AnchorSeg: Language Grounded Query Banks for Reasoning Segmentation

Rui Qian[1] , **Chuanhang Deng**[1,2] , **Qiang Huang**[1,2] , **Jian Xiong**[1],
Mingxuan Li[1], **Yingbo Zhou**[1], **Wei Zhai**[1], **Jintao Chen**[1], **Dejing Dou**[1,2]*
[1]College of Computer Science and Artificial Intelligence, Fudan University [2]BEDI Cloud
{qiianruii,dengch2000,dejingdou}@gmail.com

Abstract

Reasoning segmentation requires models to ground complex, implicit textual queries into precise pixel-level masks. Existing approaches rely on a single segmentation token <SEG>, whose hidden state implicitly encodes both semantic reasoning and spatial localization, limiting the model's ability to explicitly disentangle *what to segment* from *where to segment*. We introduce AnchorSeg, which reformulates reasoning segmentation as a structured conditional generation process over image tokens, conditioned on language-grounded query banks. Instead of compressing all semantic reasoning and spatial localization into a single embedding, AnchorSeg constructs an ordered sequence of query banks: latent reasoning tokens that capture intermediate semantic states, and a segmentation anchor token that provides explicit spatial grounding. We model spatial conditioning as a factorized distribution over image tokens, where the anchor query determines localization signals while contextual queries provide semantic modulation. To bridge token-level predictions and pixel-level supervision, we propose Token–Mask Cycle Consistency (TMCC), a bidirectional training objective that enforces alignment across resolutions. By explicitly decoupling spatial grounding from semantic reasoning through structured language grounded query banks, AnchorSeg achieves state-of-the-art results on ReasonSeg test set (67.7% gIoU and 68.1% cIoU). All codes and models are publicly available at https://github.com/rui-qian/AnchorSeg.

1 Introduction

Reasoning segmentation aims to predict pixel-level segmentation masks by grounding complex language reasoning into visual scenes (Lai et al., 2024). Unlike traditional referring segmentation, which operates on explicit object descriptions (*e.g.*, "*the red car on the left*"), reasoning segmentation involves reasoning over abstract concepts, spatial relationships, and implicit references (*e.g.*, "*the object that provides shade in this scene*") to identify and segment target regions. Such capability is essential for intelligent vision–language systems that can accurately interpret and respond to human intent in complex visual environments.

Recent advances in Large Multimodal Models (LMMs) have demonstrated remarkable progress in combining language understanding with visual perception (Liu et al., 2023b; Li et al., 2024; Alayrac et al., 2022; Wang et al., 2023; Zhu et al., 2024; Zhang et al., 2024). To endow LMMs with segmentation capability, LISA (Lai et al., 2024) introduces a special segmentation token <SEG> into the language model vocabulary. During autoregressive generation, when the model produces this token, its hidden representation is extracted and used as a single, unified query to condition a visual decoder (*e.g.*, SAM (Kirillov et al., 2023)) for mask prediction. While effective, this paradigm (Wu et al., 2024c; Xia et al., 2024; Qian et al., 2025) compresses both semantic reasoning and spatial localization cues into a single <SEG> embedding vector. Such implicit compression limits the model's ability to explicitly disentangle *what to segment* (semantic reasoning) from *where to segment* (spatial grounding), potentially hindering the model's performance in complex reasoning scenarios.

We argue that reasoning segmentation can be formulated as a structured conditional generation process, where spatial grounding is performed at the image-token level and explicitly conditioned on language-derived queries that encode intermediate reasoning states. To this end, we propose AnchorSeg, a framework built upon language grounded query banks, where an explicit **Anchor** query grounds language reasoning to visual tokens and guides **Seg**mentation. Instead of relying on a single segmentation token, AnchorSeg introduces

*Corresponding Author

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 20490–20505

a set of learnable latent reasoning tokens alongside a segmentation anchor token, which are autoregressively generated by LMMs. This design yields an ordered sequence of query embeddings: contextual queries that capture intermediate reasoning states, and an anchor query that serves as an explicit spatial grounding signal.

Our key innovation lies in how these queries interact with visual representations. We model spatial grounding as a factorized conditional distribution over image tokens, where each token's relevance to the target object is conditioned on both the contextual queries (providing semantic modulation) and the anchor query (determining spatial grounding). This formulation enables explicit token-level language grounding, producing a spatial prior that is injected into the visual features before mask decoding. The entire query banks, ordered by the autoregressive generation process, are then fed into the SAM decoder, providing structured, language-conditioned supervision for final mask prediction.

To bridge the gap between token-level spatial responses and pixel-level mask supervision, we introduce a Token–Mask Cycle Consistency (TMCC) training objective. This bidirectional constraint enforces alignment between the language-grounded token-level predictions and the ground-truth masks at both resolutions, ensuring that spatial reasoning is consistent across the language-vision hierarchy. By explicitly modeling token-level spatial response factorization through structured language-grounded query banks, AnchorSeg establishes a more effective coupling between language reasoning and visual segmentation.

Our contributions can be summarized as follows:

- We reformulate reasoning segmentation as a structured conditional generation problem, introducing language-grounded query banks that explicitly disentangle semantic reasoning from spatial grounding at the image-token level.
- We propose a factorized formulation for language-grounded conditioning, where anchor queries produce explicit localization signals and contextual queries provide semantic modulation, enabling enhanced language-to-vision alignment.
- We introduce Token–Mask Cycle Consistency (TMCC), a bidirectional training objective that enforces alignment between token-level spatial responses and pixel-level mask supervision across resolutions.
- We conduct extensive experiments on ReasonSeg and RefCOCO(+/g) datasets. The proposed AnchorSeg achieves state-of-the-art results on ReasonSeg test set (67.7% gIoU and 68.1% cIoU).

2 Related Work

2.1 Large Multimodal Models

Large Multimodal Models (LMMs) underpin reasoning segmentation by enabling joint perception and language understanding. Representative models include VisionLLM (Wang et al., 2023), MiniGPT-4 (Zhu et al., 2024), and GPT4RoI (Zhang et al., 2024), which show that integrating LLMs with visual features enables open-ended multimodal reasoning, particularly in region-level grounding and object-level understanding. BLIP (Li et al., 2023) focuses on vision–language alignment by leveraging frozen image encoders while supporting generative multimodal modeling. Flamingo (Alayrac et al., 2022) introduces a flexible interleaved vision–text interface that enables few-shot multimodal reasoning across diverse vision–language tasks. LLaVA (Liu et al., 2023b) and LLaVA-NeXT (Li et al., 2024) further enhance visual reasoning and document understanding, while more recent works (Hurst et al., 2024; Wu et al., 2024b; OpenAI, 2025) integrate text, audio, and video into unified multimodal models.

Despite existing efforts, current LMMs still primarily generate textual responses and struggle to deliver structured dense predictions, often relying on weak or implicit grounding signals rather than precise pixel-level masks (Radford et al., 2021; Li et al., 2023; Alayrac et al., 2022). Recent works extend LMMs toward pixel-level visual grounding by introducing segmentation tokens, pixel decoders, visual prompting modules and chain-of-thought reasoning for grounding (Lai et al., 2024; Ren et al., 2024; Qian et al., 2025). In contrast, our approach introduces a structured formulation that explicitly bridges language reasoning and pixel-level segmentation, enabling more interpretable grounding.

2.2 Reasoning Segmentation

Reasoning segmentation extends referring expression segmentation to settings where models are required to infer implicit visual concepts and output pixel-level masks under complex scenarios. LISA (Lai et al., 2024) pioneers this task, injecting a <SEG> token into LMMs and decoding its embedding into masks under an "embedding-as-mask" paradigm, accompanied by the ReasonSeg

benchmark for implicit reasoning queries. Follow-up works expand this formulation: GSVA (Xia et al., 2024) generalizes the segmentation token to multi-target reasoning and introduces <REJ> for non-existent-object rejection. SESAME (Wu et al., 2024c) mitigates hallucination and false-premise failures via a pipeline that verifies existence, overcomes the premise, and then segments. READ (Qian et al., 2025) looks into how <SEG> contributes to grounding and proposes a "Similarity as Points" module for enhancement.

Beyond token design, recent research explores pixel-grounded LMMs with richer interactivity. GLaMM (Rasheed et al., 2024) enables joint language generation and segmentation in grounded conversations, while PixelLM (Ren et al., 2024) integrates a lightweight pixel decoder and segmentation codebook directly into LMMs, removing reliance on the external SAM (Ravi et al., 2025). RSVP (Lu et al., 2025) further couples multimodal chain-of-thought reasoning with segmentation, generating interpretable region proposals before refinement. In this work, we model reasoning segmentation as token-level conditional generation of spatial responses, factorized over image tokens and conditioned on a structured language query bank.

3 Proposed AnchorSeg

In this section, we present AnchorSeg, which formulates reasoning segmentation as conditional decoding by jointly leveraging language-grounded spatial priors and structured query conditioning. In Fig. 1, AnchorSeg comprises three key components: 1) an LMM $\mathcal{G}_\mathcal{T}$ that autoregressively generates latent reasoning tokens and a segmentation anchor token <SEG>, forming a language-grounded query bank $\mathbf{Q}$; 2) a language grounded spatial conditioning module that computes similarity between the anchor query and image tokens to produce a spatial prior $\mathbf{P}$; 3) a SAM mask decoder conditioned on the query bank to predict the final segmentation mask. Given an input image and a textual prompt, AnchorSeg generates a <SEG> token to indicate the target object, along with a set of latent reasoning tokens that encode intermediate semantic cues. Unlike prior works, AnchorSeg explicitly grounds the <SEG> token to image tokens via a similarity map, producing a coarse spatial localization signal. Instead of using a single <SEG> token, we elevate this signal into a language-grounded spatial prior and inject it into the visual feature space, enabling segmentation to be guided by both explicit visual priors and structured language queries.

