DGSeg: Dynamic Gating of Semantic–Spatial Guided Predictions for Reasoning Segmentation

Ruizhe Zeng[1,2], Siyu Cao[1,2], Lu Zhang[1,2✉], and Zhi-yong Liu[1,2,3✉]

[1] State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
{zengruizhe2022, caosiyu2024, lu.zhang, zhiyong.liu}@ia.ac.cn
[2] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China
[3] Nanjing Artificial Intelligence Research of IA, Nanjing, 211100, China

Abstract. Reasoning segmentation aims to predict pixel-wise masks for targets given complex language queries. Existing approaches leverage Multimodal Large Language Models (MLLMs) for vision-language reasoning and generate intermediate target cues (e.g., points or boxes) to guide a segmentation model. However, compressing rich reasoning into sparse cues often introduces ambiguity and noise, preventing these cues from accurately preserving the reasoning intent. While multiple complementary cues can enrich target information, existing methods typically feed them jointly into a single segmentation process, allowing ambiguous or erroneous cues to affect the entire prediction. Therefore, we propose **DGSeg**, a reasoning segmentation framework that learns to fuse predictions guided by semantic and spatial cues. Specifically, the MLLM jointly reasons about both target identity and spatial location, producing complementary semantic and spatial cues that are fed into separate segmentation branches. Their predictions are adaptively integrated by a lightweight dynamic gating module trained with relative branch-quality supervision to suppress noisy or conflicting regions. Extensive experiments demonstrate that DGSeg consistently outperforms strong baselines on multiple benchmarks and achieves 69.6% and 67.3% gIoU on the challenging ReasonSeg validation and test splits.

Keywords: Reasoning Segmentation · Promptable Segmentation · Learnable Fusion

1 Introduction

Reasoning segmentation is the task of segmenting objects in an image given an implicit language query [26]. For instance, for "find a tool that can tighten screws", the model must reason about the functionality of entities in the scene, identify a suitable object (e.g., a screwdriver), and precisely segment it. Therefore, this task is crucial for vision systems that need to translate high-level human

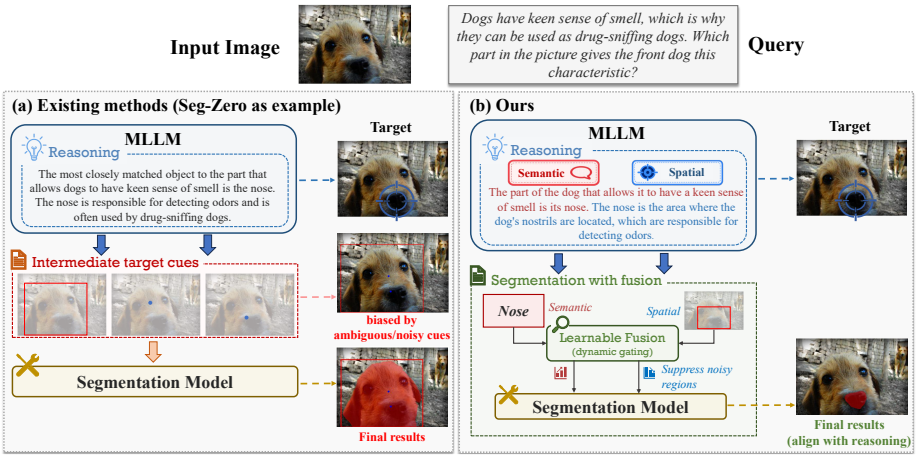


Fig. 1: Motivation of our method. Existing approaches (e.g., Seg-Zero [33]) use an MLLM to generate target cues for downstream segmentation. However, they feed cues directly without mitigating potential noise, causing results to deviate from reasoning due to ambiguous or noisy cues (e.g., oversized boxes or erroneous points). In contrast, we leverage complementary semantic–spatial reasoning and a learnable fusion process to resolve potential noise, helping the final results better align with the reasoning intent.

intent into object regions, especially in embodied settings where downstream interaction or manipulation depends on precise localization [12, 46].

While Multimodal Large Language Models (MLLMs) [2, 3, 10, 31] exhibit remarkable reasoning capabilities, they struggle with fine-grained visual localization tasks [17, 49, 53]. To leverage their ability to analyze complex queries while addressing this limitation, current methods typically combine MLLMs with promptable segmentation models, and bridge high-level MLLM reasoning and the prompt space of the segmentation model by generating intermediate target cues (e.g., latent token embeddings [4, 26, 38], points, or bounding boxes [33, 40, 54]). However, compressing rich reasoning into such target cues may degrade referential details and make the cues ambiguous or noisy, preventing them from accurately preserving the reasoning intent. As shown in Fig. 1, when the target is a dog's nose, the bounding box generated by the MLLM may loosely cover the nose while also including irrelevant regions (e.g., the entire dog), and the segmentation model may take these regions as valid targets. Although multiple complementary cues can enrich target information, current methods typically encode them into a unified representation for segmentation, where any ambiguous or erroneous cue may introduce noise into the whole segmentation process. While some recent works [15, 18] have attempted to iteratively refine these cues using MLLMs, they often require prohibitive costs in both training and inference. Thus, developing an efficient approach to reduce target referential bias caused by ambiguous or noisy cues remains a critical challenge.

To address these challenges, we propose a novel reasoning segmentation pipeline that builds learnable adaptive fusion upon separate processing of complementary target cues. Unlike existing methods that directly encode all cues into a unified segmentation process, our core insight is to isolate potential noise by independently processing complementary cues, and to guide adaptive fusion with supervision derived from the relative segmentation quality of different branches. In this way, the pipeline integrates effective information from different cues while mitigating referential bias and preserving the original reasoning intent.

Based on this pipeline, we instantiate our framework as **DGSeg**, where dynamic gating serves as a lightweight implementation of the learnable fusion mechanism. The classic view posits that visual understanding amounts to knowing *what is where* [37], suggesting that object representation requires both semantic identity and spatial localization. Following this insight, DGSeg guides the MLLM to jointly reason about *what* the target is and *where* it is, producing an explicit textual description and a bounding-box level localization as complementary semantic and spatial cues. These cues are processed by separate semantic and spatial segmentation branches, respectively. Finally, the dynamic gating module aggregates their predictions using pixel-wise fusion weights estimated from branch features.

