

# 000 001 002 003 004 005 REX-THINKER: GROUNDED OBJECT REFERRING VIA 006 CHAIN-OF-THOUGHT REASONING 007 008 009

010 **Anonymous authors**  
011 Paper under double-blind review  
012  
013  
014  
015  
016  
017  
018  
019  
020  
021  
022  
023  
024  
025  
026  
027  
028  
029  
030  
031  
032  
033  
034  
035  
036  
037  
038  
039  
040  
041  
042  
043  
044  
045  
046  
047  
048  
049  
050  
051  
052  
053  
054  
055  
056  
057  
058  
059  
060  
061  
062  
063  
064  
065  
066  
067  
068  
069  
070  
071  
072  
073  
074  
075  
076  
077  
078  
079  
080  
081  
082  
083  
084  
085  
086  
087  
088  
089  
090  
091  
092  
093  
094  
095  
096  
097  
098  
099  
0100

## ABSTRACT

Object referring aims to detect all objects in an image that match a given language description. A robust model for this task must be grounded, providing explainable and visually faithful predictions. This demands two key properties: **1) Verifiability**, through interpretable reasoning linked to visual evidence; and **2) Trustworthiness**, by rejecting expressions without matching objects, thereby mitigating hallucination. Current methods, often treating referring as direct bounding box prediction, largely fail to provide such interpretability or effectively abstain. In this work, we propose Rex-Thinker, an MLLM framework that reformulates object referring as an explicit Chain-of-Thought (CoT) reasoning task. Recognizing that standalone detectors lack sufficient language comprehension and MLLMs struggle with precise localization, Rex-Thinker adopts a two stage approach. It first leverages an open-vocabulary detector for candidate object proposals, then employs an MLLM to perform step-by-step reasoning over these candidates to assess their match with the expression. To enable this, we construct HumanRef-CoT, a large-scale (90,824 samples) CoT-style referring dataset. Each reasoning trace follows a structured planning, action, and summarization format, facilitating the learning of decomposed, grounded reasoning. Rex-Thinker is trained in two stages: supervised fine-tuning for structured reasoning, followed by GRPO-based reinforcement learning for enhanced accuracy and generalization. Extensive experiments demonstrate that Rex-Thinker not only achieves state-of-the-art performance in precision and interpretability but also drastically improves the ability to reject hallucinated outputs on in-domain evaluation, while showing strong generalization in out-of-domain settings.

## 1 INTRODUCTION

Object Referring, also known as Referring Expression Comprehension (REC) (Qiao et al. (2020); Yu et al. (2016); Mao et al. (2016); Kazemzadeh et al. (2014); Zhang et al. (2019); Yu et al. (2018); Yang et al. (2019); Liao et al. (2020)), aims to predict the bounding boxes for objects in an image that match a given natural language description, which may refer to visual attributes, spatial relations, or interactions. This task has broad applications, however, compared to standard open-vocabulary object detection (Jiang et al. (2025a); Ren et al. (2024a;b); Liu et al. (2024); Yao et al. (2022a); Jiang et al. (2024a; 2023); Li et al. (2024a); Ren et al. (2024a); Li et al. (2022); Cheng et al. (2024); Minderer et al. (2022); Zareian et al. (2021); Wu et al. (2023)), REC is significantly more challenging, as it requires both fine-grained visual grounding and more complicated language understanding.

Benefiting from the strong language comprehension capabilities of large language models (LLMs), multimodal large language models (MLLMs) have demonstrated impressive performance on this task. There are mainly two paradigms: one treats bounding box coordinates as text tokens and predicts them directly (Chen et al. (2023); You et al. (2023); Zhang et al. (2024); Wang et al. (2023); Zhan et al. (2025; 2024); Bai et al. (2025); Wu et al. (2024); Chen et al. (2024); McKinzie et al. (2024)), while the other adopts a retrieval-based strategy (Ma et al. (2024); Jiang et al. (2024b; 2025b)), where the model is given a set of candidate boxes and predicts the box indices that match the expression. Although both approaches have shown promising results, they remain fundamentally implicit, lacking interpretable reasoning steps that reveal how the model arrives at its final prediction. Furthermore, these models are prone to hallucination (Jiang et al. (2025b)), often producing outputs for objects that do not exist in the image, thereby limiting their reliability in real-world applications.