3.1 Vanilla Reasoning Segmentation with Language-Conditioned <SEG> Query

Problem Definition: We first revisit the vanilla formulation of reasoning segmentation proposed in LISA (Lai et al., 2024). Given an input image $\mathbf{x}_{img} \in \mathbb{R}^{h\times w\times c}$ and a textual prompt $\mathbf{x}_{txt}$, the goal of reasoning segmentation is to predict a binary mask $\hat{\mathbf{M}} \in \{0,1\}^{h\times w}$ corresponding to the visual concept described in the text as

$$\hat{\mathbf{M}} = \arg\max_{\hat{\mathbf{M}}} \mathcal{G}_\theta\big(\hat{\mathbf{M}} \mid \mathbf{x}_{img}, \mathbf{x}_{txt}\big), \quad (1)$$

where $\mathcal{G}_\theta = \mathcal{G}_\mathcal{T} \oplus \mathcal{G}_\mathcal{V}$ denotes a cascaded architecture composed of an LMM $\mathcal{G}_\mathcal{T}$ (*e.g.*, LLaVA (Liu et al., 2023b)), and a visual backbone model $\mathcal{G}_\mathcal{V}$ (*e.g.*, SAM (Kirillov et al., 2023)).

To enable segmentation within a language modeling framework, the vanilla approach extends the vocabulary of LMM $\mathcal{G}_\mathcal{T}$ with a special placeholder, denoted as <SEG>. During autoregressive generation, the language model produces an output token sequence as

$$\hat{\mathbf{y}}_{txt} = \mathcal{G}_\mathcal{T}(\mathbf{x}_{img}, \mathbf{x}_{txt}), \quad (2)$$

in which the <SEG> token is generated when a segmentation output is required. Let $\boldsymbol{q}_{seg}$ denote the hidden representation corresponding to the predicted <SEG> token extracted from the final transformer layer and projected via a lightweight MLP $\varphi(\cdot)$, which serves as a single, language-conditioned segmentation query that implicitly encodes both semantics and spatial guidance. On the visual side, the segmentation model $\mathcal{G}_\mathcal{V}^{enc}$ (instantiated by SAM (Kirillov et al., 2023)) first encodes image $\mathbf{x}_{img}$ into SAM's dense visual features as

$$\boldsymbol{q}_{seg} = \varphi(\tilde{\boldsymbol{h}}_{seg}) \in \mathbb{R}^d, \quad \mathbf{f} = \mathcal{G}_\mathcal{V}^{enc}(\mathbf{x}_{img}). \quad (3)$$

The final segmentation mask is then predicted by conditioning the SAM mask decoder on the single segmentation query:

$$\hat{\mathbf{M}} = \mathcal{G}_\mathcal{V}^{dec}(\mathbf{f}, \boldsymbol{q}_{seg}). \quad (4)$$

In this vanilla paradigm, the <SEG> token serves as a single, unified interface between language reasoning and visual segmentation. Both semantic reasoning and spatial localization cues are implicitly compressed into a single segmentation query, without preserving the internal structure of the language reasoning process.

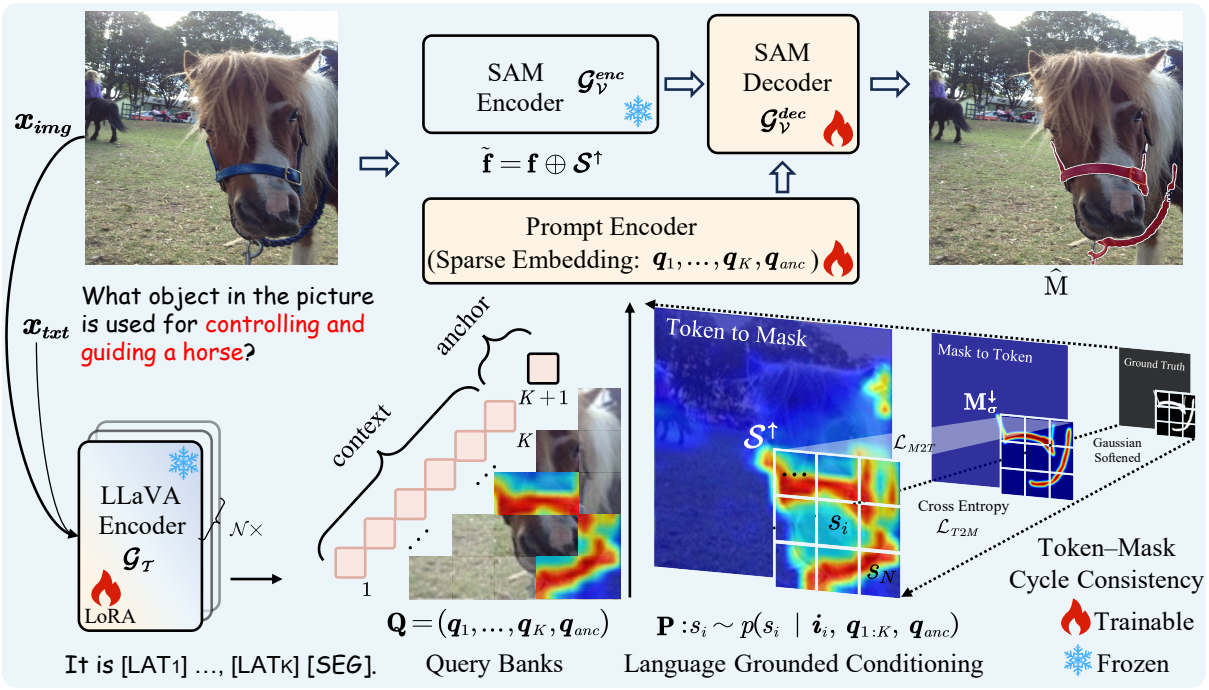

Figure 1: Overview of AnchorSeg. Given an input image and textual query, the LMM $\mathcal{G}_T$ autoregressively generates latent reasoning tokens and a segmentation anchor token <SEG>, forming a language-grounded query bank $\mathbf{Q} = \{q_1, \ldots, q_K, q_{anc}\}$. The anchor query q_{anc} computes similarity with image tokens to produce a spatial prior $\mathbf{P}$, which is injected into visual features $\mathbf{f}$. The query bank then conditions the SAM decoder to predict the final mask $\hat{\mathbf{M}}$. This design explicitly disentangles spatial grounding (anchor query) from semantic reasoning (contextual queries).

3.2 Extended Reasoning Segmentation with Language-Grounded Query Banks

While effective, the vanilla formulation relies on a single segmentation token to implicitly bridge language reasoning and mask decoding. The alignment between linguistic semantics and visual representations remains largely unexplored. We therefore reformulate reasoning segmentation as a conditional decoding problem, where spatial grounding is explicitly modeled at the image-token level and jointly conditioned on structured language queries.

Query Bank Construction. We construct a conditional query sequence by extending the vocabulary of LMM $\mathcal{G}_T$ with a set of learnable placeholders. Specifically, we introduce K latent reasoning tokens together with a segmentation token <SEG>, which are inserted into the textual response and produced autoregressively as part of the language modeling process, such that the segmentation token <SEG> is explicitly conditioned on the preceding latent reasoning tokens, yielding the ordered query sequence as $\texttt{<LAT}_1\texttt{>}, \ldots, \texttt{<LAT}_K\texttt{>}, \texttt{<SEG>}$. Rather than operating on discrete symbols, we formulate the conditional query sequence in the continuous embedding space. Let the final-layer hidden states corresponding to the latent reasoning tokens and the segmentation token be denoted as $q_1, \ldots, q_K \in \mathbb{R}^d$ and $q_{anc} \in \mathbb{R}^d$, respectively. We define the conditional query sequence as

$$\mathbf{Q} = (q_1, \ldots, q_K, q_{anc}), \tag{5}$$

where the contextual queries $q_{1:K}$ encode intermediate reasoning states that condition the segmentation process at the semantic level, while the anchor query q_{anc} serves as a spatial grounding token. This ordered query sequence provides structured, language-conditioned supervision for subsequent spatial conditioning and mask decoding.

Language Grounded Conditioning. Inspired by token-level factorization in autoregressive language modeling, we formulate language-grounded spatial conditioning as a conditional generation process over image tokens. Let $\boldsymbol{I} = \{i_1, \ldots, i_N\}$, $i_i \in \mathbb{R}^d$, denote the image token representations produced by LMM $\mathcal{G}_T$. We define a set of spatial responses $\boldsymbol{S} = \{s_1, \ldots, s_N\}$, where each s_i corresponds to the relevance of image token i_i to the target object. The conditional distribution of $\boldsymbol{S}$ given the query bank is factorized as

$$p(\boldsymbol{S} \mid \mathbf{Q}) = \prod_{i=1}^{N} p\left(s_i \,\middle|\, i_i, \underbrace{q_1, \ldots, q_K}_{\text{context}}, \underbrace{q_{anc}}_{\text{anchor}}\right). \tag{6}$$

This formulation explicitly disentangles the roles of the query tokens: the anchor query $\boldsymbol{q}_{anc}$ determines where to attend in the image, while the contextual queries $\boldsymbol{q}_{1:K}$ provide semantic modulation under which spatial grounding is performed.