Extensive experiments on referring and reasoning segmentation benchmarks demonstrate that DGSeg consistently surpasses a wide range of strong baselines. Under the zero-shot setting, it achieves 66.0% and 60.0% gIoU with the 3B-scale model and 69.6% and 67.3% gIoU with the 7B-scale model on the challenging ReasonSeg validation and test splits [26]. Ablation studies further confirm the effectiveness of the proposed pipeline, showing that both semantic-spatial cue generation and learned fusion bring consistent performance gains. Moreover, fusion behavior analysis indicates that the learned fusion module mitigates relative noise across dual branches by adaptively assigning higher weights to more accurate predictions, making the final segmentation results more consistent with the MLLM's intended reasoning.

Overall, our main contributions are summarized as follows:

- We propose a new reasoning segmentation pipeline that learns to adaptively fuse predictions guided by complementary target cues, thereby reducing the impact of ambiguous or erroneous cues and better preserving the original reasoning intent.
- We instantiate this pipeline as DGSeg, which generates semantic and spatial cues from the MLLM, processes them with separate segmentation branches, and aggregates their predictions using a lightweight dynamic gating module.
- Extensive experiments across referring and reasoning segmentation benchmarks show consistent gains over strong baselines, achieving up to 69.6% and 67.3% gIoU on the challenging ReasonSeg validation and test splits. Ablation studies and further analyses validate the effectiveness of the proposed pipeline and DGSeg.

2 Related Works

2.1 Reasoning Segmentation

Reasoning segmentation is the task that focuses on extracting the implicit referential intent in a query and grounding it into a pixel-level segmentation mask. Existing methods typically leverage an MLLM to perform vision-language reasoning and then pass the inferred target cues to a downstream segmentation model. These methods can be broadly categorized into three categories.

The first category uses embeddings of special tokens to prompt the downstream segmentation model. LISA [26] introduces a special `<SEG>` token and uses its contextual embedding from the MLLM reasoning process to condition SAM for mask prediction. CoReS [4] follows a dual-chain pipeline that first coarsely localizes the target region and then refines it into a fine-grained segmentation mask. The second line exploits intermediate signals produced during MLLM reasoning (e.g., vision-language attention maps) to extract referential information. For example, Kang et al. [23] propose a training-free reasoning segmentation method by devising hand-crafted rules to extract localization cues from attention maps. LENS [32] further introduces a plug-and-play framework that attaches a lightweight segmentation head to refine spatial information in attention maps. The third line leverages the direct textual output of MLLM reasoning as prompts to condition the downstream segmentation model. These methods typically finetune the MLLM via supervised finetuning (SFT) [8] or reinforcement learning (RL) [33, 34, 54, 57] based on the quality of generated textual outputs, while keeping the downstream segmentation model frozen.

Despite their differences, these methods often rely on a single type of target cue and directly use it to condition mask prediction. As a result, the referent can remain ambiguous or affected by noisy cues, leading to ambiguity and error propagation in the final masks.

2.2 Multimodal Large Language Model

Early research on vision-language foundation models [22, 27, 39] primarily focused on aligning image features with language for a wide range of downstream perception tasks. Recently, extending the language understanding and reasoning capabilities of LLMs [1, 45] to multimodal inputs has offered a new paradigm for tackling vision-language problems. Representative works like LLaVA [31] show that treating visual features as tokens and finetuning on high-quality multimodal instruction data can effectively elicit generalizable vision-language capabilities. Recent open-source MLLMs [3, 7, 9, 13, 47] further strengthen recognition, grounding, document understanding, and long-context multimodal reasoning through improved pretraining data, post-training recipes, and scaling strategies. For fine-grained output tasks such as segmentation, existing methods [20, 26, 51] commonly couple MLLMs with specialized perception models to leverage the strong reasoning capabilities of MLLMs. In this paradigm, a key challenge is to obtain target cues that are both complete and reliable for guiding the downstream segmentation model, which directly motivates our method.

2.3 Promptable Segmentation

Promptable segmentation predicts segmentation masks conditioned on flexible user prompts. A representative work in this line is SAM [25], which supports sparse geometric prompts such as points and bounding boxes, and dense prompts in the form of coarse masks. It encodes the image and prompts separately and then produces the final segmentation via a mask decoder. Trained on the large-scale SA-1B dataset [25], SAM demonstrates strong zero-shot segmentation capability and excellent fine-grained visual perception. Subsequent work, SAM2 [40], extends promptable segmentation to video object tracking, while SAM3 [5] introduces a DETR-based detector [6] to detect and segment all visual elements referred to by textual queries or visual exemplars.

Despite their strong generalization, promptable segmentation models are sensitive to the quality of input prompts. Stable-SAM [16] reveals that the mask decoder tends to produce biased feature activations given low-quality prompts and presents a novel deformable sampling plugin to improve prompt stability. SAMRefiner [29] introduces a prompt excavation strategy to mine diverse and accurate prompts from coarse masks for segmentation refinement. However, most existing reasoning segmentation approaches overlook the adverse impact of prompt quality, making their predictions susceptible to noise. Although iterative MLLM refinement has been explored to reduce noisy prompts [15, 59], it is computationally expensive and remains vulnerable to hallucinated cues. We therefore adopt a learnable fusion module to evaluate pixel-level features from the segmentation process and integrate dual-branch predictions.

3 Method

In this section, we present DGSeg as a concrete implementation of the proposed pipeline, which independently processes complementary cues and learns to adaptively fuse their predictions. We first describe how the MLLM generates semantic and spatial cues. We then introduce the dual-branch segmentation design and the dynamic gating module for adaptive prediction fusion. Finally, we detail the two-stage optimization strategy for cue generation and fusion learning.

3.1 Architecture of DGSeg

Framework Overview. Our framework is illustrated in Fig. 2, which consists of a reasoning model as the target cue generator and a downstream segmentation model as the cue executor. Given an input RGB image $I \in \mathbb{R}^{H\times W\times 3}$ and a language query T, the reasoning model performs high-level reasoning to produce semantic–spatial cues. The segmentation model then processes these cues through two branches and employs a dynamic gating module to fuse predictions.
Reasoning Model. Given the image–query pair $\{I, T\}$, we instruct the reasoning model F_{reason} to infer the intended target from two complementary aspects: semantic identity and spatial localization. Inspired by chain-of-thought prompting [50], which encourages models to articulate intermediate reasoning steps

