

# From Failure to Feedback: Group Revision Unlocks Hard Cases in Object-Level Grounding (Supplementary Material)

Yuyuan Liu<sup>1</sup> Yiping Ji<sup>2</sup> Anjie Le<sup>1</sup> Jiayuan Zhu<sup>1</sup> Jiazhen Pan<sup>3</sup>  
Can Peng<sup>1</sup> Jiajun Deng<sup>4</sup> Fengbei Liu<sup>5</sup>(✉) Junde Wu<sup>1</sup>

<sup>1</sup> Department of Engineering Science, University of Oxford    <sup>2</sup> Australian Institute for Machine Learning, Adelaide University  
<sup>3</sup> Technical University of Munich    <sup>4</sup> University of Science and Technology of China    <sup>5</sup> Cornell University

## Outline

In this file, we supplement more content from the following aspects to support the findings and experimental results in the main body of the paper:

- **Section A:** Additional implementation details, including the full hyper-parameter configuration and the rollout/-training setup for GRPO (Section A.1), the base criteria used in the reward function (Section A.2), and the prompt specifications for both the single-box and multi-box setups (Section A.3).
- **Section B:** Further dataset details and annotation protocols, including a comparison between RFT and SFT training data requirements.
- **Section C:** Extended experiments that support the results in the main paper, including hallucination evaluation and the full “LLM-as-Judge” prompt (Section C.1), additional gIoU results on RefCOCO (+/g) (Section C.2), object detection results on MS COCO (Section C.3), group-size ablations in GRPO (Section C.4), the impact of different initial/revision group allocations (Section C.5) and Qwen2.5VL official prompt evaluation (Section C.6).
- **Section D:** Mathematical analysis of our Consolidation process, showing that it strictly follows the policy-invariance property [8, 21] in standard reward shaping.
- **Section E:** Additional results on generalisation to other LVLM backbones, including experiments with InternVL3.5-8B across segmentation and counting tasks.
- **Section F:** Test-time inference ablations comparing single-pass inference and revision-based prompting, with analysis of the internalisation of revision behaviour.
- **Section G:** Additional qualitative results and visualisations to complement our quantitative findings.

## A. Implementation Details

In this section, we provide additional implementation details that complement the main paper. Section A.1 outlines the complete hyper parameter configuration used dur-

ing training, Section A.2 formalises the basic reward criteria for GRPO optimisation, and Section A.3 presents the full prompts for both single-box and multi-box settings. All experiments are conducted on 4xNVIDIA H20 GPUs. The peak GPU memory usage is approximately 67.5 GB, and the total training time is around 13.6 hours under our default setting.

### A.1. Hyper-parameter Configuration

Our method is developed on top of the Qwen2.5-VL-7B-Instruct model [1], implemented within the VeRL [30] framework for GRPO-based visual reasoning. We use the officially released pretrained checkpoint and enable gradient checkpointing to reduce memory consumption during training. Each input consists of a question paired with an image, with a maximum prompt length of 3900 tokens and a maximum response length of 2500 tokens.

The actor uses a global batch size of 16, while the step-wise micro-batch size per GPU is set to 2 for both gradient updates and experience collection. The actor applies a maximum gradient norm of 1.0 and incorporates a KL penalty with a coefficient of  $5 \times 10^{-3}$  under the low-variance KL formulation. Optimisation is performed using AdamW [19] with a learning rate of  $1 \times 10^{-6}$  and a weight decay of  $1 \times 10^{-2}$ . Parameter and optimiser offloading are enabled to reduce GPU memory usage, and FSDP is used without CPU offloading or dtype conversion.

For rollout generation, we use a sampling temperature of 1.0 without top-p truncation for both the initial and the revision responses, and we generate 8 candidate responses in each revision group for every prompt. Tensor parallelism is set to 2, and GPU memory utilisation during rollout is constrained to 0.4, as specified in the launch command. The overall training is conducted using 4 GPUs, with all experiments under the vLLM [12] XFormers attention backend.

### A.2. Reward Criteria (from Eq. (7) main paper)

Following prior works [16–18], our basic reward design integrates both **format** criteria, which ensure that the model

produces responses in the expected structural layout, and **accuracy** criteria, which evaluate the object-level grounding accuracy. Below we detail each component of the reward function used during training.

**(1) Structural Format Reward ( $\mathcal{R}_{\text{format}}$ ).** We require responses to follow a fixed XML-style reasoning–answering pattern. A prediction is considered structurally valid only when it contains a `<think>...</think>` block followed by a `<answer>...</answer>` block. If this pattern is satisfied, we assign a structural format reward of 1; otherwise, the reward is 0. Within the `<answer>...</answer>` region, the model must output a JSON array of objects. For each predicted object  $o_k$ , we evaluate two structural fields. First, we assign a box-format reward of 1 if the object includes a `bbox_2d` field consisting of exactly four numerical values, indicating a well-formed bounding box. Second, we assign a point-format reward of 1 if the object includes a `point_2d` field containing exactly two numerical values, indicating a valid 2D point. These checks ensure that each predicted object contains valid box and point fields before any grounding accuracy assessment is performed.

**(2) Accuracy Reward.** For each aligned predicted object and ground-truth annotation, we evaluate grounding accuracy using both bounding-box and point correctness. Let  $\hat{\mathbf{b}}_k \in \mathbb{R}^4$  and  $\hat{\mathbf{p}}_k \in \mathbb{R}^2$  denote the  $k$ -th predicted box and point parsed from the response  $o$ . The corresponding ground-truth box and point from  $y = (\mathbf{b}, \mathbf{p})$  are denoted  $\mathbf{b}_k$  and  $\mathbf{p}_k$ . We compute two rewards based on bounding-box correctness within the responses:

- **IoU reward.** A predicted bounding box is considered correct if its Intersection-over-Union (IoU) with the ground-truth box exceeds 0.5:

$$r_{\text{iou}}^{(k)} = \begin{cases} 1, & \text{if } \text{IoU}(\hat{\mathbf{b}}_k, \mathbf{b}_k) > 0.5, \\ 0, & \text{otherwise.} \end{cases} \quad (10)$$

- **Bounding-box L1 reward.** We also enforce that the L1 distance between the predicted and ground-truth coordinates is less than 10 pixels:

$$r_{\text{bbox}}^{(k)} = \begin{cases} 1, & \text{if } \|\hat{\mathbf{b}}_k - \mathbf{b}_k\|_1 < 10, \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