Under the formulation in Eq. (6), each spatial response s_i is generated conditionally from the corresponding image token $\boldsymbol{i}_i$ and the language-grounded query bank, $s_i \sim p(s_i \mid \boldsymbol{i}_i,\ \boldsymbol{q}_{1:K},\ \boldsymbol{q}_{anc})$. Practically, we instantiate the conditional distribution using an anchor-based similarity function as

$$s_i = \boldsymbol{i}_i^\top \boldsymbol{q}_{anc}, \tag{7}$$

where the anchor query explicitly produces a localization signal over image tokens. Although the similarity computation depends only on $\boldsymbol{q}_{anc}$, the contextual queries $\boldsymbol{q}_{1:K}$ influence each spatial response s_i implicitly by shaping the generation of the anchor query through autoregressive reasoning. The token-level responses $\boldsymbol{S}$ are then reshaped and projected to the spatial domain to obtain a language-grounded spatial prior $\mathbf{P} \in \mathbb{R}^{C\times H\times W}$. Next, this spatial prior is injected into the visual feature map $\mathbf{f} \in \mathbb{R}^{C\times H\times W}$ via element-wise addition, yielding conditioned visual features $\tilde{\mathbf{f}} = \mathbf{f} \oplus \mathbf{P}$.

Conditional Mask Decoding. When interacting with the SAM (Kirillov et al., 2023) mask decoder, the language-grounded query bank is treated as an ordered query sequence. Specifically, the query sequence $\{\boldsymbol{q}_1, \ldots, \boldsymbol{q}_K, \boldsymbol{q}_{anc}\}$ is augmented with learnable positional embeddings $\{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_{K+1}\}$ prior to cross-attention, ensuring both order-awareness and explicit role distinction between contextual and anchor queries. The position-augmented query bank is then used to condition the SAM mask decoder, which predicts the final segmentation mask as

$$\hat{\mathbf{M}} = \mathcal{G}_{\mathcal{V}}^{dec}\big(\tilde{\mathbf{f}},\ \{\boldsymbol{q}_1, \ldots, \boldsymbol{q}_K, \boldsymbol{q}_{anc}\}\big), \tag{8}$$

where mask prediction is explicitly conditioned on both language grounded spatial prior $\mathbf{P}$ and structured reasoning queries $(\boldsymbol{q}_{1:K}, \boldsymbol{q}_{anc})$. This formulation explicitly decouples spatial grounding and semantic reasoning. The segmentation anchor query determines spatial correspondence, while the latent reasoning queries provide contextual modulation. By unifying both components under a conditional decoding framework, AnchorSeg enables robust reasoning-based segmentation guided by structured language representations and explicit visual priors.

3.3 Training Objectives

Token–Mask Cycle Consistency (TMCC). Under the unified formulation in Eq. (6), the spatial responses $\boldsymbol{S}$ act as a latent representation bridging token-level language grounding and pixel-level mask supervision. To regularize this representation across resolutions, we introduce a token–mask cycle consistency objective. Given a ground-truth binary mask $\mathbf{M} \in \{0,1\}^{H\times W}$, we apply Gaussian smoothing to obtain a softened target mask $\mathbf{M}_\sigma \in [0,1]^{H\times W}$. 1) Token-to-Mask Consistency. We reshape and upsample the token-level responses $\boldsymbol{S}$ to the image resolution, yielding a spatial map $\mathcal{S}^{\uparrow} \in [0,1]^{H\times W}$, which is supervised by $\mathbf{M}_\sigma$ using binary cross-entropy and Dice losses as

$$\mathcal{L}_{T2M} = \lambda_{bce}\mathcal{L}_{bce}(\mathcal{S}^{\uparrow}, \mathbf{M}_\sigma) + \lambda_{dice}\mathcal{L}_{dice}(\mathcal{S}^{\uparrow}, \mathbf{M}_\sigma). \tag{9}$$

2) Mask-to-Token Consistency. Conversely, we downsample the soft target mask $\mathbf{M}_\sigma$ to the image-token resolution, obtaining $\mathbf{M}_\sigma^{\downarrow} \in [0,1]^N$, and enforce token-level alignment with $\boldsymbol{S}$ as

$$\mathcal{L}_{M2T} = \lambda_{bce}\mathcal{L}_{bce}(\boldsymbol{S}, \mathbf{M}_\sigma^{\downarrow}) + \lambda_{dice}\mathcal{L}_{dice}(\boldsymbol{S}, \mathbf{M}_\sigma^{\downarrow}). \tag{10}$$

Overall Objectives. The two terms form a bidirectional cycle consistency constraint over the latent spatial responses $\boldsymbol{S}$ as

$$\mathcal{L}_{TMCC} = \mathcal{L}_{T2M} + \mathcal{L}_{M2T}. \tag{11}$$

The final training objective combines TMCC with the language modeling loss $\mathcal{L}_{txt}$ and the segmentation loss $\mathcal{L}_{mask}$ as

$$\mathcal{L} = \lambda_{txt}\mathcal{L}_{txt} + \lambda_{mask}\mathcal{L}_{mask} + \lambda_{TMCC}\mathcal{L}_{TMCC}. \tag{12}$$

4 Experiments

4.1 Experimental Setting

Implementation Details. We employ LLaVA-1.5 (7B/13B) (Liu et al., 2023b) as the multimodal encoder $\mathcal{G}_{\mathcal{T}}$, while ViT-H SAM (Kirillov et al., 2023) serves as the vision backbone $\mathcal{G}_{\mathcal{V}}$ specifically for mask generation. Visual features are extracted using CLIP-ViT-L/14@336, which operates on inputs of size 336×336. To prevent object truncation caused by default center-cropping and to align the similarity map with SAM's input, we implement a padding-based resizing scheme (scaling the longest side to 336) and padding the image for the CLIP input. Training is performed on a single NVIDIA

Table 1: Performance comparison of reasoning segmentation models on the ReasonSeg dataset. Models are sorted by cIoU scores. * denotes results reproduced from official implementations, and ft indicates models fine-tuned on 239 samples.

Method	val		test					
	overall		short query		long query		overall	
	gIoU	cIoU	gIoU	cIoU	gIoU	cIoU	gIoU	cIoU
X-Decoder (Zou et al., 2023)	22.6	17.9	20.4	11.6	22.2	17.5	21.7	16.3
Grounded-SAM (Liu et al., 2024)	26.0	14.5	17.8	10.8	22.4	18.6	21.3	16.4
SEEM (Zou et al., 2024)	25.5	21.2	20.1	11.5	25.6	20.8	24.3	18.7
OVSeg (Liang et al., 2023)	28.5	18.6	18.0	15.5	28.7	22.5	26.1	20.8
GRES (Liu et al., 2023a)	22.4	19.9	17.6	15.0	22.6	23.8	21.3	22.0
*SESAME (Wu et al., 2024c)	40.3	41.6	28.9	26.3	37.3	31.9	34.9	30.7
LLaVA1.5-7B + OVSeg (Lai et al., 2024)	38.2	23.5	24.2	18.7	44.6	37.1	39.7	31.8
*GSVA (Xia et al., 2024)	45.6	41.5	37.9	36.5	44.3	46.0	42.8	43.8
*PixelLM (Ren et al., 2024)	49.7	49.6	39.5	38.8	49.5	45.6	47.1	44.3
LISA-7B (Lai et al., 2024)	52.9	54.0	40.6	40.6	49.4	51.0	47.3	48.4
HyperSeg-3B (Wei et al., 2025)	59.2	56.7	-	-	-	-	-	-
VISA-7B (Yan et al., 2024)	52.7	57.8	-	-	-	-	-	-
VideoLISA-3.8B (Bai et al., 2024)	61.4	67.1	43.8	42.7	56.9	57.7	53.8	54.4
LISA-7B-LLaVA1.5 (ft) (Lai et al., 2024)	61.3	62.9	48.3	46.3	57.9	59.7	55.6	56.9
READ-7B-LLaVA1.5 (ft) (Qian et al., 2025)	59.8	67.6	52.6	49.5	60.4	61.0	58.5	58.6
LISA++-7B-LLaVA1.5 (ft) (Yang et al., 2023)	64.2	68.1	49.6	**51.1**	59.3	61.7	57.0	59.5
RSVP-GPT (Lu et al., 2025)	64.7	63.1	55.4	50.4	61.9	62.5	60.3	60.0
AnchorSeg-7B-LLaVA1.5 (ft)	**68.3**	**75.9**	**57.3**	48.2	**67.0**	**71.3**	**64.6**	**65.9**
Qwen3-VL-8B (Liang et al., 2026)	70.3	70.0	-	-	-	-	66.0	53.7
Seg-ReSearch-8B (Liang et al., 2026)	**73.3**	72.2	-	-	-	-	67.4	59.0
LLaVA1.5-13B + OVSeg (Lai et al., 2024)	37.9	26.4	27.1	19.4	46.1	40.6	41.5	34.1
LISA-13B-LLaVA1.5 (Lai et al., 2024)	57.7	60.3	50.8	50.0	54.7	50.9	53.8	50.8
LISA-13B-LLaVA1.5(ft) (Lai et al., 2024)	65.0	72.9	55.4	50.6	63.2	65.3	61.3	62.2
READ-13B-LLaVA1.5 (ft) (Qian et al., 2025)	-	-	55.4	**53.7**	64.4	65.1	62.2	62.8
AnchorSeg-13B-LLaVA1.5 (ft)	67.9	**73.0**	**59.6**	51.8	**70.2**	**73.3**	**67.7**	**68.1**

Table 2: Results for referring expression comprehension on RefCOCO, RefCOCO+, and RefCOCOg (Precision@0.5). (full-ft) denotes full training of the u-LLaVA LLM. For u-LLaVA-7B, we report results using the "mask2bbox" strategy to ensure fair comparison.