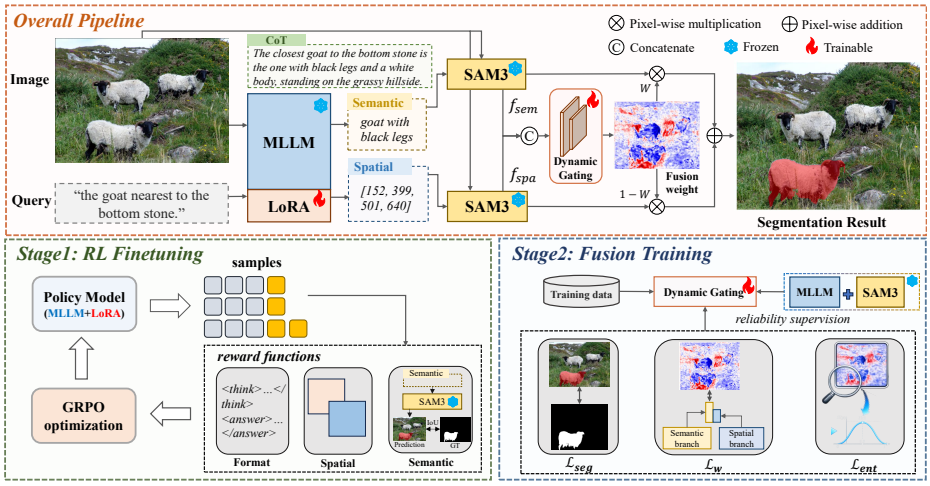


Fig. 2: Overall framework of DGSeg. Given an image and a language query, the MLLM first generates complementary semantic and spatial cues. These cues are processed by separate segmentation branches to produce initial results. A learnable dynamic gating module evaluates the features from dual-branch segmentation and adaptively fuses these results.

before producing final outputs, we guide F_{reason} to first generate an explicit reasoning trace c_{CoT}. The model then outputs a concise textual description c_{sem} and a bounding-box-level localization c_{spa} for the intended target, i.e.,

$$c_{\text{CoT}}, c_{\text{sem}}, c_{\text{spa}} = F_{\text{reason}}(I, T). \tag{1}$$

Compared with prior approaches that rely solely on geometric prompts such as points, bounding boxes [21, 33, 54] or latent embeddings of special tokens [26, 56], our formulation characterizes the target from both semantic and spatial perspectives and provides a more complete target representation for downstream segmentation.

Segmentation Model. We adopt SAM3 [5] as the segmentation model F_{seg} since it supports both semantic and spatial inputs. In the original SAM3 framework, multiple prompts are encoded jointly into a single unified prompt representation, which allows unreliable prompts from any source to affect the entire prediction. To mitigate this issue, DGSeg processes the semantic cue c_{sem} and spatial cue c_{spa} via separate branches to produce mask logits $\ell_{\text{sem}}, \ell_{\text{spa}} \in \mathbb{R}^{H\times W}$ and the corresponding mask predictions m_{sem} and m_{spa}.

Dynamic Gating Module. After the semantic and spatial branches produce their predictions, we introduce a lightweight dynamic gating module F_{dg} to adaptively estimate pixel-wise fusion weights for fusing the dual-branch outputs. The input to F_{dg} is the feature embedding $f \in \mathbb{R}^{1\times c\times h\times w}$ from the pixel decoder of SAM3 [5], which encodes the joint representation of the image and the cues and preserves informative segmentation patterns such as boundary structures

and high-response regions. Then, we concatenate the pixel embeddings f_{sem} and f_{spa} from the semantic and spatial branches, and feed them into a lightweight network composed of two convolutional layers to jointly evaluate the features from the two branches and predict a pixel-wise fusion weight map W:

$$W = \sigma(F_{\text{dg}}(\text{Concate}(f_{\text{sem}}, f_{\text{spa}}))) \in [0,1]^{1\times h\times w}, \tag{2}$$

where σ denotes the sigmoid activation function, and h and w denote the spatial resolution of the feature embeddings.

We then resize W to match the spatial resolution of the logit maps (denoted as $W^{\uparrow}$), and compute the final logit and mask prediction $\{\ell, m\}$ by adaptively combining the logits from the two branches:

$$\ell = W^{\uparrow} \odot \ell_{\text{sem}} + (1 - W^{\uparrow}) \odot \ell_{\text{spa}}, \tag{3}$$

$$m = \mathbb{I}[\ell > 0]\,, \tag{4}$$

where $\odot$ denotes element-wise multiplication and $\mathbb{I}[\cdot]$ is the indicator function.

3.2 Training Strategy

Strategy Overview. We introduce a two-stage training strategy to optimize DGSeg effectively across both cue generation and mask prediction. In the first stage, we freeze the segmentation model and finetune the MLLM via reinforcement learning to produce accurate semantic and spatial cues. In the second stage, we freeze the MLLM and the segmentation backbone to train the dynamic gating module specifically through supervised learning. This separated design ensures stable supervision for adaptively fusing dual-branch predictions and prevents interference between optimizing the cue generator and the segmentation executor.
Stage 1: RL Finetuning. In this stage, we finetune the MLLM with GRPO [43] to improve target cue generation while preserving generalization. Our objective is to ensure the reasoning outputs are compatible with the segmentation model and reliable for identifying the intended target. Specifically, the MLLM is encouraged to produce structured outputs that can be parsed into extractable cues which can accurately refer to the target. To this end, we optimize a composite reward comprising three terms, including a format reward for valid structured outputs and semantic and spatial rewards that assess cue reliability. We detail each reward component below.

- **Format reward.** We require the MLLM to place its explicit reasoning process within `<think></think>` tags and the final response within `<answer></answer>` tags so that the target cues can be reliably parsed. Let the MLLM output be y, we define

$$\mathcal{R}_{\text{format}} = \begin{cases} 1, & \text{if } y \text{ follows the predefined output format,} \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

- **Spatial reward.** To encourage accurate coarse localization, we use bounding-box IoU as the reward. Given the predicted bounding box target cue c_{spa} and the ground-truth box b^*, we define

$$\mathcal{R}_{\text{spatial}} = \text{IoU}(c_{\text{spa}}, b^*) = \frac{|c_{\text{spa}} \cap b^*|}{|c_{\text{spa}} \cup b^*|}. \tag{6}$$

- **Semantic reward.** In image classification, semantic correctness is often evaluated by matching a predicted category to a ground truth label [14]. However, category labels are costly to annotate and may not be well-aligned with the semantic prompt space that the segmentation model operates on. Inspired by human feedback alignment [11], we define a semantic reward based on the mask preference of the segmentation model. Specifically, we use the extracted semantic cue c_{sem} to obtain the prediction m_{sem} from the semantic branch. The semantic reward is then defined as the IoU between m_{sem} and the ground truth m^*:

$$\mathcal{R}_{\text{semantic}} = \text{IoU}(m_{\text{sem}}, m^*) = \frac{|m_{\text{sem}} \cap m^*|}{|m_{\text{sem}} \cup m^*|}. \tag{7}$$