**For each associated predicted point**, we evaluate its proximity to the ground-truth point as well as whether it lies within the ground-truth bounding box:

$$r_{\text{point}}^{(k)} = \begin{cases} 1, & \text{if } \|\hat{\mathbf{p}}_k - \mathbf{p}_k\|_2 < 30 \text{ and } \hat{\mathbf{p}}_k \in \hat{\mathbf{b}}_k, \\ 0, & \text{otherwise.} \end{cases} \quad (12)$$

Together, for the accuracy components, we compute the object-wise accuracy reward by summing the three sub-rewards:

$$r_{\text{acc}}^{(k)} = r_{\text{iou}}^{(k)} + r_{\text{bbox}}^{(k)} + r_{\text{point}}^{(k)}.$$

Averaging over all matched  $K$  objects yields the overall accuracy reward:

$$\mathcal{R}_{\text{acc}}(o, y) = \frac{1}{K} \sum_{k=1}^K r_{\text{acc}}^{(k)}.$$

**Final Reward.** The total basic reward used during optimization is the sum of format-level and accuracy-level scores:

$$\mathcal{R}(o, y) = \mathcal{R}_{\text{format}}(o) + \mathcal{R}_{\text{acc}}(o, y). \quad (13)$$

In our implementation, the total reward is with the maximum achievable score is 5.

### A.3. Prompt Details

As shown in Fig. A1, we present the prompt specifications for both the single-box and multi-box setups. Both prompts follow a similar linguistic structure, with the primary difference lying in how point annotations are handled, as discussed in Section 4 of the main paper. In the single setup, two representative points are annotated, corresponding to the centers of the two largest inscribed circles within the object. In contrast, the multi setup assigns only one representative point to each referenced object.

## B. Dataset Details

In Table A1, we compare the datasets used in our Reinforcement Fine Tuning (RFT) approach for both the single-box and multi-box setups with the datasets adopted by mainstream Supervised Fine Tuning (SFT) methods. In contrast to prior SFT approaches [13, 24, 27], which depend on large scale segmentation datasets, RFT requires only a very small amount of training data. For instance, LISA [13] is trained on 150K images and 1.2M masks from multiple dense annotation datasets. PerceptionGPT [24] and GSVA [34] also rely on hundreds of thousands of annotated masks. In comparison, our (single) follows the Seg-Zero [17] and uses only 9K annotated samples, while the (multi) follows VisionReasoner [18] and uses only 7K samples. This significant difference in training data scale explains why, in Table 1 and Table 2 of the main paper, RFT based models may achieve slightly lower performance than SFT based methods on certain dataset specific referring segmentation and grounding benchmarks. However, it is important to note that on ReasonSeg [13], our method achieves even stronger results than SFT models, and our average performance across all benchmarks is also higher. These results show that RFT can provide better generalisation while requiring far fewer annotations than traditional SFT pipelines.

## C. Additional Experiments

In this section, we provide extended experimental results that complement the findings reported in the main paper.

Table A1. **Comparison of the training data for referring segmentation and grounding** between Supervised Fine Tuning (SFT) and Reinforcement Fine Tuning (RFT) methods. The SFT approaches rely on large scale segmentation datasets with dense mask annotations, whereas our RFT settings for both single object and multi object training require only a small amount of annotated data.

Method	Train set w/ mask	# img.	# mask
<i>Supervised Fine Tuning Datasets</i>			
LISA [13], SEGLLM [32]	ADE20K [40], COCO-Stuff [4], PACO-LVIS [26], PartImageNet [9], PASCAL-Part [5], refCLEF [29], refCOCO [39], refCOCO+ [39], refCOCOg [20]	150K	1.2M
PerceptionGPT [24]	refCOCO [39], refCOCO+ [39], refCOCOg [20], Visual Genome [11], Flickr30k [38]	150K	3M
GSVA [34]	ADE20K [40], COCO-Stuff [4], PACO-LVIS [26], PartImageNet [9], PASCAL-Part [5], refCLEF [29], refCOCO [39], refCOCO+ [39], refCOCOg [20], gRefCOCO [15]	150K	1.2M
GLaMM [27]	GranD [27] (Automatically annotated for SA-1B), GranD-f [27] (Based on Flickr-30K [38], RefCOCOg [20], PSG [35])	11M	810M
Read [25]	refCOCO [39], refCOCO+ [39], refCOCOg [20], gRefCOCO [15]	100K	82K
SAM4MLLM [6]	refCOCO [39], refCOCO+ [39], refCOCOg [20], gRefCOCO [15], ADE20K [40], PACO-LVIS [26], PartImageNet [9]	145K	260K
<i>Reinforcement Fine Tuning Datasets</i>			
Seg-Zero [17], Ours (single)	refCOCOg [39]	9k	9k
VisionReasoner [18], Ours (multi)	grefCOCO [15] Lisa++ [36] PACO-LVIS [26]	7k	7k

We first introduce the hallucination evaluation protocol and full judging prompt used for analysing CoT reasoning behaviour (Section C.1). We then present additional results on referring segmentation [39] under the gIoU metric (Section C.2) and object detection results on COCO [14] (Section C.3). Next, we conduct a series of ablation studies examining the effect of group size and its placement within the revision stages of GRPO (Sections C.4 and C.5). In the last, we evaluate our method using the official Qwen2.5VL grounding prompt to demonstrate its prompt-agnostic object grounding performance (Section C.6).

### C.1. Hallucination Experiment

We show the full prompt details for the hallucination checks in Fig. C2. To systematically assess hallucinations in the chain-of-thought (CoT) reasoning, we adopt an “**LLM-as-Judge**” protocol and use GPT-5 to verify the generated reasoning for each sample. The protocol relies solely on the predicted boxes as the visual evidence source and does not use any localization-based metric; GT boxes serve only as semantic disambiguation for the query. Following previous hallucination judgement works [2, 3, 10, 23], the prompt requires GPT-5 to output three binary judgments, each capturing a different failure mode in CoT reasoning:

**1. off topic:** determines whether the reasoning addresses the same concept as the query. This reflects whether the model correctly understands the user intent. For instance, in the first example of Fig. C2, the reasoning discusses a “penguin” while the predicted box contains a bench, leading to an off-topic label.