Method	RefCOCO			RefCOCO+			RefCOCOg	
	val	testA	testB	val	testA	testB	val	test
u-LLaVA-7B (Xu et al., 2024) (full-ft)	86.04	89.47	82.26	74.09	81.16	66.61	79.87	81.68
LISA-Vicuna-7B (Lai et al., 2024) (ft)	85.39	88.84	82.59	74.23	79.46	68.40	79.34	80.42
GSVA-Vicuna-7B (Xia et al., 2024) (ft)	86.27	89.92	83.77	72.81	78.78	68.01	81.58	81.83
AnchorSeg-LLaVA1.5-7B	**89.10**	**92.10**	**84.49**	**80.89**	**85.87**	**73.29**	**84.09**	**84.45**
LISA-Vicuna-13B (Lai et al., 2024) (ft)	85.92	89.05	83.16	74.86	81.08	68.87	80.09	81.48
GSVA-Vicuna-13B (Xia et al., 2024) (ft)	87.71	90.49	84.57	76.52	81.69	70.35	83.90	84.85
AnchorSeg-LLaVA1.5-13B	**91.34**	**94.56**	**87.54**	**84.25**	**89.75**	**77.71**	**87.79**	**87.09**

Table 3: Benchmarking Generalized Referring Expression Segmentation (GRES) on the gRefCOCO dataset (Liu et al., 2023a). Models are listed in ascending order based on the cIoU of the *val* set. Values are derived from (Liu et al., 2023a). N-acc. denotes the accuracy of identifying null targets, and ft indicates models fine-tuned on the gRefCOCO training split.

Method	Validation Set			Test Set A			Test Set B		
	gIoU	cIoU	N-acc.	gIoU	cIoU	N-acc.	gIoU	cIoU	N-acc.
MattNet (Yu et al., 2018)	48.24	47.51	41.15	59.30	58.66	44.04	46.14	45.33	41.32
LTS (Jing et al., 2021)	52.70	52.30	-	62.64	61.87	-	50.42	49.96	-
VLT (Ding et al., 2021)	52.00	52.51	47.17	63.20	62.19	48.74	50.88	50.52	47.82
CRIS (Wang et al., 2022)	56.27	55.34	-	63.42	63.82	-	51.79	51.04	-
LAVT (Yang et al., 2022)	58.40	57.64	49.32	65.90	65.32	49.25	55.83	55.04	48.46
ReLA (Liu et al., 2023a)	63.60	62.42	56.37	70.03	69.26	59.02	61.02	59.88	58.40
LISA-Vicuna-7B (Lai et al., 2024)	32.21	38.72	2.71	48.54	52.55	6.37	39.65	44.79	5.00
GSVA-Vicuna-7B (Xia et al., 2024)	63.32	61.70	56.45	70.11	69.23	63.50	61.34	60.26	58.42
LISA-Vicuna-7B (ft) (Lai et al., 2024)	61.63	61.76	54.67	66.27	68.50	50.01	58.84	60.63	51.91
GSVA-Vicuna-7B (ft) (Xia et al., 2024)	66.47	63.29	62.43	71.08	69.93	65.31	62.23	60.47	60.56
AnchorSeg-LLaVA1.5-7B(ft)	**74.76**	**68.68**	**76.00**	**75.75**	**73.43**	**71.78**	**68.25**	**64.56**	**68.27**

H800 GPU (80GB). We use a mixed dataset of ReasonSeg and referring segmentation samples, comprising approximately ∼10k images. We utilize LoRA (Hu et al., 2022) ($r = 8$ for 7B; $r = 64$ for 13B) and optimize via AdamW (Loshchilov and Hutter, 2019) with a learning rate of 3e-4 and 100 warmup steps.

Evaluation Metrics. Consistent with established methodologies in prior works (Kazemzadeh et al., 2014; Mao et al., 2016; Lai et al., 2024), we employ specific metrics for different tasks. For Reasoning Segmentation evaluation, we report gIoU and cIoU. Specifically, gIoU represents the mean of the Intersection-over-Union (IoU) scores calculated for each individual image, while cIoU is derived from the ratio of the cumulative intersection to the cumulative union across the dataset. For the Referring Expression Comprehension (REC) task, we adopt Precision@0.5 metric, where a prediction is considered correct if the IoU is at least 0.5.

4.2 Results on ReasonSeg Dataset

Comparison with the State-of-the-Art. In Table 1, AnchorSeg performs favorably against competitive baselines. On the ReasonSeg validation split, our 7B model achieves 68.3% gIoU and 75.9% cIoU, outperforming RSVP-GPT by +3.6% and +12.8%, respectively. This advantage carries over to the test set, where AnchorSeg surpasses RSVP-GPT by +4.3% gIoU and +5.9% cIoU on the overall split, with notable gains on long queries (+5.1% gIoU and +8.8% cIoU). Scaling to 13B further improves performance. Compared to READ-13B, AnchorSeg achieves up to +5.5% gIoU and +5.3% cIoU on the test overall split, while delivering substantial improvements on long queries, validating its effectiveness in complex reasoning scenarios.

4.3 Results on RefCOCO(+/g) Dataset

Comparison with the State-of-the-Art. Table 2 presents the quantitative results for referring expression comprehension based on Precision@0.5. AnchorSeg consistently outperforms GSVA-Vicuna-7B across all evaluated benchmarks. On the RefCOCO dataset, AnchorSeg achieves 89.10%, 92.10%, and 84.49% on the val, testA, and testB splits, respectively, surpassing the corresponding GSVA results. On RefCOCO+, AnchorSeg improves over GSVA by +8.08%, +7.09%, and +5.28% on the val, testA, and testB splits. On RefCOCOg, AnchorSeg attains 84.09% on val and 84.45% on test, exceeding GSVA by +2.51% and +2.62%, respectively.

4.4 Results on gRefCOCO Dataset

Comparison with the State-of-the-Art. In Table 3, we report results of AnchorSeg-LLaVA1.5-7B on the gRefCOCO dataset. Compared to the fine-tuned GSVA-Vicuna-7B, AnchorSeg achieves consistent improvements across all metrics. On the validation set, AnchorSeg achieves 74.76% gIoU, 68.68% cIoU, and 76.00% N-acc, improving over GSVA by +8.29%, +5.39%, and +13.57%, respectively. On Test A, AnchorSeg attains 75.75% gIoU, 73.43% cIoU, and 71.78% N-acc, with gains of +4.67%, +3.50%, and +6.47%. On Test B, AnchorSeg further improves over GSVA by +6.02% gIoU, +4.09% cIoU, and +7.71% N-acc.

4.5 Qualitative Results

Figure 2 shows that AnchorSeg achieves more precise segmentation results compared to prior methods such as LISA (Lai et al., 2024) and SESAME (Wu et al., 2024c). The model is particularly effective in fine-grained cases, as illustrated by the accurate segmentation of the target in the 4^{th} row.

4.6 Ablation Study

In this section, we conduct ablation studies on the ReasonSeg *val* set to investigate the contribution of each core component in our proposed AnchorSeg.

Table 4: Ablation on Conditional Query Sequence Scaling. We report performance with varying sequence lengths N. Note that $N = K + 1$, comprising K latent reasoning tokens and one anchor query <SEG>. $N = 8$ yields the best performance.

Exp. ID	Sequence Len. (N)	gIoU	cIoU
1	4	64.8	73.4
2	8	**68.3**	**75.9**
3	16	64.1	72.8
4	32	67.4	71.3

Impact of Conditional Query Sequence Scaling. We investigate the effect of scaling the conditional query sequence $\mathbf{Q}$. We vary the total length N (where $N = K + 1$, consisting of K latent reasoning tokens and one anchor query <SEG>) to evaluate the representational capacity. As shown in Table 4, a short sequence ($N = 4$, *i.e.*, $K = 3$) restricts the capacity to encode intermediate reasoning states, resulting in degraded performance (73.4% cIoU). The performance peaks at $N = 8$,

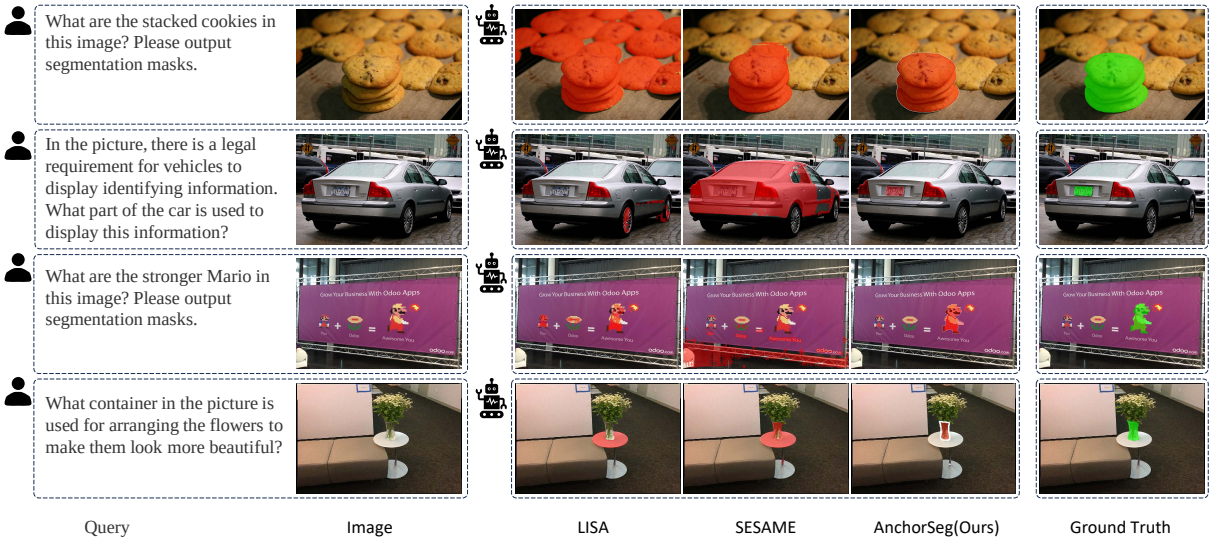

Figure 2: Visual comparison of AnchorSeg (Ours) with prior works on the ReasonSeg *val* set.

achieving the best 68.3% gIoU and 75.9% cIoU, indicating an optimal balance between semantic reasoning and spatial grounding. However, further expansion ($N \geq 16$) leads to saturation, with cIoU dropping to 72.8% at $N = 16$, likely due to the increased optimization difficulty in the continuous embedding space.