Stage 2: Fusion Training. In this stage, we freeze the MLLM and the segmentation backbone to train only the dynamic gating module F_{dg} for optimal prediction fusion. Since the final segmentation outcome is jointly determined by cue quality and gating weights, optimizing segmentation loss $\mathcal{L}_{\text{seg}}$ alone provides ambiguous supervision for dual-branch fusion. We therefore add an auxiliary target that encourages the gate to favor the branch that is closer to the ground truth. Specifically, given the two branch masks m_{sem} and m_{spa} and the ground-truth mask m^*, we compute

$$s_{\text{sem}} = \text{IoU}(m_{\text{sem}}, m^*), \qquad s_{\text{spa}} = \text{IoU}(m_{\text{spa}}, m^*), \tag{8}$$

and derive a soft fusion target weight as

$$W^* = \left(\frac{\exp(s_{\text{sem}}/\tau)}{\exp(s_{\text{sem}}/\tau) + \exp(s_{\text{spa}}/\tau)} \right)^{H \times W}, \tag{9}$$

where τ controls the sharpness. We take W^* to supervise the predicted weights $W^{\uparrow}$ with a binary cross-entropy loss $\mathcal{L}_{\text{w}}$, and an entropy penalty $\mathcal{L}_{\text{ent}}$ to discourage overly averaged weights:

$$\mathcal{L}_{\text{ent}} = -W^{\uparrow} \log(W^{\uparrow}) - (1 - W^{\uparrow}) \log(1 - W^{\uparrow}). \tag{10}$$

Finally, the overall training objective is

$$\mathcal{L} = \mathcal{L}_{\text{seg}} + \lambda_{\text{w}} \mathcal{L}_{\text{w}} + \lambda_{\text{ent}} \mathcal{L}_{\text{ent}}. \tag{11}$$

During training, we anneal λ_{w} over iterations t to provide strong guidance early while avoiding convergence to the soft target. Finally, we emphasize pixels where fusion matters most by reweighting the loss, assigning higher weights to pixels where the two branches disagree compared to those with consistent predictions. Detailed configurations for these weight hyperparameters are provided in the supplementary material.

Table 1: Performance comparison on ReasonSeg benchmark under the **zero-shot setting**.

Method	MLLM	Val		Test	
		gIoU	cIoU	gIoU	cIoU
OVSeg [28]	–	28.5	18.6	26.1	20.8
SEEM [60]	–	25.5	21.2	24.3	18.7
Grounded-SAM [41]	–	26.0	14.5	21.3	16.4
Seg-Zero [33]	Qwen2.5-VL-3B	62.6	58.5	56.1	48.6
Seg-R1 [54]	Qwen2.5-VL-3B	60.8	56.2	55.3	46.6
RISE [57]	Qwen2.5-VL-3B	52.6	43.1	52.9	45.1
PIXELTHINK [48]	Qwen2.5-VL-3B	62.3	58.5	58.8	52.1
CoPRS [36]	Qwen2.5-VL-3B	61.3	**60.6**	57.8	**52.7**
DGSeg	Qwen2.5-VL-3B	**66.0**	58.3	**60.0**	50.3
LISA [26]	LLaVA1.5-7B	53.6	52.3	48.7	48.8
SAM4MLLM [8]	Qwen-VL-7B	46.1	48.1	–	–
RSVP [35]	Qwen2-VL-7B	58.6	48.5	56.6	51.6
Seg-Zero [33]	Qwen2.5-VL-7B	62.6	62.0	57.5	52.0
SAM-R1 [21]	Qwen2.5-VL-7B	54.0	55.8	60.2	54.3
Seg-R1 [54]	Qwen2.5-VL-7B	58.6	41.2	56.7	53.7
RISE [57]	Qwen2.5-VL-7B	62.0	55.3	58.7	52.5
PIXELTHINK [48]	Qwen2.5-VL-7B	63.8	62.7	60.2	55.8
CoPRS [36]	Qwen2.5-VL-7B	65.2	64.5	59.8	55.1
SAM-Veteran [15]	Qwen2.5-VL-7B	68.2	**67.3**	62.6	56.1
SAM3 Agent [5]	Qwen2.5-VL-7B	65.4	50.5	62.6	56.2
DGSeg	Qwen2.5-VL-7B	**69.6**	65.5	**67.3**	**63.6**

4 Experiment

4.1 Experimental Setup

Implementation Details. We use Qwen2.5-VL [3] as the reasoning model and SAM3 [5] as the segmentation backbone. In Stage 1, we finetune the MLLM with GRPO [43] using a per-GPU batch size of 8 and sample each training instance 8 times. Furthermore, we finetune the MLLM using LoRA [19], and the number of trainable parameters is substantially fewer than in prior reasoning segmentation approaches [33, 58]. In Stage 2, we freeze the MLLM and SAM3 and train only the dynamic gating module for 3 epochs with a batch size of 1, using a learning rate of 1×10^{-4} and a weight decay of 1×10^{-4}. All experiments are conducted on two NVIDIA H100 GPUs with 80GB memory each. More details can be found in the supplementary material.

Dataset and Evaluation Metrics. We train both stages using 9,000 instances sampled from RefCOCOg [55]. Following [26,33], we evaluate on ReasonSeg [26], reporting both gIoU and cIoU, as well as on RefCOCO [24], RefCOCO+ [55],

Table 2: Performance comparison on RefCOCO, RefCOCO+ and RefCOCOg benchmarks.

Method	MLLM	RefCOCO testA	RefCOCO+ testA	RefCOCOg test	Avg.
LAVT [52]	–	75.8	68.4	62.1	68.8
ReLA [30]	–	76.5	71.0	66.0	71.2
Seg-Zero [33]	Qwen2.5-VL-3B	**79.3**	73.7	71.5	74.8
Seg-R1 [54]	Qwen2.5-VL-3B	76.0	66.8	67.9	70.2
RISE [57]	Qwen2.5-VL-3B	76.5	72.1	68.9	72.5
PIXELTHINK [48]	Qwen2.5-VL-3B	78.7	72.9	72.2	74.6
DGSeg	Qwen2.5-VL-3B	78.9	**74.5**	**72.3**	**75.2**
LISA [26]	LLaVA-7B	76.5	67.4	68.5	70.8
PixelLM [42]	LLaVA-7B	76.5	71.7	70.5	72.9
Seg-Zero [33]	Qwen2.5-VL-7B	80.3	76.2	72.6	76.4
Seg-R1 [54]	Qwen2.5-VL-7B	78.7	70.9	71.4	73.7
SAM-R1 [21]	Qwen2.5-VL-7B	79.2	74.7	73.1	75.7
RISE [57]	Qwen2.5-VL-7B	79.7	**77.7**	73.4	**76.9**
PIXELTHINK [48]	Qwen2.5-VL-7B	79.3	74.8	**73.9**	76.0
SAM-Veteran [15]	Qwen2.5-VL-7B	**80.8**	76.6	73.4	**76.9**
SAM3 Agent [5]	Qwen2.5-VL-7B	58.4	52.2	55.1	55.2
DGSeg	Qwen2.5-VL-7B	80.3	76.4	**73.9**	**76.9**

and RefCOCOg [55], reporting cIoU as the evaluation metric. All experiments on the ReasonSeg dataset follow the zero-shot setting as there are no reasoning segmentation instances used during training. Since cIoU is computed as the ratio between the summed intersection and summed union over all images, it is biased towards large images [26, 44], so we primarily focus on gIoU for ReasonSeg. All results in experiments are in %.