**2. attribute mismatch:** checks whether the attributes

Table C2. **Comparison on the RefCOCO family** in terms of gIoU. We report performance on RefCOCO (+/g) [39] under the test splits. The checkpoint is identical to that in the main paper. The best results are highlighted in red.

Method	RefCOCO	RefCOCO+	RefCOCOg
	testA	testA	test
Qwen2-VL-7B [31]	68.7	65.7	63.5
Qwen2.5-VL-7B [1]	79.9	76.8	72.8
VisionReasoner-7B [18]	78.9	74.9	71.3
Ours (single)	<b>81.34</b>	<b>78.69</b>	<b>76.23</b>
Ours (multi)	79.99	76.79	74.10

or claims made in the reasoning text are verifiable from the predicted boxes. This metric is crucial for identifying hallucinated properties that are not visually grounded: a model may point to the correct region but still invent colors, shapes, states, or relations that are not supported by the image. In the first example, the claim “aligned with the viewer” doesn’t support any evidence from the predicted bbox, thus producing an attribute-mismatch judgment.

**3. vague claim:** evaluates whether the reasoning is overly generic and lacks a concrete, verifiable anchor tied to the predicted boxes. This distinguishes true evidence-grounded reasoning from generic restatements of the query. If the text does not single out the target using box-level evidence, it is marked as vague.

We apply this prompt to all 995 samples across the validation and test splits of the ReasonSeg [13] dataset, filling {query}, {think}, {pred\_bbox}, and {gt\_bbox} accordingly. The aggregated final hallucination case numbers (for each type) are reported in Table 7 of the main paper.

User Prompt (init. attempt)
<p>Please find “{query}” with a bounding box and points. Compare the difference between objects and find the most closely matched one.</p> <p>Output the thinking process in <code>&lt;think&gt; ... &lt;/think&gt;</code> and the final answer in <code>&lt;answer&gt; ... &lt;/answer&gt;</code> tags.</p> <p>Output one bounding box and two points corresponding to the two largest inscribed circles inside the referenced object in JSON format.</p> <p>i.e., <code>&lt;think&gt; thinking process here &lt;/think&gt;</code>  <code>&lt;answer&gt;{Answer}&lt;/answer&gt;</code></p>
User Prompt (revision group)
<p>You previously predicted bounding box ({box}) and points ({points}) for the question “{query}”, with the reasoning “{think}”.</p> <p>Rethink whether the previous bounding box and points match the target object.</p> <ul style="list-style-type: none"> <li>– If <b>yes</b>: keep the object; tighten the box to its boundaries and adjust the two points inside the object.</li> <li>– If <b>no</b>: discard the predictions and output the correct object with one box and two points (the centers of the two largest inscribed circles).</li> </ul> <p>Respond with exactly:</p> <p><code>&lt;think&gt; thinking process here &lt;/think&gt;</code>  <code>&lt;answer&gt;{example}&lt;/answer&gt;</code></p>

User Prompt (init. attempt)
<p>Please find “{query}” with bounding boxes and points. Compare the difference between object(s) and identify the most closely matched one(s).</p> <p>Output the thinking process in <code>&lt;think&gt; ... &lt;/think&gt;</code> and the final answer in <code>&lt;answer&gt; ... &lt;/answer&gt;</code> tags.</p> <p>Return the bbox(es) and point(s) of the referenced object(s) in JSON format.</p> <p>i.e., <code>&lt;think&gt; thinking process here&lt;/think&gt;</code>  <code>&lt;answer&gt;{example}&lt;/answer&gt;</code></p>
User Prompt (revision group)
<p>You previously predicted bounding box(es) ({box}) and point(s) ({points}) for the question “{query}”, with the reasoning {think}.</p> <p>Rethink whether the previous bbox(es) and point(s) match the target object(s).</p> <ul style="list-style-type: none"> <li>– If <b>yes</b>: keep the same object(s); tighten each bbox to the object boundary and set the point to the object center.</li> <li>– If <b>no</b>: discard previous predictions and output the correct object(s), each with one bbox and one point inside it.</li> </ul> <p>Respond with exactly:</p> <p><code>&lt;think&gt; thinking process here &lt;/think&gt;</code>  <code>&lt;answer&gt;{example}&lt;/answer&gt;</code></p>

Figure A1. **Single- and multi-box setup prompts** used for training and inference. The top and bottom panel show the single and multi settings, respectively. The {query} is replaced with the textual query for each of the sample and {box}, {points}, and {think} are filled with the results parsed from initial attempt.

Table C3. **Object Detection Results** on the COCO dataset [14]. Class confidence is evaluated based on bounding-box size. All results are computed using the official COCO evaluation API, with  $AP_{0.50:0.95}$  as the primary metric. The checkpoint is identical to that in the main paper. Best results are highlighted in red.

Qwen2.5VL-7B	VisionReasoner	Ours (multi)
29.2	37.7	<b>39.2</b>

*Note.* Since the class confidence of Large Vision-Language Models has not yet been unified under a widely agreed protocol, we report these results in the supplementary material instead of the main paper.

Table C4. **Comprehensive COCO Detection Metrics.** All results follow the official COCO evaluation protocol. AP denotes Average Precision and AR denotes Average Recall.

Metric	Small	Medium	Large	All
$AP_{50:95}$	0.236	0.458	0.603	<b>0.392</b>
$AP_{50}$	–	–	–	0.579
$AP_{75}$	–	–	–	0.407
$AR_1$	–	–	–	0.355
$AR_{10}$	–	–	–	0.495
$AR_{100}$	0.279	0.544	0.745	0.501

## C.2. gIoU Results in Referring Segmentation

As shown in Table C2, we report gIoU performance on the RefCOCO (+/g) [39] benchmarks and compare our method with prior work. We note that most existing referring segmentation methods focus primarily on cIoU as the standard metric [13, 17, 25, 37], while the gIoU results in Table C2 are directly borrowed from VisionReasoner [18] for comparability. From the results, our multi-box setting achieves consistently better performance across all three RefCOCO datasets. Specifically, compared with VisionReasoner [18], our method obtains **+1.09%** improvement on RefCOCO (testA), **+1.89%** on RefCOCO+ (testA), and a significant **+2.80%** gain on RefCOCOg (test). Notably, our single-box setting achieves an even larger improvement, surpassing VisionReasoner [18] by **+2.44%** on RefCOCO, **+3.79%** on RefCOCO+, and **+4.93%** on RefCOCOg. These results demonstrate that the proposed group revision leads to more stable grounding and improves the model’s ability to localise the referred object under the stricter gIoU criterion.