Table 5: Ablation on the Contextual ($\boldsymbol{q}_{1:K}$) and Anchor ($\boldsymbol{q}_{anc}$) Queries, Spatial Prior P, and TMCC.

Exp. ID	P	TMCC	$\boldsymbol{q}_{1:K}$	$\boldsymbol{q}_{anc}$	gIoU	cIoU
1			✓	✓	51.8	61.4
2		✓	✓	✓	58.4	66.3
3	✓		✓	✓	68.3	71.6
4	✓	✓		✓	67.5	74.0
5	✓	✓	✓	✓	**68.3**	**75.9**

Table 6: Ablation on Token-Mask Cycle Consistency (TMCC). We investigate the contribution of the cycle consistency terms: $\mathcal{L}_{T2M}$ and $\mathcal{L}_{M2T}$. The combination yields the best synergy.

Exp. ID	$\mathcal{L}_{T2M}$	$\mathcal{L}_{M2T}$	gIoU	cIoU
1	✓		67.4	73.8
2		✓	64.4	68.2
3	✓	✓	**68.3**	**75.9**

Impact of Language Grounded Conditioning. We progressively analyze the effects on factorization of contextual $\boldsymbol{q}_{1:K}$ and anchor $\boldsymbol{q}_{anc}$ queries, spatial prior $\mathbf{P}$, and TMCC. In Table 5, Exp. 1 denotes that neither the spatial prior nor TMCC is used during training. In this case, the contextual queries $\boldsymbol{q}_{1:K}$ and the anchor query $\boldsymbol{q}_{anc}$ have no functional differentiation and fully degrade into $\boldsymbol{q}_{1:K+1}$ <SEG> tokens, all serving as segmentation placeholder tokens (*i.e.*, placeholders with no role-specific distinction). Exp. 2 denotes that both the spatial prior and TMCC are applied during training. However, the spatial prior is not provided to SAM (*i.e.*, it is not added to the image feature). Exp. 3 incorporates the spatial prior $\mathbf{P}$ (derived from $\boldsymbol{q}_{anc}$) but omits TMCC, yielding a significant boost to 71.6% cIoU. Exp. 4 applies both $\mathbf{P}$ and TMCC but discards the contextual queries $\boldsymbol{q}_{1:K}$, improving to 74.0% cIoU. Finally, the full method (Exp. 5) re-introduces $\boldsymbol{q}_{1:K}$ to the decoder, achieving the highest 75.9% cIoU, which demonstrates the effectiveness of the proposed design.

Impact of Token-Mask Cycle Consistency. We assess the effectiveness of the proposed Token-Mask Cycle Consistency (TMCC), which regularizes the latent spatial responses $\boldsymbol{S}$ via a bidirectional constraint. As shown in Table 6, optimizing with only $\mathcal{L}_{T2M}$ yields 67.4% gIoU and 73.8% cIoU. In contrast, using $\mathcal{L}_{M2T}$ alone leads to a performance drop to 64.4% gIoU, likely due to the coarse resolution of the latent space. By jointly optimizing both objectives ($\mathcal{L}_{TMCC} = \mathcal{L}_{T2M} + \mathcal{L}_{M2T}$), the model achieves the best performance, reaching 68.3% gIoU and 75.9% cIoU.

4.7 Runtime Analysis

We evaluate the computational efficiency of AnchorSeg using a single NVIDIA A100-SXM4-40GB GPU, with results reported in Tables 7 and 8. During training on the ReasonSeg dataset (batch size of 2), our method incurs an average iteration latency of 2.94 seconds and a peak memory usage of 29.89 GB. Compared to previous methods such as LISA (Lai et al., 2024) (1.26 s, 23.68 GB) and GSVA (Xia et al., 2024) (2.13 s, 25.73 GB), AnchorSeg exhibits a slightly higher training latency and memory usage, mainly due to the addi-

Table 7: Comparing the training cost of our AnchorSeg to state-of-the-art methods.

Model	Training Latency (s)	Memory Usage (GB)	Trainable (%)	Trainable Params	Total Params
SESAME (Wu et al., 2024c)	1.11	23.15	3.73%	288.25M	7.73B
PixelLM (Ren et al., 2024)	1.22	23.02	5.25%	375.72M	7.16B
LISA (Lai et al., 2024)	1.26	23.68	3.74%	288.25M	7.71B
GSVA (Xia et al., 2024)	2.13	25.73	3.73%	288.26M	7.73B
AnchorSeg	2.94	29.89	3.94%	306.29M	7.77B

Table 8: Comparing the runtime speed of our AnchorSeg to state-of-the-art methods.

Model	GSVA (Xia et al., 2024)	SESAME (Wu et al., 2024c)	LISA (Lai et al., 2024)	PixelLM (Ren et al., 2024)	AnchorSeg
Speed (FPS)	3.98	4.64	4.68	9.24	4.00

tional query construction and spatial conditioning modules. For inference performance (Table 8), AnchorSeg achieves a throughput of 4.00 FPS with batch size 1. This is comparable to GSVA (Xia et al., 2024) (3.98 FPS) and LISA (Lai et al., 2024) (4.68 FPS), while being slower than PixelLM (Ren et al., 2024) (9.24 FPS). Overall, the results show that AnchorSeg maintains a reasonable trade-off between model complexity and runtime efficiency.

5 Conclusion

In this work, we reformulate reasoning segmentation as a structured conditional generation problem over image tokens, conditioned on language-grounded query banks. We identified a key limitation in existing approaches: compressing all semantic reasoning and spatial localization into a single `<SEG>` token embedding restricts the model's ability to explicitly disentangle *what to segment* from *where to segment*. To address this, we introduce AnchorSeg, which decouples spatial grounding from semantic reasoning through structured language grounded query banks. We hope our factorized formulation and explicit query bank design will inspire future research on language-grounded visual perception and structured multimodal reasoning.

6 Limitations

Although AnchorSeg effectively decouples spatial grounding from semantic reasoning through structured language-grounded query banks, several limitations remain.

Dependence on LMM Quality. The quality of the language-grounded query bank depends on the underlying LMM's ability to generate meaningful latent reasoning tokens. While our approach benefits from stronger LMMs that produce more discriminative intermediate representations, weaker or smaller LMMs may generate less informative contextual queries, potentially limiting the effectiveness of the factorized formulation. Future work could explore self-supervised pretraining objectives designed to enhance the quality of latent reasoning tokens.

Fixed Query Bank Size. AnchorSeg uses a fixed number K of latent reasoning tokens for all queries, regardless of reasoning complexity. Simple queries (*e.g.*, "*the red car*") may not require multiple reasoning states, while highly complex queries (*e.g.*, "*the object that could prevent water damage in this scene*") might benefit from more intermediate reasoning states. Adaptive mechanisms that dynamically adjust the number of latent tokens based on query complexity could improve both efficiency and performance.

Computational Overhead. Compared to vanilla single-token approaches, AnchorSeg introduces additional computational costs through multiple query tokens, token-level similarity computation, and the TMCC training objective. While these components are essential for explicit disentanglement of semantic reasoning and spatial grounding, more efficient architectures or sparse attention mechanisms could reduce the computational burden for real-time applications.

Generalization Beyond Reasoning Segmentation. While our approach demonstrates strong performance on reasoning segmentation tasks, its applicability to other vision-language tasks such as visual question answering, image captioning, or open-vocabulary detection remains unexplored. Investigating how language-grounded query banks can be adapted to broader multimodal understanding tasks is an important direction for future research.

Acknowledgments This work was supported by Dejing Dou's Research Startup Fund from Fudan University and the computations in this research were performed using the CFFF platform of Fudan University.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L. Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. 2022. Flamingo: a visual language model for few-shot learning. In *NeurIPS*.

Zechen Bai, Tong He, Haiyang Mei, Pichao Wang, Ziteng Gao, Joya Chen, Zheng Zhang, and Mike Zheng Shou. 2024. One token to seg them all: Language instructed reasoning segmentation in videos. In *NeurIPS*.

Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. 2018. Coco-stuff: Thing and stuff classes in context. In *CVPR*.

Xianjie Chen, Roozbeh Mottaghi, Xiaobai Liu, Sanja Fidler, Raquel Urtasun, and Alan Yuille. 2014. Detect what you can: Detecting and representing objects using holistic models and body parts. In *CVPR*.