4.2 Main Results

Results on Reasoning Segmentation Benchmarks. Table 1 reports the zero-shot performance on ReasonSeg, where DGSeg demonstrates remarkable effectiveness. With the 3B-scale model, DGSeg achieves **66.0**/**60.0**% gIoU on the validation and test splits. Compared with recent strong baselines built on the same MLLM backbone, DGSeg improves over Seg-Zero [33] by 3.4% and 3.9% gIoU on the validation and test splits, respectively, and surpasses CoPRS [36] by 4.7% and 2.2% gIoU. The advantages of our design are further amplified when scaled to a 7B backbone, where DGSeg reaches **69.6**/**67.3**% gIoU, surpassing SAM-Veteran [15] by 1.4% and 4.7% on the validation and test splits, respectively.

Results on Referring Segmentation Benchmarks. Table 2 reports the performance on the RefCOCO, RefCOCO+ and RefCOCOg datasets. DGSeg

Table 3: Ablation on fusion strategies within the proposed pipeline.

Variant	Val	Test
Oracle fusion	67.9	62.8
Baseline (no fusion)	60.8	55.7
Non-learnable fusion		
Ours + average fusion	63.8	58.2
Ours + confidence-based weighting	64.1	57.2
Learnable fusion		
Ours + learned scalar	64.0	57.8
Ours + MoE	65.6	**60.6**
Ours + dynamic gating	**66.0**	60.0

achieves strong results across all splits, reaching a leading average score of 75.2% with the 3B model and 76.9% with the 7B model. Compared with recent baselines built on the same MLLM backbone, DGSeg improves over Seg-Zero [33] on average. These results demonstrate that DGSeg ensures the final segmentation aligns with the reasoning intent by leveraging semantic-spatial cues and adaptive fusion on dual-branch predictions.

4.3 Ablation Study

We conduct ablations to analyze the effectiveness of our proposed pipeline and DGSeg design. All experiments are evaluated on the model based on Qwen2.5-VL-3B [3], the benchmark is ReasonSeg, and the metric is gIoU. More ablation experimental details and results can be found in the supplementary material.

Ablation on the Proposed Pipeline. We first evaluate the effectiveness of the proposed pipeline by comparing different fusion strategies in Tab. 3. Oracle fusion uses the soft target weight W^* as an upper bound, while the baseline directly encodes semantic and spatial cues without fusion. To further analyze the role of fusion, we compare both non-learnable strategies, including average fusion and confidence-based weighting, and learnable strategies, including learned scalar, MoE, and dynamic gating.

Results show that learnable fusion strategies generally achieve stronger performance than non-learnable ones, demonstrating the effectiveness of learning the fusion process from the training data. By learning the relationship between pixel-level branch features and segmentation quality, these strategies can better assess the relative reliability of different cues and adapt the fusion weights accordingly. Notably, MoE achieves comparable performance to dynamic gating, suggesting that the proposed pipeline can flexibly accommodate advanced learnable fusion strategies. The integration of more advanced fusion methods into our pipeline can be explored for future work.

Analysis of Reward Design. We next analyze the reward design used in the RL finetuning stage and summarize the results in Tab. 4. Here, "label reward"

Table 4: Ablation on the reward design.

Variant	Val	Test
w/o Stage 1	36.0	33.7
w/o format reward	64.6	59.6
w/o semantic reward	62.1	58.5
w/o spatial reward	64.1	59.4
R_{semantic} = label reward	64.7	59.1
Full reward (ours)	**66.0**	**60.0**

Table 5: Ablation on the dual-branch segmentation strategy.

Variant	Val	Test
Baseline (no fusion)	60.8	55.7
Semantic–only	55.3	52.8
Spatial–only	63.4	54.9
Learnable fusion (ours) (dynamic gate)	**66.0**	**60.0**

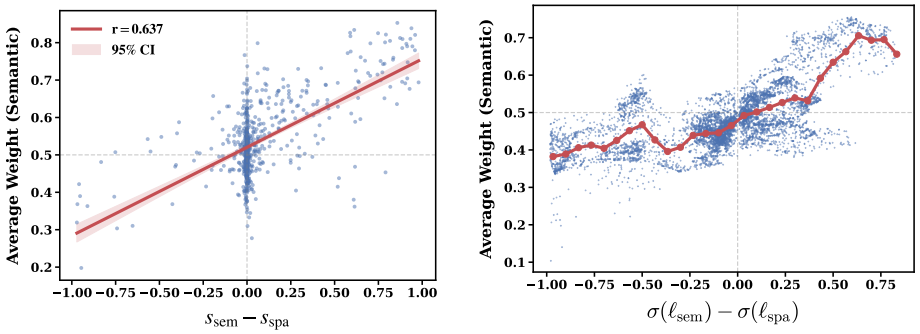
Table 6: Analysis of computational efficiency. We report peak GPU memory usage, floating-point operations (FLOPs), and frames per second (FPS) for the no-fusion baseline and DGSeg, measured on a single NVIDIA H100 GPU.

Method	Peak Memory (GB) ↓	FLOPs (GFLOPs) ↓	FPS ↑
Baseline (no fusion)	11.75	7375.7	0.280
DGSeg	12.24 (+**4.2%**)	7395.1 (+**0.3%**)	0.275 (**-2.0%**)

uses category labels as the category label supervision, while the semantic reward in DGSeg is based on the segmentation model preference. The results indicate that DGSeg benefits from jointly optimizing the output format and the quality of the generated cues. Specifically, the preference optimization based on SAM3 yields better results when applied to semantic cues rather than directly to category labels. This suggests that, to tightly couple the reasoning model with the segmentation model, it is more important to generate semantic cues that are better aligned with the prompt space of the segmentation model.