Figure 1: An example of Rex-Thinker for object referring with CoT reasoning of planning (task decomposition), action (evaluating each candidate), and summarization (final decision). Each step is grounded in a specific hint box (as denoted in the left image), enabling interpretable predictions.

We argue that a robust referring system should be *grounded*, i.e., its predictions must be both explainable and tightly linked to visual evidence. This requires two essential properties: **1) Verifiable**, by providing an explicit reasoning process that allows its decisions to be examined and traced to specific image regions; and **2) Trustworthy**, by minimizing hallucinated outputs and learning to reject when no object in the image satisfies the given description. To meet these criteria, we draw inspiration from how humans naturally approach referring expressions. For example, when asked to locate “the person wearing a blue shirt”, humans would typically first identify all people in the image, then examine each one to determine whether it matches the described attribute. This step-by-step approach reflects a grounded reasoning process, i.e., first localizing relevant object candidates, and then carefully verifying each one against the expression.

Motivated by this observation, we propose Rex-Thinker, an MLLM framework that performs object referring through explicit Chain-of-Thought (CoT) reasoning. Specifically, given an image and a referring expression, we first use an open-vocabulary object detector (Liu et al. (2024)) to extract all candidate object boxes corresponding to the referred category. These candidate boxes, along with the image and the expression, are then passed into the MLLM to form a two stage system. Rex-Thinker follows a structured CoT framework consisting of three key stages as shown in Figure 1: **1) Planning**, where the model decomposes the referring expression into subgoals; **2) Action**, where the model examines each candidate box to determine whether it satisfies its current subgoal; **3) Summarization**, where it aggregates the intermediate decisions to produce the final prediction. Following DeepSeek-R1 (Guo et al. (2025a)), we instruct the model to place its reasoning steps within a `<think>...</think>` block and to output the final prediction inside a `<answer>...</answer>` block. This structured reasoning process not only improves interpretability, but also enables transparent and verifiable predictions, as each reasoning step is grounded in a specific candidate region.

To support this CoT framework, we construct a CoT-style referring dataset named HumanRef-CoT, containing 90,824 samples generated by prompting GPT-4o (Hurst et al. (2024)) on the HumanRef (Jiang et al. (2025b)) dataset. Each example is annotated with a structured reasoning trace following the planning, action, and summarization paradigm, enabling explicit supervision for step-by-step reasoning. We train our model in two stages: a cold-start supervised fine-tuning phase to teach the model how to perform structured reasoning, followed by reinforcement learning (RL) based on Group Relative Policy Optimization (GRPO) (Shao et al. (2024)) to further improve accuracy and generalization. Experiments demonstrate that our CoT-based approach consistently outperforms direct coordinate prediction baselines. On the in-domain HumanRef benchmark, our model achieves state-of-the-art results with higher detection accuracy and significantly fewer hallucinated outputs, especially on rejection cases. In out-of-domain evaluations on RefCOCOg (Mao et al. (2016)), the model trained only on HumanRef-CoT shows strong zero-shot generalization. Further fine-tuning with GRPO on RefCOCOg yields additional performance gains while preserving the model’s ability to perform grounded CoT reasoning across arbitrary object categories. Our contributions lie in:

- We formulate the grounded object referring task as a **planning–action–summarization** problem, leveraging Chain-of-Thought reasoning to build a verifiable and trustworthy system.

- 108 • We introduce HumanRef-CoT, the first large-scale dataset for grounded object referring with  
109 step-by-step reasoning annotations, enabling the supervised training of model interpretability.  
110
- 111 • We propose Rex-Thinker, a grounded object referring model trained via cold-start SFT and  
112 GRPO-based reinforcement learning. Rex-Thinker achieves SOTA performance on the Human-  
113 Ref benchmark and demonstrates strong generalization on out-of-domain scenes and objects.  
114