Henghui Ding, Chang Liu, Suchen Wang, and Xudong Jiang. 2021. Vision-language transformer and query generation for referring segmentation. In *ICCV*.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *ICLR*.

Yutao Hu, Qixiong Wang, Wenqi Shao, Enze Xie, Zhenguo Li, Jungong Han, and Ping Luo. 2023. Beyond one-to-one: Rethinking the referring image segmentation. In *ICCV*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. GPT-4o system card. In *arXiv*.

Ya Jing, Tao Kong, Wei Wang, Liang Wang, Lei Li, and Tieniu Tan. 2021. Locate then segment: A strong pipeline for referring image segmentation. In *CVPR*.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. 2014. Referitgame: Referring to objects in photographs of natural scenes. In *EMNLP*.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, et al. 2023. Segment anything. In *ICCV*.

Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. 2024. Lisa: Reasoning segmentation via large language model. In *CVPR*.

Bo Li, Hao Zhang, Kaichen Zhang, Dong Guo, Yuanhan Zhang, Renrui Zhang, Feng Li, Ziwei Liu, and Chunyuan Li. 2024. LLaVA-next: What else influences visual instruction tuning beyond data?

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. In *ICML*.

Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. 2023. Open-vocabulary semantic segmentation with mask-adapted clip. In *CVPR*.

Tianming Liang, Qirui Du, Jian-Fang Hu, Haichao Jiang, Zicheng Lin, and Wei-Shi Zheng. 2026. SegReSearch: Segmentation with interleaved reasoning and external search. In *arXiv*.

Chang Liu, Henghui Ding, and Xudong Jiang. 2023a. GRES: Generalized referring expression segmentation. In *CVPR*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023b. Visual instruction tuning. In *NeurIPS*.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, and et al Su, Hang. 2024. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *ECCV*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *ICLR*.

Yi Lu, Jiawang Cao, Yongliang Wu, Bozheng Li, Licheng Tang, Yangguang Ji, Chong Wu, Jay Wu, and Wenbo Zhu. 2025. RSVP: Reasoning segmentation via visual prompting and multi-modal chain-of-thought. In *ACL*.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, and et al. Yuille, Alan L. 2016. Generation and comprehension of unambiguous object descriptions. In *CVPR*.

Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. 2017. The mapillary vistas dataset for semantic understanding of street scenes. In *ICCV*.

OpenAI. 2025. Gpt-5 system card. `https://cdn.openai.com/gpt-5-system-card.pdf`.

Rui Qian, Xin Yin, and Dejing Dou. 2025. Reasoning to attend: Try to understand how <SEG> token works. In *CVPR*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *ICML*.

Vignesh Ramanathan, Anmol Kalia, Vladan Petrovic, Yi Wen, Baixue Zheng, Baishan Guo, Rui Wang, Aaron Marquez, Rama Kovvuri, Abhishek Kadian, et al. 2023. PACO: Parts and attributes of common objects. In *CVPR*.

Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M Anwer, Eric Xing, Ming-Hsuan Yang, and Fahad S Khan. 2024. GlaMM: Pixel grounding large multimodal model. In *CVPR*.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. 2025. SAM 2: Segment anything in images and videos. In *ICML*.

Zhongwei Ren, Zhicheng Huang, Yunchao Wei, Yao Zhao, Dongmei Fu, Jiashi Feng, and Xiaojie Jin. 2024. PixelLM: Pixel reasoning with large multimodal model. In *CVPR*.

Wenhai Wang, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao, et al. 2023. VisionLLM: large language model is also an open-ended decoder for vision-centric tasks. In *NeurIPS*.

Zhaoqing Wang, Yu Lu, Qiang Li, Xunqiang Tao, Yandong Guo, Mingming Gong, and Tongliang Liu. 2022. CRIS: Clip-driven referring image segmentation. In *CVPR*.

Cong Wei, Yujie Zhong, Haoxian Tan, Yong Liu, Jie Hu, Dengjie Li, Zheng Zhao, and Yujiu Yang. 2025. HyperSeg: Hybrid segmentation assistant with fine-grained visual perceiver. In *CVPR*.

Jianzong Wu, Xiangtai Li, Xia Li, Henghui Ding, Yunhai Tong, and Dacheng Tao. 2024a. Towards robust referring image segmentation. In *TIP*.

Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. 2024b. Next-GPT: Any-to-any multimodal llm. In *ICML*.

Tsung-Han Wu, Giscard Biamby, David Chan, Lisa Dunlap, Ritwik Gupta, Xudong Wang, Joseph E Gonzalez, and Trevor Darrell. 2024c. See, say, and segment: Teaching lmms to overcome false premises. In *CVPR*.

Zhuofan Xia, Dongchen Han, Yizeng Han, Xuran Pan, Shiji Song, and Gao Huang. 2024. GSVA: Generalized segmentation via multimodal large language models. In *CVPR*.

Jinjin Xu, Liwu Xu, Yuzhe Yang, Xiang Li, Yanchun Xie, Yi-Jie Huang, and Yaqian Li. 2024. u-LLaVA: Unifying multi-modal tasks via large language model. In *ECAI*.

Cilin Yan, Haochen Wang, Shilin Yan, Xiaolong Jiang, Yao Hu, Guoliang Kang, Weidi Xie, and Efstratios Gavves. 2024. VISA: Reasoning video object segmentation via large language models. In *ECCV*.

Senqiao Yang, Tianyuan Qu, Xin Lai, Zhuotao Tian, Bohao Peng, Shu Liu, and Jiaya Jia. 2023. LISA++: An improved baseline for reasoning segmentation with large language model. In *arXiv*.

Zhao Yang, Jiaqi Wang, Yansong Tang, Kai Chen, and et al. Zhao, Hengshuang. 2022. LAVT: Language-aware vision transformer for referring image segmentation. In *CVPR*.

Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. 2018. MAttNet: Modular attention network for referring expression comprehension. In *CVPR*.

Shilong Zhang, Peize Sun, Shoufa Chen, Min Xiao, Wenqi Shao, Wenwei Zhang, Yu Liu, Kai Chen, et al. 2024. GPT4RoI: Instruction tuning large language model on region-of-interest. In *ECCV*.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. 2017. Scene parsing through ade20k dataset. In *CVPR*.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2024. MiniGPT-4: Enhancing vision-language understanding with advanced large language models. In *ICLR*.

Xueyan Zou, Zi-Yi Dou, Jianwei Yang, Zhe Gan, Linjie Li, Chunyuan Li, Xiyang Dai, Harkirat Behl, Jianfeng Wang, Lu Yuan, et al. 2023. Generalized decoding for pixel, image, and language. In *CVPR*.

Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Wang, Lijuan Wang, and et al. Gao, Jianfeng. 2024. Segment everything everywhere all at once. In *NeurIPS*.

A Ethics and Societal Impact

Declaration of LLM Usage. During manuscript preparation, Large Language Models (LLMs) were used in a limited manner for linguistic refinement, including checking for grammatical accuracy and improving sentence flow. Additionally, LLMs assisted in generating routine, non-critical code snippets (*e.g.*, data loading utilities). We affirm that the conceptualization, methodological framework, and experimental design are entirely the original work of the authors. All AI-generated content was rigorously reviewed and validated by the authors, who assume full responsibility for the accuracy and integrity of the final manuscript.

Broader Impact Statement. This study is intended primarily for academic research. We affirm that the data acquisition process strictly adhered to established ethical protocols and data protection regulations. Furthermore, the deployment of all models in this work is in full compliance with their specific licensing terms. We are committed to upholding research integrity and have assessed that this work poses no significant risk of adverse societal consequences.

B Datasets

Following the data organization paradigm established by prior works, we construct a comprehensive training mixture to equip our model with multi-granularity visual understanding and complex reasoning capabilities. As shown in Table 9, our training data is categorized into eight primary types: (1) sem_seg for fundamental semantic understanding; (2) refer_seg for standard referring expression comprehension; (3) neg_refer_seg and (4) correct_refer_seg for robust handling of false premises; (5) vqa for general conversational abilities; and finally, (6) reason_seg, (7) reason_seg_plus, and (8) multi_reason_seg to facilitate advanced logical deduction and multi-target grounding.

Semantic Segmentation (sem_seg). This category incorporates widely-used public datasets, including ADE20K, COCO-Stuff, Pascal Part, PACO-LVIS, and Mapillary. To align with the instruction-following objective, we randomly select specific categories for each image and reformulate the data into a visual question-answering (VQA) format using predefined templates.

Referring Segmentation (refer_seg). The RefClef dataset (also referred to as ReferItGame) is a foundational benchmark for referring expression comprehension, comprising 130K expressions for nearly 100K regions across 20,000 images. Unlike later COCO-based benchmarks, RefClef includes annotations for both discrete object instances and amorphous "stuff" categories.

The RefCOCO series (including RefCOCO, RefCOCO+, and RefCOCOg) consists of large-scale benchmarks for referring expression comprehension and segmentation based on MS-COCO. RefCOCO and RefCOCO+ contain concise phrases, with the latter prohibiting absolute spatial terms to emphasize appearance attributes. In contrast, RefCOCOg features longer, more complex descriptive sentences.

RefZom is a challenging benchmark established to address the "one-to-one" assumption limitation in traditional referring segmentation by incorporating three distinct settings: one-to-zero (no target), one-to-one (single target), and one-to-many (multiple targets). It comprises a total of 55,078 images and 74,942 annotated objects. The dataset is partitioned into a training set of 43,749 images (58,356 objects) and a testing set of 11,329 images (16,586 objects).