## C.3. Object Detection Results

In Table C3, we compare our multi-box setup with VisionReasoner [18] and Qwen2.5VL-7B [1] on the object detection task of the MS COCO [14] dataset. Our method achieves a **+1.5%** improvement under the  $AP_{0.5:0.95}$  metric, indicating that group revision further enhances the model’s ability to localise objects in complex scenes. We addition-

- (1) Examine the IMAGE and REASONING\_TEXT for the given Query.  
 Purpose split:
- GT\_BBOXES: define the TRUE target concept/category for the Query (disambiguation only).
  - PRED\_BBOXES: the ONLY regions used to verify REASONING\_TEXT (hallucination check).
- Do NOT compare predictions to GT (no IoU, no localization grading).
- (2) Inputs:
- Query: user's target concept/goal. (Assume the target is present in the image.)
  - GT\_BBOXES: list of [x1,y1,x2,y2] ints for the TRUE target(s). Used only to clarify what the Query refers to.
  - PRED\_BBOXES: list of [x1,y1,x2,y2] ints (predicted regions). Treat as anonymous crops.
  - REASONING\_TEXT: free text describing the target roughly in the PRED\_BBOXES region(s).
- (3) Global (image-level) judgments over ALL PRED\_BBOXES:
- off\_topic (0|1):  
 Does REASONING\_TEXT address the SAME concept as the Query's target?
    - Use GT\_BBOXES only to understand the intended concept/category.
    - 0 = clearly on-topic wrt the Query's target; 1 = off-topic.
  - attribute\_mismatch (0|1):
    - 1 if the claims are not directly visible/checkable from the PRED\_BBOXES, or conflict with what is seen.
    - 1 if PRED\_BBOXES=[] and the text makes checkable claims (no supporting evidence).
    - 0 if at least one PRED\_BBOX visibly supports the attributes/states stated in REASONING\_TEXT.
    - If evidence across boxes is mixed, use the majority.
  - vague\_claim (0|1):  
 Is REASONING\_TEXT too generic/hedged/unverifiable from PRED\_BBOXES?
    - 1 (vague) if it:
      - Restates the query or makes generic scene statements without isolating a particular instance.
      - Cannot justify which instance is the target among distractors.
    - 0 (not vague) if it includes at least one concrete, verifiable anchor tied to the PRED\_BBOXES that singles out the target.
- (4) Respond EXACTLY with strict JSON on one line:  
 {"off\_topic":0,"attribute\_mismatch":0,"vague\_claim":0,"reason":"<=100 chars, concise, no new claims, no newlines"}

#### Judge Rules:

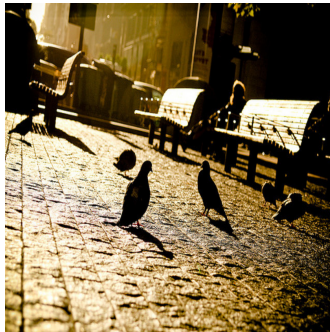
- Judge ONLY against PRED\_BBOXES; never against GT\_BBOXES.
- Do NOT grade localization quality (no IoU, no position accuracy).
- Use only visually checkable evidence from the IMAGE/PRED\_BBOXES; ignore inferred/hidden properties.

#### JSON Rules:

- Keys: off-topic, attribute\_mismatch, vague\_claim, reason (snake\_case).
- Values: integers 0 or 1; reason is ASCII ≤500 chars. No newlines.
- No extra keys, no comments, no trailing commas, no leading/trailing spaces.

#### Example 1:

Query: the bench closest to the person taking the photo.



Think Text: "The penguin appears roughly aligned with the viewer's position."  
 "pred\_bbox": [532, 247, 766, 500], "gt\_bbox": [534, 242, 801, 510],  
 "answer": {"off\_topic":1, "attribute\_mismatch":1, "vague\_claim":1, "reason":"The text asserts the  
 → penguin which is off-topic even the bbox includes a bench. The claims of 'roughly aligned  
 → with the viewer' aren't verifiable and are hedged, so attributes don't match and the  
 → statement is vague."}

Example 2:

Query: determine which object is intended for manual use and commonly features in both the preparation and the consumption of food.  
 ↪ preparation and the consumption of food.



Think Text: To determine which object is intended for manual use and commonly features in both  
 ↪ the preparation and the consumption of food, let's analyze the elements in the  
 ↪ image:\n\n1. **Person**: The person is cooking, but they are not directly interacting with  
 ↪ an object that fits the description.\n2. **Pan**: The pan is being used to cook food, but it  
 ↪ is primarily a cooking implement rather than one that is manually consumed.\n3. **Cutting  
 ↪ Board**: The cutting board is used for chopping food, a preparation action, and does not fit  
 ↪ the criteria of being consumed.\n4. **Spoons**: While spoons are for consumption, the  
 ↪ question specifies "commonly features in both preparation and consumption of  
 ↪ food." \n\nConsidering the above elements:\n- The "spoons" being used for cooking are  
 ↪ tools involved both in food preparation and in eating. \n\n Thus, the answer focuses on  
 ↪ "spoons."

"pred\_bbox": [[574, 482, 707, 588]], "gt\_bbox": [[574, 482, 707, 588], [380, 641, 425, 715]],  
 "answer": {"off\_topic":0, "attribute\_mismatch":0, "vague\_claim":0, "reason":"The spoon described  
 ↪ in box ([574, 482, 707, 588]) fits "manual use" and "preparation + consumption"}

Query: {query}  
 Think Text: {think}  
 "pred\_bbox": {pred\_bbox}, "gt\_bbox": {gt\_bbox}

Figure C2. **LLM-as-Judge Prompt for GPT-5**. We show the full prompt used for hallucination evaluation in CoT reasoning, including the query, image and bounding boxes, together with two illustrative examples. The placeholders {query}, {think}, {pred\_bbox}, and {gt\_bbox} are instantiated using inference outputs and data samples from ReasonSeg [13] across both validation and test splits (995 cases).

