}_i^\top \boldsymbol{K}_i(u, v)$ ▷ $S_i(u, v) \in \mathbb{R}$

  end for

end for

$\boldsymbol{H}_{\text{prior}} \leftarrow \mathcal{F}_{\text{fuse}}\big([\boldsymbol{S}_i]_{i=1}^{n_{\text{head}}}\big)$ ▷ $\mathcal{F}_{\text{fuse}}$: small conv fusion head, $\mathbb{R}^{n_{\text{head}} \times H \times W} \to \mathbb{R}^{H \times W}$

return $\boldsymbol{H}_{\text{prior}}$

B.2 Design Details

Reward Function Design. We use a scalar mask score in $[0, 1]$: given predicted mask and ground truth mask, we compute three overlap metrics (soft IoU, soft Dice, and hard IoU) and take their weighted sum with fixed coefficients 0.5, 0.2, and 0.3, respectively, providing a stable localization signal for how well the prediction covers the instance. For valid outputs, the score is 1.0 by default and is reduced to 0.9 if the `<think>` content is longer than 2048 characters, or if any non-whitespace text appears before `<think>` or after the special token. Thus the five canonical cases are: invalid (0.0); valid and clean (1.0); valid but long `<think>` (0.9); valid but extra text before `<think>` (0.9); valid but extra text after the special token (0.9). For each sample, we take a weighted sum of these two components as the final reward that is assigned to the last valid response token so that GRPO updates the entire trajectory. The relative weights are specified in Section 4.4.

Positional Prior Heatmap Generation. To make the computation of the positional prior $\boldsymbol{H}_{\text{prior}}$ fully reproducible, we detail the heatmap generation procedure in Algorithm 1, starting from the image keys $\boldsymbol{K}$, the concentration query $\boldsymbol{Q}$, and the per-head scaled dot-product scores $S_i(u, v)$. The convolutional fusion head $\mathcal{F}_{\text{fuse}}$ then aggregates $\{\boldsymbol{S}_i\}_{i=1}^{n_{\text{head}}}$ into the final positional prior $\boldsymbol{H}_{\text{prior}} \in \mathbb{R}^{H \times W}$.

B.3 Training configuration

Data and preprocessing. We train on the RefCOCO series. The maximum prompt length is 1300 tokens and the maximum response length is 2000 tokens. For the policy input, images are capped at 705,600 pixels and downsampled if needed; a minimum of 3,136 pixels is enforced. SAM ViT-H features initialize the vision branch.

Hardware and precision. Experiments run on a single node with 8 GPUs. Computation uses bfloat16 for model parameters and fp32 for reductions and buffers.

Parallelism. The policy (VLLM) is trained with Fully Sharded Data Parallel. The rollout service uses tensor parallelism of size 4. The reference worker is also sharded; optimizer state is offloaded.

Batching. Global batch size is 16 (before repeating G times for GRPO). For the actor, micro-batch per device is 2 for updates and 8 for experience collection. Rollout batch size is 16 and the group size is $G = 8$ responses per input.

Optimization. We use AdamW with weight decay 0.01 and $(\beta_1, \beta_2) = (0.9, 0.999)$. The base learning rate is 1.6×10^{-6} with multipliers $25\times$ (query head), $10\times$ (position/prompt encoder), and $5\times$ (mask decoder). Gradient clipping uses a max norm of 1.0. The schedule is one cycle with a final division factor of about 6.7 and no warmup. Total planned training steps are 31,250. Gradient checkpointing is enabled.

Table 6: Comparison on 3B Models.

Method	#Params(B)	GFLOPs	val	test
Seg-R1-3B	3.97	9096.69	56.2	46.6
Seg-Zero-3B	3.97	–	53.1	48.6
CoPRS-3B (Ours)	4.39	9551.52	**60.6**	**52.7**

Table 7: Comparison on 7B Models.

Method	#Params(B)	GFLOPs	val	test
Seg-R1-7B	8.51	20198.96	41.2	53.7
Seg-Zero-7B	8.51	21816.71	62.0	52.0
CoPRS-7B (Ours)	8.93	22283.68	**64.5**	**55.1**

GRPO settings. We use GRPO with sampling number 8, clip ratio 0.2, group-relative advantages, and a fixed KL penalty coefficient 0.2 (low-variance form). The entropy coefficient is 0.0.

Segmentation objectives. Unless noted, $\lambda_{\text{SEG}} = 0.3$, $\lambda_d = 1.5$, and $\lambda_f = 0.0$ at the start; at step 1,500 we set $\lambda_d = 3.0$ and $\lambda_f = 10.0$. Losses are computed only on the valid (unpadded) region.

Rollout and decoding. Rollouts use a VLLM backend with sampling enabled (temperature 1.0, top-p 1.0, top-k disabled). Execution uses bfloat16, up to 64 concurrent sequences, and a cap of 17,408 batched tokens. Chunked prefill is enabled. One image is used per sample.

B.4 Inference efficiency

Compared to representative baselines Seg-Zero and Seg-R1 using GRPO, CoPRS only adds a lightweight query head and a small extra computation for positional prior. To verify the inference efficiency of CoPRS, we conduct experiments on ReasonSeg using the same Qwen2.5-VL-3B/7B. Tables 6 and 7 report the total number of parameters, GFLOPs, and cIoU on ReasonSeg. We observe that CoPRS achieves substantially better performance under both backbones, with comparable parameter counts and inference costs.