G-RefCOCO is a large-scale benchmark for Generalized Referring Expression Segmentation (GRES), comprising 278,232 expressions, including 80,022 multi-target and 32,202 empty-target ones, referring to the objects in 19,994 images. The images are split into four subsets: training, validation, test-A, and test-B, following the same UNC partition of RefCOCO.

Negative Referring Segmentation (neg_refer_seg). R-RefCOCO(+/g) datasets extend RefCOCO, RefCOCO+, and RefCOCOg by introducing negative text inputs that require empty mask predictions. Adhering to the original UNC and UMD partitions, they augment the training data with a 1:1 positive-negative ratio and significantly expand the validation sets with approximately 10 negative descriptions per reference object to rigorously evaluate robustness.

Correct Referring Segmentation (correct_refer_seg). FP-RefCOCO(+/g) extend the RefCOCO series to facilitate the False Premise Correction task. The datasets utilizes large language models to create context-aware false premises by modifying specific objects, attributes, or relations. Preserving original splits, the benchmark maintains a 1:1 positive-to-negative ratio across roughly 20k-25k images per dataset to evaluate reasoning robustness.

Table 9: Overview of the training dataset mixture.

Dataset Types	Source Datasets	Total	Train Split	Val Split	Test Split
sem_seg	ADE20K (Zhou et al., 2017)	22.21K	20.21K	2k	n.a.
	COCO-Stuff (Caesar et al., 2018)	118.29K	n.a.	n.a.	n.a.
	Pascal Part (Chen et al., 2014)	4.37K	n.a.	n.a.	n.a.
	PACO-LVIS (Ramanathan et al., 2023)	45.79K	n.a.	n.a.	n.a.
	Mapillary (Neuhold et al., 2017)	18.00K	n.a.	n.a.	n.a.
refer_seg	RefClef (Kazemzadeh et al., 2014)	17.98K	n.a.	n.a.	n.a.
	RefCOCO (Kazemzadeh et al., 2014)	19.99K	16.99K	1.50K	1.50K
	RefCOCO+ (Kazemzadeh et al., 2014)	19.99K	16.99K	1.50K	1.50K
	RefCOCOg (Mao et al., 2016)	25.80K	21.90K	1.3K	2.6K
	RefZom (Hu et al., 2023)	55.08K	43.75K	n.a.	11.33K
	G-RefCOCO (Liu et al., 2023a)	19.99K	16.99K	1.50K	1.50K
neg_refer_seg	R-RefCOCO (Wu et al., 2024a)	15.10K	n.a.	n.a.	n.a.
	R-RefCOCO+ (Wu et al., 2024a)	15.10K	n.a.	n.a.	n.a.
	R-RefCOCOg (Wu et al., 2024a)	20.04K	n.a.	n.a.	n.a.
correct_refer_seg	FP-RefCOCO (Wu et al., 2024c)	19.99K	16.99K	1.50K	1.50K
	FP-RefCOCO+ (Wu et al., 2024c)	19.99K	16.99K	1.50K	1.50K
	FP-RefCOCOg (Wu et al., 2024c)	25.80K	21.90K	1.3K	2.6K
vqa	LLaVA-Instruct-150K (Liu et al., 2023b)	157.71K	n.a.	n.a.	n.a.
reason_seg	ReasonSeg (Lai et al., 2024)	1.22K	239	200	779
reason_seg_plus	instance_seg (Yang et al., 2023)	58.50K	n.a.	n.a.	n.a.
	cot (Yang et al., 2023)	3.04K	n.a.	n.a.	n.a.
	conversations (Yang et al., 2023)	2.65K	n.a.	n.a.	n.a.
	caption (Yang et al., 2023)	1.34K	n.a.	n.a.	n.a.
multi_reason_seg	MultiReasonSeg (Ren et al., 2024)	105.34K	102.35K	942	2.05K

Visual Question Answering (vqa). We utilize the LLaVA-Instruct-150k dataset, which consists of approximately 157.71K vision-language instruction samples. This dataset is incorporated to maintain the model's general conversational proficiency and its ability to follow diverse natural language instructions.

Reasoning Segmentation (reason_seg). ReasonSeg is a benchmark established to evaluate reasoning segmentation capabilities, comprising 1,218 image-instruction-mask data samples. The images are annotated with implicit text instructions that necessitate complex reasoning or world knowledge, categorized into short phrases and long sentences. To ensure reliable assessment, the dataset is partitioned into 239 training, 200 validation, and 779 testing samples, allocating a larger proportion to testing for rigorous evaluation.

Enhanced Reasoning Segmentation (reason_seg_plus). ReasonSeg-Plus is based on the COCO2017 dataset. The Instance Segmentation subset focuses on training models for instance-level delineation, while the COT (Chain-of-Thought) subset enhances global scene understanding and reasoning segmentation. Additionally, the Conversation subset equips models with the ability to perform segmentation within dynamic dialogues, and the Caption subset integrates segmentation tasks directly into image captioning workflows.

Multi-target Reasoning Segmentation (multi_reason_seg). MUSE is a large-scale benchmark established for the Multi-target Reasoning Segmentation task, comprising 246k question-answer pairs and 910k instance-level mask annotations sourced from the LVIS dataset. It features complex queries with arbitrary numbers of open-set targets, averaging 3.7 objects per response. The dataset is partitioned into 239k training, 2.8k validation, and 4.3k testing samples to rigorously evaluate pixel-level reasoning and understanding capabilities.

C Implementation Details

Given an input image $\mathbf{x}_{img} \in \mathbb{R}^{h \times w \times c}$ and a textual prompt $\mathbf{x}_{txt}$, LLaVA ($\mathbf{x}_{img}$, $\mathbf{x}_{txt}$) produces the image token embedding $\boldsymbol{I}_{img} \in \mathbb{R}^{576 \times 4096}$ and the <SEG> token embedding $\boldsymbol{q}_{anc} \in \mathbb{R}^{4096}$. The similarity map is computed as $\boldsymbol{q}_{anc} \times \boldsymbol{I}_{img}^{T} = (1 \times 4096) \times (576 \times 4096)^{T} = 1 \times 576$, followed by reshaping ($576 \rightarrow 24 \times 24$) into $1 \times 24 \times 24$, and then interpolating ($1 \times 24 \times 24 \rightarrow 1 \times 256 \times 256$) and convolving ($1 \times 256 \times 256 \rightarrow 256 \times 64 \times 64$) to obtain the spatial prior $\mathbf{P} \in \mathbb{R}^{256 \times 64 \times 64}$. Note

that SAM's dense visual features are given by $\mathbf{f} = \mathcal{G}_{\mathcal{V}}^{enc}(\mathbf{x}_{img}) \in \mathbb{R}^{256\times64\times64}$. We inject the spatial prior $\mathbf{P}$ into the visual feature map $\mathbf{f}$ via element-wise addition, yielding conditioned visual features $\tilde{\mathbf{f}} = \mathbf{f} \oplus \mathbf{P}$ for SAM mask generation.

C.1 Interpolation Details for Token–Mask Cycle Consistency (TMCC)

We specify the interpolation operators used in TMCC and refer to Algorithm 1 for the complete token-to-image resizing pipeline.

Token-to-Mask interpolation. Given token-level responses $\boldsymbol{S} \in \mathbb{R}^N$ with $N = G^2$ (*e.g.*, $G = 24$), the continuous similarity map used by the token-to-mask term $\mathcal{L}_{T2M}$ is constructed exactly following Algorithm 1 (normalization, reshape to $G \times G$, bilinear upsampling to $L_{vl} = 336$, aspect-ratio aligned crop, bilinear resize back to the supervision resolution). In our code, all bilinear resizes are implemented via PyTorch `F.interpolate(..., mode='bilinear')`. When resizing within the SAM-style preprocessing (`apply_image_torch`), we explicitly set `align_corners=False` and `antialias=True`, and then pad zeros on the bottom/right to obtain a square canvas.

Mask-to-Token interpolation and softening. For the reverse direction $\mathcal{L}_{M2T}$, the ground-truth mask $\mathbf{M} \in \{0,1\}^{H\times W}$ is first mapped to the $L_{vl} = 336$ canvas using the same resize-and-pad operator as above (bilinear resize with `align_corners=False`, `antialias=True`, followed by bottom/right zero padding), and then downsampled to the token grid using nearest-neighbor interpolation:

$$\mathbf{M}^{grid} = \mathrm{Interp}\big(\mathbf{M}^{336};\, G \times G,\, \texttt{nearest}\big) \in \{0,1\}^{G\times G}. \tag{13}$$

The Gaussian soft target is then constructed *on the token grid* as

$$\begin{aligned} \mathbf{M}_\sigma^{grid} &= \mathbf{M}^{grid} * G_\sigma \in [0,1]^{G\times G}, \\ \mathbf{M}_\sigma^{\downarrow} &= \mathrm{vec}\Big(\mathbf{M}_\sigma^{grid}\Big) \in [0,1]^N. \end{aligned} \tag{14}$$

We implement G_σ using a normalized 2D Gaussian kernel with default $\sigma = 7.0$ and nominal kernel size 31, with adaptive kernel sizing for small masks and `reflect` padding during convolution.

C.2 Extending the Anchor Query to Multiple Spatial Anchors

While AnchorSeg uses a single segmentation anchor query $\boldsymbol{q}_{anc}$ to produce a language-grounded spatial prior, the framework naturally extends to *multiple* spatial anchors to better support multi-target scenarios and more complex spatial relations.