Table C5. **Group Number Ablation** in the revision process. We evaluate different sampling sizes in the GRPO group on both segmentation [13, 39] and counting [7] tasks.

Group	RefCOCOg [39]	ReasonSeg [13]	PixMo [7]
n = 4	70.59	63.73	73.92
n = 8	70.67	67.48	75.89
n = 12	70.93	64.02	74.58

ally provide a detailed analysis of small, medium, and large objects for both AP and AR metrics in Table C4. All results are computed using the official COCO evaluation toolkit.

### C.4. Group Number Ablation

Table C5 presents an ablation study on the group size used during the revision stage in GRPO. We evaluate three sampling configurations (4, 8, and 12 group members) across the validation sets of RefCOCOg [39], ReasonSeg [13], and PixMo [7]. Increasing the group size from 4 to 8 consistently improves performance on all three benchmarks, with notable gains on ReasonSeg (from 63.73 to 67.48) and

Table C6. **Ablation on Group Number between Init. Attempt (I) and Group Revision (R)**. The top row uses 1 or 2 initial responses reviewed by 8 or 4 revisions, while the bottom row swaps the roles, with 8 or 4 initial responses reviewed by 1 or 2 revisions. The results are based on both segmentation [13, 39] and counting [7] tasks and the best results are highlighted in red.

Revise Style	I	R	Refcocog	ReasonSeg	Pixmo
	1	8	70.67	67.48	75.89
	2	4	70.96	66.64	74.58
	8	1	64.99	60.36	71.71
	4	2	68.43	66.92	73.57

PixMo (from 73.92 to 75.89). However, further increasing the group size to 12 produces mixed outcomes: RefCOCOg continues to improve (70.93), while ReasonSeg and PixMo show slight drops compared to the 8 member setting.

### C.5. Group Number on Different Stages

Table C6 presents an ablation study on the number of group members involved in the Initial Attempt (I) and the Group

Table C7. **Comparison on RefCOCO (+/g)** [39] in cIoU, evaluated using Qwen2.5VL [1] with its default grounding prompt: "Locate {Question} in this image and output the bbox coordinates in JSON format." The best results are highlighted in red.

Method	RefCOCO		RefCOCO+		RefCOCOg
	testA	testB	testA	testB	test
Qwen2-VL-7B	58.9	48.0	52.3	41.9	52.1
Qwen2.5-VL-7B*#†	77.9	66.5	74.0	55.6	70.9
Ours (single)*	<b>79.45</b>	<b>69.52</b>	<b>74.86</b>	59.98	<b>72.30</b>
Ours (multi)†	79.32	69.08	74.36	<b>60.09</b>	72.08

Revision (R) of our multi setup. The top half of the table evaluates settings where 1 or 2 initial responses are reviewed by groups of 8 or 4 members, while the bottom half examines the inverse configuration, where 8 or 4 initial responses are reviewed by much smaller groups of only 1 or 2 members. From the results, we observe that allocating a larger number of revisers (R) to a small number of initial attempts (I) consistently yields the best performance across all three benchmarks. For example, the 1 to 8 and 2 to 4 settings achieve the strongest scores on ReasonSeg [13] and Pixmo [7], and also attain the highest score on RefCOCOg (70.96). Conversely, when many initial attempts are reviewed by only 1–2 revisers (8 to 1 and 4 to 2), performance drops noticeably, particularly on RefCOCOg [39] and ReasonSeg [13]. These findings suggest that the group revision benefits more from having a diverse and larger group in GRPO, which helps reasoning and correct the initial reasoning more effectively than increasing the number of initial attempts alone.

### C.6. Results with Qwen2.5VL Official Prompt

Table C7 presents the cIoU results on the RefCOCO (+/g) benchmarks [39], evaluated using Qwen2.5VL [1]’s default grounding prompt: "Locate {Question} in this image and output the bbox coordinates in JSON format.". Despite using a different prompt for object-level grounding during RFT (shown in Fig. A.3), our trained LVLMs still enhance the model’s general object-level grounding capability when compared with the original base LVLm, Qwen2.5VL-7B [1]. Our multi-box setup achieves solid performance on all RefCOCO (+/g) evaluation splits and surpasses Qwen2.5VL-7B [1], especially on the more challenging testB splits where referring ambiguity is higher. For instance, the multi setting yielding improvements of **+1.42%** on RefCOCO testA, **+4.49%** on RefCOCO+ testB, and **+1.18%** on RefCOCOg. Our single-box setup further strengthens this ability and yields the best results overall. It achieves the best results, including **79.45%** and **69.52%** on RefCOCO testA and testB, and **72.30%** on RefCOCOg. These results show that the group revision optimisation paradigm improves object-level grounding in LVLMs in a prompt-agnostic and

robust manner.

## D. Mathematical Analysis of Policy Invariance in our Reward Shaping

In this section, we provide a minimal proof sketch showing that our consolidation signal follows the **policy-invariance principle of reward shaping** [8, 21, 33], which guarantees that the shaping may modify the magnitude or smoothness of the reward used during optimisation, but not the preference ordering over candidate actions. This ensures that the learning signal is refined without altering the optimal policy induced by the original reward function.

### Lemma 1 (Preference preservation relative to GRPO).

*Statement.* We begin by considering the standard GRPO, which operates on a group of responses  $\{o_i^{(\text{GRPO})}\}_{i=1}^G$  with reward:

$$r_i^{(\text{GRPO})} = \mathcal{R}_{\text{format}}(o_i^{(\text{GRPO})}) + \mathcal{R}_{\text{acc}}(o_i^{(\text{GRPO})}, y). \quad (14)$$

Our method introduces a group revision stage, where each revision candidate  $o_i^{(2)}$  is associated with a shaping potential  $\Phi(s_{\text{shape},i}^{(2)})$  as defined in Eq. (7). For any pair of responses  $a, b$  within the same sample, if  $a$  achieves better grounding quality than  $b$ , then the following holds:

$$\Phi(s_{\text{shape},a}^{(2)}) < \Phi(s_{\text{shape},b}^{(2)}) \Rightarrow r_a > r_b \quad \text{and} \quad r_a^{(\text{GRPO})} > r_b^{(\text{GRPO})}. \quad (15)$$

Here,  $r_a$  and  $r_b$  denote the rewards under our optimisation. Thus, our shaped reward induces the **same** preference ordering as the standard GRPO reward, ensuring that no contradictory supervision is introduced.