Analysis of Dual-Branch Segmentation Design. We then analyze the dual-branch segmentation design in Tab. 5. The baseline with no fusion follows the standard prompting scheme of SAM3 by encoding semantic and spatial cues jointly in a single forward pass. To examine the contribution of each cue type, we further report semantic-only and spatial-only variants, which perform segmentation using only the semantic or spatial branch, respectively. Results indicate that joint encoding can allow noise from any cue to contaminate the prediction, leading to performance that can even be worse than using the spatial cue alone. In contrast, DGSeg evaluates the pixel-level features from dual branches and suppresses noisy regions, yielding the best overall performance.

Additionally, we analyze the computational overhead introduced by the dual-branch architecture. As shown in Tab. 6, our proposed components incur only a 0.3% increase in FLOPs and a 2.0% drop in FPS, while effectively boosting the overall reasoning segmentation performance.



(a) The relationship between IoU difference of the semantic and spatial branches and the average fusion weights of the semantic branch. "CI" means Confidence Interval.

(b) The relationship between sigmoid logit difference of the semantic and spatial branches and the average fusion weights of the semantic branch at the pixel level.

Fig. 3: Visualization of the analysis on the dynamic gating module.

Correlation Analysis of Fusion Weights. We further examine whether the proposed learnable fusion module in DGSeg effectively assesses segmentation features and filters unreliable regions from both global and local perspectives.

For global correlation, we compute the IoU of the semantic and spatial branch predictions with the ground truth mask, and use their IoU difference $s_{\text{sem}} - s_{\text{spa}}$ as a proxy for relative branch quality, where a larger gap indicates that the predictions of the semantic branch are closer to ground truth than those of the spatial branch. We also compute the average fusion weight over foreground regions where fusion is more meaningful. As shown in Fig. 3a, the IoU difference exhibits a strong positive correlation ($r = 0.637$) with the predicted fusion weight. This quantitative relationship indicates that by learning to analyze joint feature representations of the image and the cues, the module can effectively identify the superior branch and assign appropriate fusion weights.

For local correlation, we analyze foreground pixels by randomly sampling those where the predictions from the two branches are inconsistent (i.e., exhibiting a sign mismatch in logits), and we correlate the fusion weights with the sigmoid logit difference. Figure 3b shows that when semantic evidence is stronger (positive logit difference), the predicted weight tends to favor the semantic branch, and vice versa. This confirms that by learning to evaluate the features from the segmentation branches, the dynamic gating module evaluates the distribution of local high response regions and boundary structures, enabling it to perform effective local denoising.

4.4 Qualitative Analysis

Visualization of Gating Results. Figure 4 presents qualitative comparisons between the two branches and the final results. As shown, relying solely on either semantic or spatial cues often results in ambiguous or even erroneous target

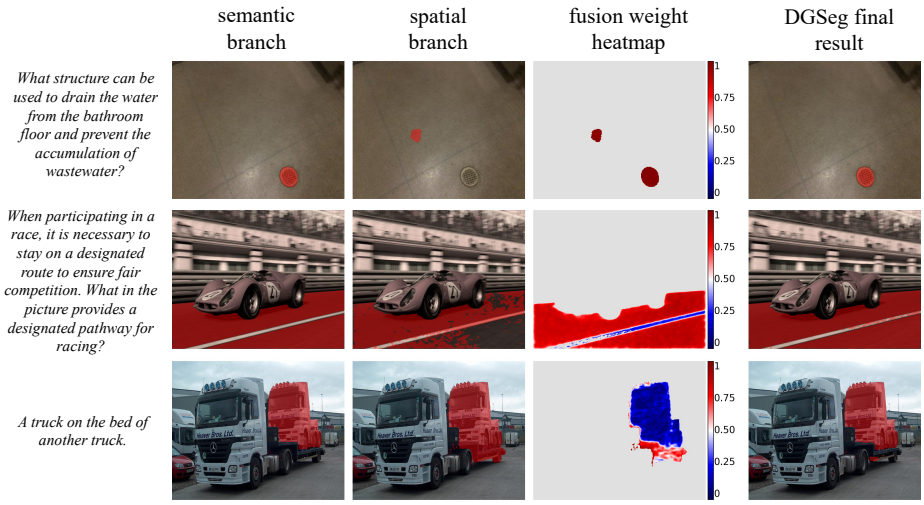


Fig. 4: Visualization of gating results in DGSeg. For each example, we show the semantic and spatial branch predictions, final results after dynamic gating, and the foreground fusion weights of the semantic branch. Red favors the semantic branch, and blue favors the spatial branch. We set all background regions to gray for better visualization.

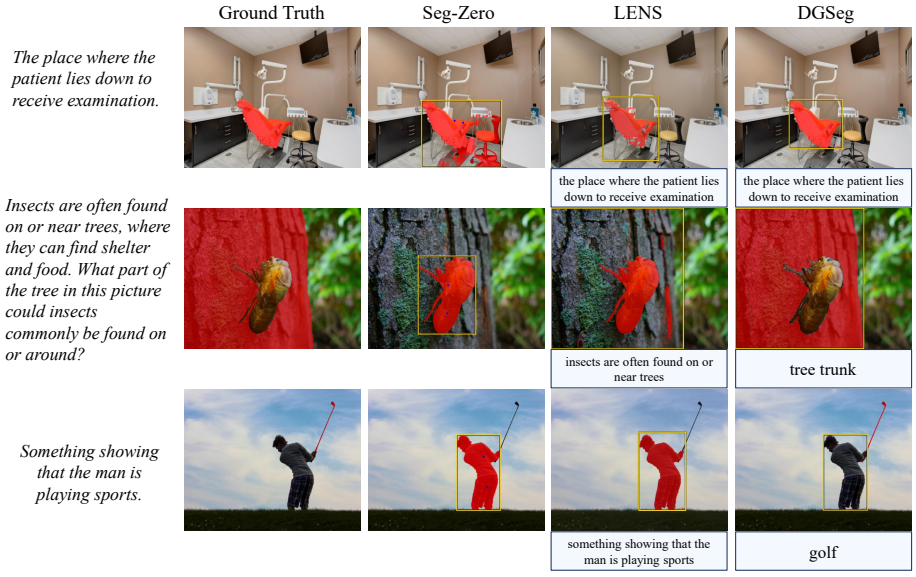


Fig. 5: Qualitative comparison on ReasonSeg. We compare strong baselines including Seg-Zero [33] and LENS [58]. For each example, the left shows the query and the right visualizes the predictions and the generated target cues. DGSeg produces semantic-spatial cues and performs adaptive fusion, leading to better performance even when some cues are ambiguous or noisy.

references. Consequently, the segmentation model is prone to producing incorrect areas within the mask. In contrast, our proposed dynamic gating module effectively identifies unreliable regions and adaptively adjusts the weights of the two branches. For example, in the bottom case, it detects the erroneous segmentation of the larger truck from the spatial branch and increases the weight of the semantic branch in that area.