Multi-target & multi-anchor formulation. Let N_{seg} denote the number of *targets* to be segmented in an image (i.e., the number of masks to be predicted). For each target $m \in \{1,\dots,N_{seg}\}$, the LMM can generate T anchor tokens $\{\texttt{<SEG>}_{m,t}\}_{t=1}^{T}$, yielding anchor queries $\{\boldsymbol{q}_{anc}^{(m,t)}\}_{t=1}^{T}$ (optionally sharing the same contextual query bank $\boldsymbol{q}_{1:K}$). For each anchor, we compute

$$s_i^{(m,t)} = \boldsymbol{i}_i^\top \boldsymbol{q}_{anc}^{(m,t)},$$

producing spatial priors $\{\mathbf{P}^{(m,t)}\}_{t=1}^{T}$ for target m. These priors can be injected into the SAM feature map either independently (*e.g.*, running the mask decoder T times per target) or jointly via a lightweight fusion:

$$\tilde{\mathbf{f}}^{(m)} = \mathbf{f} \oplus \Phi\Big(\mathrm{concat}\left(\mathbf{P}^{(m,1)},\dots,\mathbf{P}^{(m,T)}\right)\Big), \tag{15}$$

where $\Phi(\cdot)$ is a small conv head. This yields one fused spatial prior per target and allows anchors to specialize to different regions within each target.

When multi-anchor helps. We expect multiple anchors to be particularly beneficial for (i) explicit multi-object prompts ("segment {A,B,C}"), (ii) one-to-many referring and open-vocabulary multi-target segmentation, and (iii) queries involving structured spatial relations (*e.g.*, "the two cups on the left of the plate"), where a single coarse spatial prior may be insufficient to disambiguate multiple valid regions.

Potential challenges. Multi-anchor variants introduce additional computation (roughly linear in T if decoding is repeated) and may require mechanisms to mitigate anchor competition and duplicate predictions. Practical options include anchor-wise NMS/merging, diversity regularization across anchors, or assigning anchors to targets via bipartite matching during training when instance-level supervision is available. We leave a full exploration of optimal multi-anchor generation and training as future work.

Algorithm 1 Language Grounded Conditioning Algorithm

Require: Input image $\mathbf{x}_{img} \in \mathbb{R}^{h\times w\times c}$ and text prompt $\mathbf{x}_{txt}$, vision-language model (*e.g.*, LLaVA) that outputs contextual queries $\boldsymbol{q}_{1:K} \in \mathbb{R}^{K\times d}$ and anchor queries $\{\boldsymbol{q}_{anc}^{(t)} \in \mathbb{R}^d\}_{t=1}^{N_{seg}}$ (default $N_{seg}=1$). For each t, the ordered query bank is $\mathbf{Q}^{(t)} = \big(\boldsymbol{q}_1, \ldots, \boldsymbol{q}_K, \boldsymbol{q}_{anc}^{(t)}\big)$. Image token embeddings $\boldsymbol{I} \in \mathbb{R}^{N\times d}$ (stacking $\boldsymbol{I} = \{\boldsymbol{i}_1, \ldots, \boldsymbol{i}_N\}$ with $\boldsymbol{i}_j \in \mathbb{R}^d$). Resize long-side target $L_{vl} = 336$ for similarity alignment and $L_{sam} = 256$ for SAM input. SAM visual features are given by $\mathbf{f} = \mathcal{G}_{\mathcal{V}}^{enc}(\mathbf{x}_{img}) \in \mathbb{R}^{C\times H\times W}$, with $(C, H, W) = (256, 64, 64)$. f_θ is a three-layer convolutional head with channel dimensions $1 \to 4 \to 16 \to C$.

Ensure: Conditioned visual features $\tilde{\mathbf{f}} \in \mathbb{R}^{C\times H\times W}$ per anchor query $\boldsymbol{q}_{anc}$ for SAM mask generation.

for $t = 1, \ldots, N_{seg}$ **do**

$\boldsymbol{s}^{(t)} \leftarrow \boldsymbol{I}\,\boldsymbol{q}_{anc}^{(t)} \in \mathbb{R}^N$ /* Equivalently $s_j^{(t)} = \boldsymbol{i}_j^\top \boldsymbol{q}_{anc}^{(t)}$ for $j = 1, \ldots, N$ */

$\bar{\boldsymbol{s}}^{(t)} \leftarrow \dfrac{\boldsymbol{s}^{(t)} - \min(\boldsymbol{s}^{(t)})}{\max(\boldsymbol{s}^{(t)}) - \min(\boldsymbol{s}^{(t)}) + \epsilon}$

$G \leftarrow \sqrt{N};\ \tilde{\mathbf{P}}^{(t)} \leftarrow \mathrm{reshape}\Big(\bar{\boldsymbol{s}}^{(t)};\, G \times G\Big)$

$\tilde{\mathbf{P}}^{(t)} \leftarrow \mathrm{interp}\Big(\tilde{\mathbf{P}}^{(t)};\, L_{vl} \times L_{vl},\, \texttt{bilinear}\Big)$

$\alpha \leftarrow \dfrac{L_{vl}}{\max(h, w)};\ h' \leftarrow \lfloor h\alpha + 0.5 \rfloor,\ w' \leftarrow \lfloor w\alpha + 0.5 \rfloor$

$\tilde{\mathbf{P}}^{(t)} \leftarrow \mathrm{interp}\Big(\tilde{\mathbf{P}}^{(t)}[:, 0{:}h', 0{:}w']^{(t)};\, h \times w,\, \texttt{bilinear}\Big)$ /* Crop and restore to $h \times w$. */

$\alpha \leftarrow \dfrac{L_{sam}}{\max(h, w)};\ h' \leftarrow \lfloor h\alpha + 0.5 \rfloor,\ w' \leftarrow \lfloor w\alpha + 0.5 \rfloor$

$\tilde{\mathbf{P}}^{(t)} \leftarrow \Big(\mathrm{pad}(\cdot;\, (0,\, L_{sam} - w',\, 0,\, L_{sam} - h')) \circ \mathrm{interp}(\cdot;\, h' \times w',\, \texttt{bilinear})\Big)\Big(\tilde{\mathbf{P}}^{(t)}\Big)$

/* Resize long side to L_{sam}, then right/bottom zero-pad to $L_{sam} \times L_{sam}$. */

$\mathbf{P}^{(t)} \leftarrow f_\theta\Big(\tilde{\mathbf{P}}^{(t)}\Big),\ f_\theta : \mathbb{R}^{1\times L_{sam}\times L_{sam}} \to \mathbb{R}^{C\times H\times W}$

$\tilde{\mathbf{f}}^{(t)} \leftarrow \mathbf{f} \oplus \mathbf{P}^{(t)}$

end for

return $\{\tilde{\mathbf{f}}^{(t)}\}_{t=1}^{N_{seg}}$

Table 10: Training recipe of AnchorSeg for different benchmarks.

Benchmarks	Configuration	Model	
		AnchorSeg-LLaVA1.5-7B	AnchorSeg-LLaVA1.5-13B
Reasoning Segmentation	dataset	`reason_seg:refer_seg:sem_seg`=1:1:1	
	sem_seg_data	`ade20k||cocostuff||pascal_part||mapillary`	
	refer_seg_data	`refclef||refcoco||refcoco+||refcocog||refzom||grefcoco`	
	epochs / steps_per_epoch	120 / 10k	30 / 10k
	grad_accumulation_steps	10	
	optimizer/lr	AdamW/ 3×10^{-4}	AdamW/1×10^{-4}
	betas / warmup_num_steps	(0.9, 0.95) / 100 steps	
	lora_r	8	64
	$\lambda_{bce}/\lambda_{dice}$	2.0/4.0	2.0/4.0
	zero_stage	2	
	num_classes_per_sample	3	
	batch_size	2 samples / GPU	
	weight_decay	0.00	
	gradient_clipping	1.0	
Referring Expression Comprehension (REC)	dataset	`refer_seg:sem_seg:neg_refer_seg:correct_refer_seg`=1:1:1:1	
	refer_seg_data	`refclef||refcoco||refcoco+||refcocog||refzom||grefcoco`	
	sem_seg_data	`ade20k||cocostuff||pascal_part||mapillary`	
	neg_refer_seg_data	`R-refcoco||R-refcoco+||R-refcocog`	
	correct_refer_seg_data	`fprefcoco||fprefcoco+||fprefcocog`	
	epochs / Steps	30 / 10k	
	grad_accumulation_steps	10	
	optimizer/lr	AdamW/ 3×10^{-4}	AdamW/1×10^{-4}
	lora_r	8	64
	$\lambda_{bce}/\lambda_{dice}$	2.0/4.0	2.0/4.0
	zero_stage	2	
	num_classes_per_sample	3	
	batch_size	2 samples / GPU	
	weight_decay	0.00	
	gradient_clipping	1.0	
Generalized Referring Expression Segmentation (GRES)	dataset	`refer_seg`	
	refer_seg_data	`refzom||grefcoco`	
	epochs / steps	30 / 10k	
	grad_accumulation_steps	10	
	optimizer/lr	AdamW/ 3×10^{-4}	AdamW/1×10^{-4}
	lora_r	8	64
	$\lambda_{bce}/\lambda_{dice}$	2.0/4.0	2.0/4.0
	zero_stage	2	
	num_classes_per_sample	3	
	batch_size	2 samples / GPU	
	weight_decay	0.00	
	gradient_clipping	1.0	