B.5 LLM usage statement

In preparing this paper, we used a large language model (LLM) for polishing at the sentence level. We do not directly include the text generated by LLM in our paper. Instead, we use it solely as a reference and for guidance. The model was given the following prompt to guide the text refinement process:

"Slightly polish it sentence by sentence, and give the reasons. Not latex code. Disable online search and do not find citations yourself. You must avoid changing any statistics and avoid distorting my statements."

This prompt was specifically designed to ensure that the LLM's revisions were limited to language refinement and that no statistics or experimental results were altered. The LLM was also instructed not to perform any online searches or generate citations. All final content, including experimental data and results, remains the responsibility of the authors.

C Appendix: Additional Sample Visualizations

Successful Cases. In Figure 8, we present instances from the same category and those relevant to the instruction, all showing elevated responses in the heatmap (yellow regions). More importantly, the heatmap concentrates on the instance specified by the instruction, producing a sharp peak over the target (deep red regions). This concentration guides the decoder, yielding masks with accurate boundaries. These results indicate that the MLLM reasons over the image and text input and identifies the correct referent, while the positional prior concentrates the instances for further precise mask prediction.

Failure Cases. In Figure 7, the first two rows depict scenes with many nearby instances, while the last three rows contain very small targets. Two failure modes emerge. (i) Resolution bottleneck: the positional prior is computed at 256×256 and the SAM embeddings at 64×64; when the longer image side exceeds 2k pixels, tiny objects can vanish after resizing and the decoder cannot reliably recover them. (ii) Same class crowd ambiguity: in dense groups of similar objects (e.g., crowds of people), the positional prior often spreads across many candidates with weak contrast, suggesting that a text only instruction is insufficient to disambiguate near duplicates and that the model has not

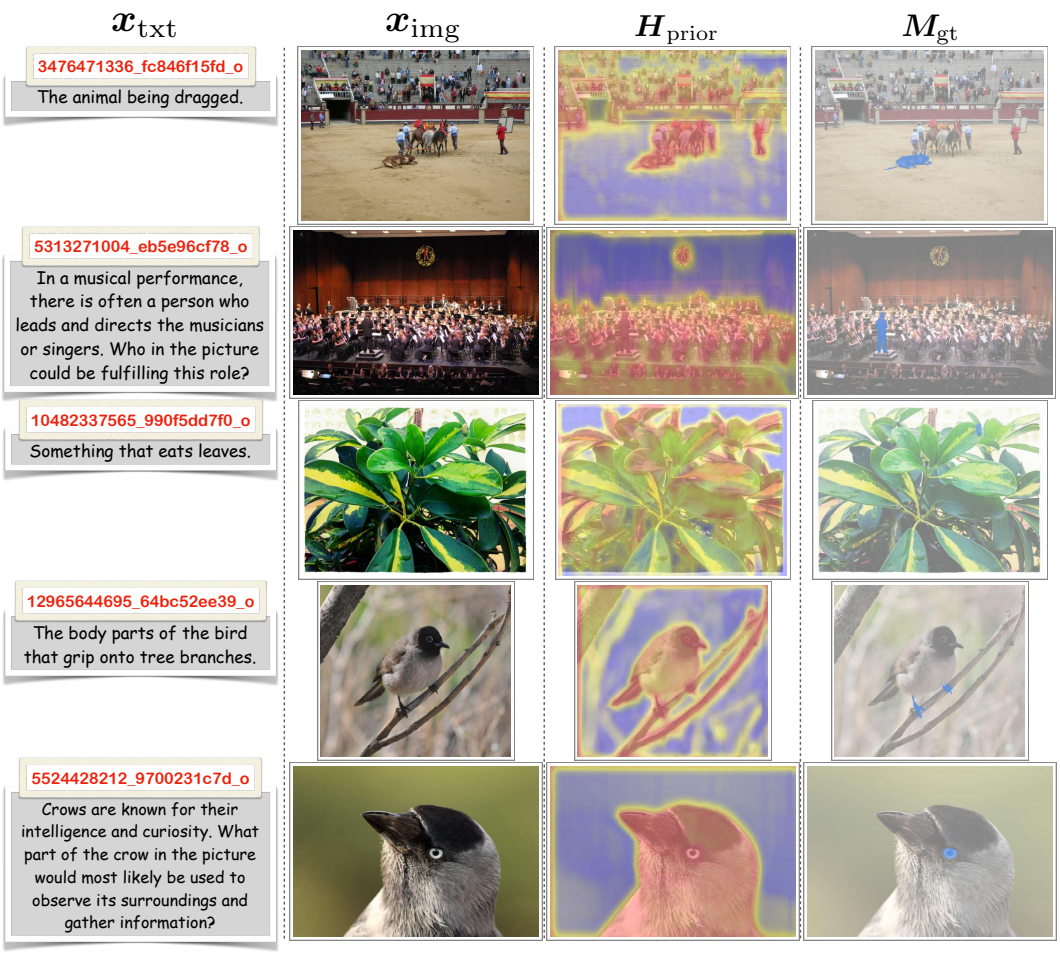


Figure 7: **Failure cases.** With sample ID exposed, all samples are from the ReasonSeg test split. From left to right: image-text pair, positional prior and ground truth mask.

fully learned the subtle semantic cues needed. These observations suggest that higher resolution inputs or multi-scale features, together with stronger instance level language grounding, are likely to improve performance on such cases.

Figure 8: **Additional successful cases.** With sample ID exposed, all samples are from the ReasonSeg test split. From left to right: image-text pair, positional prior, predicted mask, and response.