Qualitative Comparisons. Figure 5 presents visual comparisons between DGSeg-3B and representative baselines including Seg-Zero [33] and LENS [58] on challenging cases of ReasonSeg [26]. Seg-Zero provides purely spatial prompts (a bounding box and two points) to guide the downstream segmentation model, which can lead to ambiguous referents and irrelevant regions. For example, the predicted mask in the first row includes unrelated objects due to the oversized bounding box. While LENS also shares a strategy of combining semantic and spatial information and has been finetuned on the ReasonSeg training split, it lacks a mechanism to assess target cues and is highly susceptible to erroneous information (evidenced by the failure case in the second row). In contrast, DGSeg adaptively suppresses noisy branches and remains robust even when some target cues are ambiguous or noisy (e.g., the spatial cue fails to cover the golf).

5 Conclusion

In this paper, we propose a novel reasoning segmentation pipeline based on complementary cue processing and learnable adaptive fusion, aiming to reduce the impact of ambiguous or erroneous cues in MLLM-guided segmentation. We instantiate this pipeline as DGSeg, which generates semantic and spatial cues from the MLLM, processes them with separate segmentation branches, and aggregates their predictions using a lightweight dynamic gating module. Extensive experiments on reasoning and referring segmentation benchmarks demonstrate the effectiveness of DGSeg, while ablation studies further validate the benefit of the proposed pipeline and its learnable fusion process. Our analysis also shows that the pipeline can accommodate different advanced fusion strategies, suggesting a promising direction for future exploration of more effective fusion designs in reasoning segmentation.

Acknowledgements

This research is supported by the National Key Research and Development Program of China under Grant No. 2025YFE0216600 and Key Science & Technology Project of Anhui Province under Grant No. 202523j08050027.

References

1. Bai, J., Bai, S., Chu, Y., Cui, Z., Dang, K., Deng, X., Fan, Y., Ge, W., Han, Y., Huang, F., et al.: Qwen technical report. arXiv preprint arXiv:2309.16609 (2023)

2. Bai, J., Bai, S., Yang, S., Wang, S., Tan, S., Wang, P., Lin, J., Zhou, C., Zhou, J.: Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. arXiv preprint arXiv:2308.12966 (2023)
3. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., et al.: Qwen2.5-vl technical report. arXiv preprint arXiv:2502.13923 (2025)
4. Bao, X., Sun, S., Ma, S., Zheng, K., Guo, Y., Zhao, G., Zheng, Y., Wang, X.: Cores: Orchestrating the dance of reasoning and segmentation. In: ECCV (2024)
5. Carion, N., Gustafson, L., Hu, Y.T., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K.V., Khedr, H., Huang, A., et al.: Sam 3: Segment anything with concepts. arXiv preprint arXiv:2511.16719 (2025)
6. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: ECCV (2020)
7. Chen, B., Xu, Z., Kirmani, S., Ichter, B., Sadigh, D., Guibas, L., Xia, F.: Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In: CVPR (2024)
8. Chen, Y.C., Li, W.H., Sun, C., Wang, Y.C.F., Chen, C.S.: Sam4mllm: Enhance multi-modal large language model for referring expression segmentation. In: ECCV (2024)
9. Chen, Z., Wang, W., Cao, Y., Liu, Y., Gao, Z., Cui, E., Zhu, J., Ye, S., Tian, H., Liu, Z., et al.: Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. arXiv preprint arXiv:2412.05271 (2024)
10. Chen, Z., Wu, J., Wang, W., Su, W., Chen, G., Xing, S., Zhong, M., Zhang, Q., Zhu, X., Lu, L., et al.: Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In: CVPR (2024)
11. Christiano, P.F., Leike, J., Brown, T.B., Martic, M., Legg, S., Amodei, D.: Deep reinforcement learning from human preferences. In: NeurIPS (2017)
12. Collins, J.A., Houff, C., Tan, Y.L., Kemp, C.C.: Forcesight: Text-guided mobile manipulation with visual-force goals. In: ICRA (2024)
13. Dai, W., Li, J., Li, D., Tiong, A.M.H., Zhao, J., Wang, W., Li, B., Fung, P., Hoi, S.: Instructblip: Towards general-purpose vision-language models with instruction tuning. In: NeurIPS (2023)
14. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009)
15. Du, T., Li, H., Fan, Z., Zhang, J., Pan, P., Zhang, Y.: Sam-veteran: An mllm-based human-like sam agent for reasoning segmentation. In: ICLR (2026)
16. Fan, Q., Tao, X., Ke, L., Ye, M., Zhang, D., Wan, P., Tai, Y.W., Tang, C.K.: Stable segment anything model. In: ICLR (2025)
17. He, H., Li, G., Geng, Z., Xu, J., Peng, Y.: Analyzing and boosting the power of fine-grained visual recognition for multi-modal large language models. In: ICLR (2025)
18. He, X., Zhang, Y., Gao, S., Li, W., Hong, L., Chen, M., Jiang, K., Fu, J., Zhang, W.: Rsagent: Learning to reason and act for text-guided segmentation via multi-turn tool invocations. arXiv preprint arXiv:2512.24023 (2025)
19. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models. In: ICLR (2022)
20. Hu, Y., Stretcu, O., Lu, C.T., Viswanathan, K., Hata, K., Luo, E., Krishna, R., Fuxman, A.: Visual program distillation: Distilling tools and programmatic reasoning into vision-language models. In: CVPR (2024)