*Proof sketch.* As shown in Eq. (5), a smaller alignment cost  $\Phi(s_{\text{shape},a}^{(2)})$  indicates better IoU and lower L1 errors with respect to the ground-truth result. Then we have:

$$\Phi(s_{\text{shape},a}^{(2)}) < \Phi(s_{\text{shape},b}^{(2)}) \Rightarrow \mathcal{R}_{\text{acc}}(o_a^{(2)}, y) > \mathcal{R}_{\text{acc}}(o_b^{(2)}, y). \quad (16)$$

From Eq. (6),  $\Delta\phi_i$  is a non-negative, monotonically decreasing function for fixed  $\Phi(s_{\text{shape}}^{(1)})$ , so we can derive:

$$\Phi(s_{\text{shape},a}^{(2)}) < \Phi(s_{\text{shape},b}^{(2)}) \Rightarrow \Delta\phi_a \geq \Delta\phi_b \geq 0. \quad (17)$$

If we ignore the format term  $\mathcal{R}_{\text{format}}$ , which is shared across all valid responses, Eq. (7) gives:

$$r_a - r_b = (\mathcal{R}_{\text{acc}}(o_a^{(2)}, y) - \mathcal{R}_{\text{acc}}(o_b^{(2)}, y)) + \omega(\Delta\phi_a - \Delta\phi_b), \quad (18)$$

where the first term is strictly positive, and the second one is non-negative. Hence, the shaped reward is a monotonic refinement of the same ground-truth objective used in standard GRPO, and in particular, induces the same preference ordering  $r_a^{(\text{GRPO})} > r_b^{(\text{GRPO})}$ .  $\square$

Table E8. **InternVL3.5-8B Results** on Segmentation and Counting. Evaluation metrics follow Tab. 1 (main paper).

Method	ReasonSeg		RefCOCO		RefCOCO+		RefCOCOg	Average	Pixmo		Count	Average
	val	test	testA	testB	testA	testB	test		val	test	test	
InternVL3.5-8B [Arxiv 2025]	54.07	50.94	77.40	69.28	74.39	61.62	70.61	65.47	63.80	64.52	79.56	69.29
Ours (single-object)	63.29	59.71	78.35	71.57	75.86	62.54	72.09	69.06	–	–	–	–
Ours (multi-object)	65.62	63.57	77.36	70.33	75.27	61.94	71.52	69.37	70.66	69.83	88.30	76.26

Table F9. **Inference Strategy Ablation** with direct single-pass and **test-time revision** in ReasonSeg and RefCOCO datasets.

Method	ReasonSeg				RefCOCO			
	Valid		Test		TestA		TestB	
Qwen2.5VL-7B	56.9	60.9	52.1	57.3	77.9	78.8	66.5	68.0
Ours (single-object)	67.0	65.4	61.1	61.8	80.8	80.7	73.2	74.2
Ours (multi-object)	67.5	66.9	66.7	66.1	78.0	78.9	69.5	70.6

## E. Generalisation to Other LVLMs

Tab. E8 reports results with a different backbone (InternVL3.5-8B) on both segmentation and counting tasks. We keep the training framework unchanged and replace the Qwen backbone with InternVL3.5-8B. Consistent improvements in both single- and multi-object settings suggest that our method generalises across backbones.

## F. Inference Ablations

Tab. F9 reports test-time ablations comparing direct single-pass inference and revision-based prompting. While the base Qwen model benefits substantially from test-time revision, our trained models show only marginal gains on RefCOCO [14] and slight drops on ReasonSeg [13]. This suggests that revision behaviour is largely internalised: test-time revision can improve localisation in simpler cases but may disrupt already correct predictions in more complex queries.

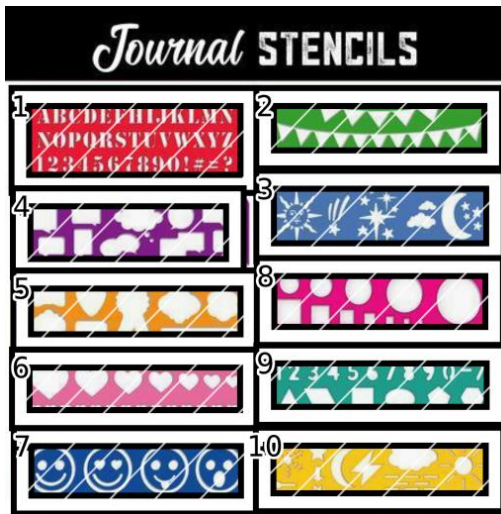
## G. Additional Qualitative Results

In this section, we present qualitative results from our method on counting [22], as well as multi-box and single-box reasoning segmentation [13]. Specifically, Fig. G3 shows visualisations for the counting task, while Fig. G4 and Fig. G5 illustrate results from the single-box and multi-box setups, respectively. Across these examples, the models trained with our group revision strategy exhibit clearer grounding and more reliable object-level localisation in complex scenes.

## References

- [1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 1, 3, 4, 7
- [2] Zechen Bai, Pichao Wang, Tianjun Xiao, Tong He, Zongbo Han, Zheng Zhang, and Mike Zheng Shou. Hallucination of multimodal large language models: A survey. *arXiv preprint arXiv:2404.18930*, 2024. 3
- [3] Catarina G Belém, Pouya Pezeshkpour, Hayate Iso, Seiji Maekawa, Nikita Bhutani, and Estevam Hruschka. From single to multi: How llms hallucinate in multi-document summarization. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 5276–5309, 2025. 3
- [4] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1209–1218, 2018. 3
- [5] Xianjie Chen, Roozbeh Mottaghi, Xiaobai Liu, Sanja Fidler, Raquel Urtasun, and Alan Yuille. Detect what you can: Detecting and representing objects using holistic models and body parts. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1971–1978, 2014. 3
- [6] Yi-Chia Chen, Wei-Hua Li, Cheng Sun, Yu-Chiang Frank Wang, and Chu-Song Chen. Sam4mllm: Enhance multimodal large language model for referring expression segmentation. In *European Conference on Computer Vision*, pages 323–340. Springer, 2024. 3
- [7] Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 91–104, 2025. 6, 7
- [8] Sam Devlin, Logan Yliniemi, Daniel Kudenko, and Kagan Tumer. Potential-based difference rewards for multiagent reinforcement learning. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pages 165–172, 2014. 1, 7
- [9] Ju He, Shuo Yang, Shaokang Yang, Adam Kortylewski, Xiaoding Yuan, Jie-Neng Chen, Shuai Liu, Cheng Yang, Qihang Yu, and Alan Yuille. Partimagenet: A large, high-quality dataset of parts. In *European Conference on Computer Vision*, pages 128–145. Springer, 2022. 3
- [10] Chaoya Jiang, Hongrui Jia, Mengfan Dong, Wei Ye, Haiyang Xu, Ming Yan, Ji Zhang, and Shikun Zhang. Hal-eval: A universal and fine-grained hallucination evaluation framework for large vision language models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 525–534, 2024. 3
- [11] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017. 3
- [12] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao

How many stencils are there in the image?



... Each stencil appears to be a separate item, and there are multiple stencils visible in the image. I will count each distinct stencil present in the image. Answer: **10**

How many ice creams are there in the image?



The image shows a grid of ice cream cones, each with a different flavor and color. Observing the image, there are 8 ice cream cones, each occupying a separate cell in the grid. Answer: **8**

How many birdhouse patterns are there in the image?



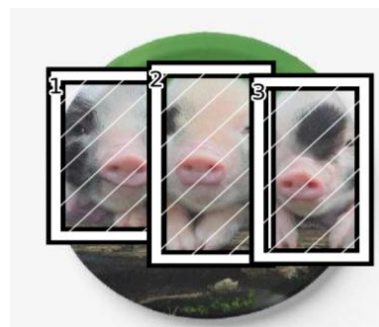
... Observing the image, each square contains a different birdhouse design, and there are no repeated patterns. Therefore, the count should be the total number of squares, which is 9. Answer: **9**

How many stars are there in the image?



The task is to count the number of stars in the image. Observing the image, there are five stars hanging on a string. Each star is distinct and separate from the others, and they are evenly spaced along the string. Answer: **5**

How many pigs are there in the image?



The image shows three piglets with their heads and upper bodies visible. The question asks for the number of pigs in the image, which can be directly counted from the image. Answer: **3**

How many posters are there in the image?



The image shows a grid of movie posters, and each poster appears to be distinct and separate from the others. By visually counting the number of individual posters, we can determine the total count. Answer: **10**

Figure G3. Qualitative results for the counting task on CountBench [22]. The final answer is obtained by counting the number of detected bounding boxes.

Something indicating the identity of the car



... The license plate is a key identifier for the car, providing information about the country and possibly the region where the car is registered. The license plate in the image is "DE-84-73," which suggests that the car is registered in Germany. The car's make and model can also be inferred from the design and features, but the license plate is the most direct way to identify the car's registration and possibly its origin.

The object used for stirring milk or coffee



The object used for stirring milk or coffee in the image is the straw in the glass. The straw is inserted into the drink and is likely used to stir the coffee and milk mixture. The straw is the most closely matched object for stirring milk or coffee in this context.

... What in the picture can be worn on the head to enhance the appearance?



The image shows a person with green face paint and an afro hairstyle, wearing a floral crown. The crown is made of leaves, flowers, and other natural elements, which can be considered a decorative accessory often worn during festive occasions or wedding ceremonies. The most closely matched object in the picture that can be worn on the head to enhance the appearance is the floral crown.

When preparing for a festive event like Halloween, people often use certain objects to decorate their homes. What object in the picture would be suitable for this purpose?



The image shows an old-fashioned car decorated with pumpkins and sunflowers, which are commonly used for Halloween decorations. The pumpkins are placed in the back of the car, and the sunflowers are attached to the side, adding a festive touch. ... The most closely matched object for Halloween decoration in the picture is the pumpkins in the back of the car. They are large and orange, which are typical colors and shapes associated with Halloween.

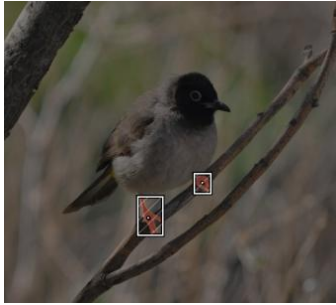
In a chicken salad, which ingredients in the picture provide a fresh and crunchy texture?



The fresh and crunchy texture in a chicken salad would typically come from items like lettuce, croutons, and possibly some vegetables. In the picture, the lettuce and croutons are the most likely candidates for providing a fresh and crunchy texture. The lettuce is fresh and crisp, and the croutons are likely to be crunchy. The tomato slice is fresh but not crunchy, and the tuna is not fresh or crunchy.

Figure G4. **Qualitative results for reasoning segmentation** with the single-box setup. The generated masks are produced by SAM2 [28], guided by a single bounding box and two points inside the objects of interest.

The body parts of the bird that grip onto tree branches



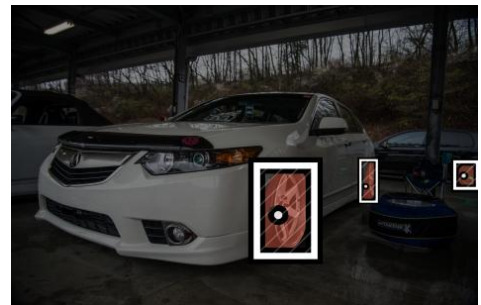
... Observing the image, the bird is perched on a branch, and the visible parts that appear to be gripping the branch are the feet. These are typically located at the bottom of the bird's body and are used for perching.

... serve grilled corn on the cob. What item in the picture would likely be used to drain excess water from the corn before serving?



The question asks for an item used to drain excess water from grilled corn. Observing the image, there is a basket-like structure that appears to be used for draining. This structure is visible in the background and seems to be the most likely candidate for the item mentioned in the question.

... which part of the car in the picture will be in direct contact with the ground?



The question asks for the part of the car that will be in direct contact with the ground when the car is parked. In the image, the car is already parked, and the tires are visible. The tires are the part of the car that will be in direct contact with the ground.

The cars that are not taxi



Observing the image, there are multiple yellow taxis in the foreground, and a black car is visible on the right side of the image. This black car is not a taxi, as it does not have the distinctive yellow color and taxi markings.

... it is common to see symbolic representations or idols of deities being worshipped. What objects in the picture could be used as idols or representations of deities?



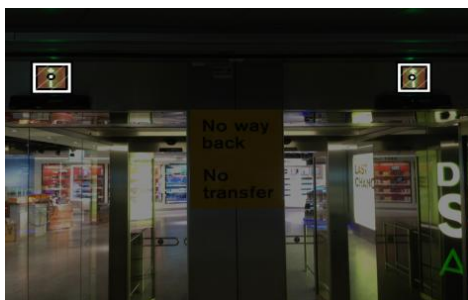
... The objects that could be used as idols or representations of deities include the figures themselves, as they are central to the scene and are depicted in a manner that suggests they are being revered. The other objects in the image, such as the fire, the pot, and the offerings, are not typically considered as idols or representations of deities in the same way.

The frictional part used for igniting



... The question asks for the frictional part used for igniting, which is typically the black tip at the end of the matchstick. Observing the image, the black tips at the end of the matchsticks are the frictional parts used for igniting.

... What in the picture has neon lights that can help guide people to a specific location?



In the image, there are two green neon signs on the top left and right sides of the image, which are likely used to guide people. These signs are typically used in commercial areas to attract attention and create a vibrant atmosphere.



The motorcycles tied together

The question asks for the motorcycles that are tied together. Observing the image, there are two blue motorcycles parked side by side, and they appear to be tied together with a chain. The red motorcycle is parked separately and not tied to the blue ones.

Figure G5. **Qualitative results for reasoning segmentation** with the multi-box setup. The generated masks are produced by SAM2 [28], guided by multiple bounding boxes, each with one point inside the objects of interest.

- Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023. 1
- [13] Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9579–9589, 2024. 2, 3, 4, 6, 7, 8
- [14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 3, 4, 8
- [15] Chang Liu, Henghui Ding, and Xudong Jiang. Gres: Generalized referring expression segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 23592–23601, 2023. 3
- [16] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 1
- [17] Yuqi Liu, Bohao Peng, Zhisheng Zhong, Zihao Yue, Fanbin Lu, Bei Yu, and Jiaya Jia. Seg-zero: Reasoning-chain guided segmentation via cognitive reinforcement. *arXiv preprint arXiv:2503.06520*, 2025. 2, 3, 4
- [18] Yuqi Liu, Tianyuan Qu, Zhisheng Zhong, Bohao Peng, Shu Liu, Bei Yu, and Jiaya Jia. Visionreasoner: Unified visual perception and reasoning via reinforcement learning. *arXiv preprint arXiv:2505.12081*, 2025. 1, 2, 3, 4
- [19] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 1
- [20] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11–20, 2016. 3
- [21] Andrew Y Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *Icml*, pages 278–287. Citeseer, 1999. 1, 7
- [22] Roni Paiss, Ariel Ephrat, Omer Tov, Shiran Zada, Inbar Mosseri, Michal Irani, and Tali Dekel. Teaching clip to count to ten. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3170–3180, 2023. 8, 9
- [23] Shrey Pandit, Jiawei Xu, Junyuan Hong, Zhangyang Wang, Tianlong Chen, Kaidi Xu, and Ying Ding. Medhallu: A comprehensive benchmark for detecting medical hallucinations in large language models. *arXiv preprint arXiv:2502.14302*, 2025. 3
- [24] Renjie Pi, Lewei Yao, Jiahui Gao, Jipeng Zhang, and Tong Zhang. Perceptiongpt: Effectively fusing visual perception into llm. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 27124–27133, 2024. 2, 3
- [25] Rui Qian, Xin Yin, and Dejing Dou. Reasoning to attend: Try to understand how< seg> token works. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. 3, 4
- [26] Vignesh Ramanathan, Anmol Kalia, Vladan Petrovic, Yi Wen, Baixue Zheng, Baishan Guo, Rui Wang, Aaron Marquez, Rama Kovvuri, Abhishek Kadian, et al. Paco: Parts and attributes of common objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7141–7151, 2023. 3
- [27] Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M Anwer, Eric Xing, Ming-Hsuan Yang, and Fahad S Khan. Glamm: Pixel grounding large multimodal model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13009–13018, 2024. 2, 3
- [28] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 10, 11
- [29] Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. Grounding of textual phrases in images by reconstruction. In *European Conference on Computer Vision*, pages 817–834. Springer, 2016. 3
- [30] Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*, 2024. 1
- [31] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 3
- [32] XuDong Wang, Shaolun Zhang, Shufan Li, Konstantinos Kallidromitis, Kehan Li, Yusuke Kato, Kazuki Kozuka, and Trevor Darrell. Segllm: Multi-round reasoning segmentation. *arXiv preprint arXiv:2410.18923*, 2024. 3
- [33] Eric Wiewiora. Potential-based shaping and q-value initialization are equivalent. *Journal of Artificial Intelligence Research*, 19:205–208, 2003. 7
- [34] Zhuofan Xia, Dongchen Han, Yizeng Han, Xuran Pan, Shiji Song, and Gao Huang. Gsva: Generalized segmentation via multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3858–3869, 2024. 2, 3
- [35] Jingkang Yang, Yi Zhe Ang, Zujin Guo, Kaiyang Zhou, Wayne Zhang, and Ziwei Liu. Panoptic scene graph generation. In *European conference on computer vision*, pages 178–196. Springer, 2022. 3
- [36] Senqiao Yang, Tianyuan Qu, Xin Lai, Zhuotao Tian, Bohao Peng, Shu Liu, and Jiaya Jia. Lisa++: An improved baseline for reasoning segmentation with large language model. *arXiv preprint arXiv:2312.17240*, 2023. 3
- [37] Zuyao You and Zuxuan Wu. Seg-r1: Segmentation can be surprisingly simple with reinforcement learning. *arXiv preprint arXiv:2506.22624*, 2025. 4
- [38] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New

similarity metrics for semantic inference over event descriptions. *Transactions of the association for computational linguistics*, 2:67–78, 2014. [3](#)

- [39] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *European conference on computer vision*, pages 69–85. Springer, 2016. [3](#), [4](#), [6](#), [7](#)
- [40] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127(3):302–321, 2019. [3](#)