21. Huang, J., Xu, Z., Zhou, J., Liu, T., Xiao, Y., Ou, M., Ji, B., Li, X., Yuan, K.: Sam-r1: Leveraging sam for reward feedback in multimodal segmentation via reinforcement learning. In: NeurIPS (2025)
22. Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., V.Le, Q., Sung, Y., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision. In: ICML (2021)
23. Kang, S., Kim, J., Kim, J., Hwang, S.J.: Your large vision-language model only needs a few attention heads for visual grounding. In: CVPR (2025)
24. Kazemzadeh, S., Ordonez, V., Matten, M., Berg, T.L.: Referitgame: Referring to objects in photographs of natural scenes. In: EMNLP (2014)
25. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: ICCV (2023)
26. Lai, X., Tian, Z., Chen, Y., Li, Y., Yuan, Y., Liu, S., Jia, J.: Lisa: Reasoning segmentation via large language model. In: CVPR (2024)
27. Li, J., Selvaraju, R.R., Gotmare, A.D., Joty, S., Xiong, C., Hoi, S.C.H.: Align before fuse: Vision and language representation learning with momentum distillation. In: NeurIPS (2021)
28. Liang, F., Wu, B., Dai, X., Li, K., Zhao, Y., Zhang, H., Zhang, P., Vajda, P., Marculescu, D.: Open-vocabulary semantic segmentation with mask-adapted clip. In: CVPR (2023)
29. Lin, Y., Li, H., Shao, W., Yang, Z., Zhao, J., He, X., Luo, P., Zhang, K.: Samrefiner: Taming segment anything model for universal mask refinement. In: ICLR (2025)
30. Liu, C., Ding, H., Jiang, X.: Gres: Generalized referring expression segmentation. In: CVPR (2023)
31. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. In: NeurIPS (2023)
32. Liu, J., Chen, L.: Segmentation as a plug-and-play capability for frozen multimodal llms. arXiv preprint arXiv:2510.16785 (2025)
33. Liu, Y., Peng, B., Zhong, Z., Yue, Z., Lu, F., Yu, B., Jia, J.: Seg-zero: Reasoning-chain guided segmentation via cognitive reinforcement. arXiv preprint arXiv:2503.06520 (2025)
34. Liu, Y., Qu, T., Zhong, Z., Peng, B., Liu, S., Yu, B., Jia, J.: Visionreasoner: Unified reasoning-integrated visual perception via reinforcement learning. arXiv preprint arXiv:2505.12081 (2025)
35. Lu, Y., Cao, J., Wu, Y., Li, B., Tang, L., Ji, Y., Wu, C., Wu, J., Zhu, W.: Rsvp: Reasoning segmentation via visual prompting and multi-modal chain-of-thought. In: ACL (2025)
36. Lu, Z., Li, L., Wang, J., Feng, Y., Chen, B., Chen, K., Wang, Y.: Coprs: Learning positional prior from chain-of-thought for reasoning segmentation. arXiv preprint arXiv:2510.11173 (2025)
37. Marr, D.: Vision: A computational investigation into the human representation and processing of visual information. MIT press (2010)
38. Qian, R., Yin, X., Dou, D.: Reasoning to attend: Try to understand how< seg> token works. In: CVPR (2025)
39. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML (2021)
40. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., et al.: Sam 2: Segment anything in images and videos. In: ICLR (2025)

41. Ren, T., Liu, S., Zeng, A., Lin, J., Li, K., Cao, H., Chen, J., Huang, X., Chen, Y., Yan, F., et al.: Grounded sam: Assembling open-world models for diverse visual tasks. arXiv preprint arXiv:2401.14159 (2024)
42. Ren, Z., Huang, Z., Wei, Y., Zhao, Y., Fu, D., Feng, J., Jin, X.: Pixellm: Pixel reasoning with large multimodal model. In: CVPR (2024)
43. Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y., Wu, Y., et al.: Deepseekmath: Pushing the limits of mathematical reasoning in open language models. arXiv preprint arXiv:2402.03300 (2024)
44. Shen, Y., Li, C., Xiong, F., Jeong, J.O., Wang, T., Latman, M., Unberath, M.: Reasoning segmentation for images and videos: A survey. arXiv preprint arXiv:2505.18816 (2025)
45. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al.: Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971 (2023)
46. Vuong, A.D., Vu, M.N., Huang, B., Nguyen, N., Le, H., Vo, T., Nguyen, A.: Language-driven grasp detection. In: CVPR (2024)
47. Wang, P., Bai, S., Tan, S., Wang, S., Fan, Z., Bai, J., Chen, K., Liu, X., Wang, J., Ge, W., et al.: Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. arXiv preprint arXiv:2409.12191 (2024)
48. Wang, S., Fang, G., Kong, L., Li, X., Xu, J., Yang, S., Li, Q., Zhu, J., Wang, X.: Pixelthink: Towards efficient chain-of-pixel reasoning. arXiv preprint arXiv:2505.23727 (2025)
49. Wang, Y., Gao, D., Li, B., Long, R., Yi, L., Cai, X., Yang, L., Zhang, J., Yu, S., Xuan, Q.: Cof: Coarse to fine-grained image understanding for multi-modal large language models. In: ICASSP (2025)
50. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E.H., Le, Q.V., Zhou, D., et al.: Chain-of-thought prompting elicits reasoning in large language models. In: NeurIPS (2022)
51. Yan, C., Wang, H., Yan, S., Jiang, X., Hu, Y., Kang, G., Xie, W., Gavves, E.: Visa: Reasoning video object segmentation via large language models. In: ECCV (2024)
52. Yang, Z., Wang, J., Tang, Y., Chen, K., Zhao, H., Torr, P.H.: Lavt: Language-aware vision transformer for referring image segmentation. In: CVPR (2022)
53. Yao, Y., Li, L., Song, J., Chen, C., He, Z., Wang, Y., Wang, X., Gu, T., Li, J., Teng, Y., et al.: Argus inspection: do multimodal large language models possess the eye of panoptes? In: ACMMM (2025)
54. You, Z., Wu, Z.: Seg-r1: Segmentation can be surprisingly simple with reinforcement learning. arXiv preprint arXiv:2506.22624 (2025)
55. Yu, L., Poirson, P., Yang, S., Berg, A.C., Berg, T.L.: Modeling context in referring expressions. In: ECCV (2016)
56. Yuan, H., Li, X., Zhang, T., Sun, Y., Huang, Z., Xu, S., Ji, S., Tong, Y., Qi, L., Feng, J., Ming-Hsuan, Y.: Sa2va: Marrying sam2 with llava for dense grounded understanding of images and videos. arXiv preprint arXiv:2501.04001 (2025)
57. Zhou, Q., Yang, L., Jia, Y., Gao, J., Ni, W., Wu, J., Wang, Q.: Reasoning via implicit self-supervised emergence for instruction segmentation. In: AAAI (2026)
58. Zhu, L., Ouyang, B., Zhang, Y., Cheng, T., Hu, R., Shen, H., Ran, L., Chen, X., Yu, L., Liu, W., Xinggang, W.: Lens: Learning to segment anything with unified reinforced reasoning. In: AAAI (2026)
59. Zhu, M., Tian, Y., Chen, H., Zhou, C., Guo, Q., Liu, Y., Yang, M., Shen, C.: Segagent: Exploring pixel understanding capabilities in mllms by imitating human annotator trajectories. In: CVPR (2025)

60. Zou, X., Yang, J., Zhang, H., Li, F., Li, L., Wang, J., Wang, L., Gao, J., Lee, Y.J.: Segment everything everywhere all at once. In: NeurIPS (2023)