## 115 2 RELATED WORK

116  
117 **MLLM-based Object Referring Methods.** Recent progress in multimodal large language models  
118 (MLLMs) OpenAI (2023); Bai et al. (2025); Wu et al. (2024); Chen et al. (2024); Deitke et al.  
119 (2024); Agrawal et al. (2024); Wang et al. (2024); Li et al. (2024b; 2025); Liu et al. (2023); Chen  
120 et al. (2025a); Guo et al. (2025b); Yao et al. (2022b;b); Yang et al. (2023b) has led to strong perfor-  
121 mance in referring expression comprehension. Existing approaches typically follow two paradigms.  
122 One line of work treats bounding box coordinates as textual tokens Chen et al. (2021) and directly  
123 generates them during decoding Chen et al. (2023); You et al. (2023); Wang et al. (2023); Zhan  
124 et al. (2025); Zhang et al. (2024). The other line formulates the task as retrieval Jiang et al. (2024b);  
125 Ma et al. (2024); Jiang et al. (2025b), where a detector proposes candidate regions and the model  
126 selects the best-matching box indices based on the input expression. This decouples localization  
127 from semantic understanding and simplifies learning. While both paradigms achieve strong results  
128 on standard benchmarks such as RefCOCO/+g Mao et al. (2016); Yu et al. (2016), they face key  
129 limitations: a lack of interpretability and an inability to abstain when no object in the image matches  
130 the expression Jiang et al. (2025b). To address this, we introduce a Chain-of-Thought reasoning  
131 framework that enables step-by-step evaluation over candidate boxes. This improves interpretabil-  
132 ity, reduces hallucinations, and grounds the model’s predictions in the input image.  
133

134 **Reasoning-based LLMs and MLLMs.** Recent work in large language models (Jaech et al.  
135 (2024); Guo et al. (2025a); Team et al. (2025); Muennighoff et al. (2025); Xiang et al. (2025);  
136 Xiong et al. (2025); Chu et al. (2025); OpenAI et al. (2025)) has demonstrated that reasoning ability  
137 can be significantly enhanced through Chain-of-Thought (CoT) training or reinforcement learning-  
138 based post-training. OpenAI o1 (Jaech et al. (2024)) model demonstrates that inference-time scaling  
139 can greatly enhance performance on complex tasks like math and coding. DeepSeek-R1 (Guo et al.  
140 (2025a)) introduces GRPO (Shao et al. (2024)) as a post-training method to improve reasoning with-  
141 out requiring costly critic models. In the multimodal domain, efforts such as LLaVA-CoT (Xu et al.  
142 (2024)) and LlamaV-o1 (Thawakar et al. (2025)) aim to enhance reasoning by constructing CoT-  
143 style data or employing multi-step curriculum learning, without relying on reinforcement learning.  
144 For referring expression comprehension task, reinforcement learning has been used in early work  
145 to model the grounding process as a sequential reasoning problem, such as through iterative shrink-  
146 ing Sun et al. (2021) or dynamic reasoning steps Zhang et al. (2023). More recently, inspired by  
147 DeepSeek-R1 (Guo et al. (2025a)), a growing number of works adopt GRPO-based post-training to  
148 endow MLLMs with reasoning capabilities. GRPO has been successfully applied to enhance multi-  
149 modal reasoning across a wide range of domains, including mathematical problem solving (Yang  
150 et al. (2025); Peng et al. (2025); Zhang et al. (2025); Deng et al. (2025); Wei et al. (2025)), video  
151 understanding (Feng et al. (2025); Liao et al. (2025)), and perception tasks (Liu et al. (2025a;b);  
152 Ma et al. (2025); Shen et al. (2025); Yu et al. (2025)) such as object detection, segmentation, and  
153 referring expression comprehension. Following the DeepSeek-R1 paradigm, we first fine-tune Rex-  
154 Thinker on structured CoT data to teach the model how to perform grounded object reasoning.  
155 GRPO is then applied in a second stage to further improve accuracy and generalization.  
156

## 157 3 CHAIN-OF-THOUGHT REASONING REFERRING DATA

158 High-quality supervision is critical for teaching the model to reason explicitly. To this end, we  
159 develop a data engine that generates structured referring annotations aligned with our Chain-of-  
160 Thought formulation. In this section, we introduce the design principles of our CoT reasoning  
161 structure and present the data construction pipeline that transforms existing REC annotations into  
step-by-step reasoning traces suitable for supervised training.

162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174

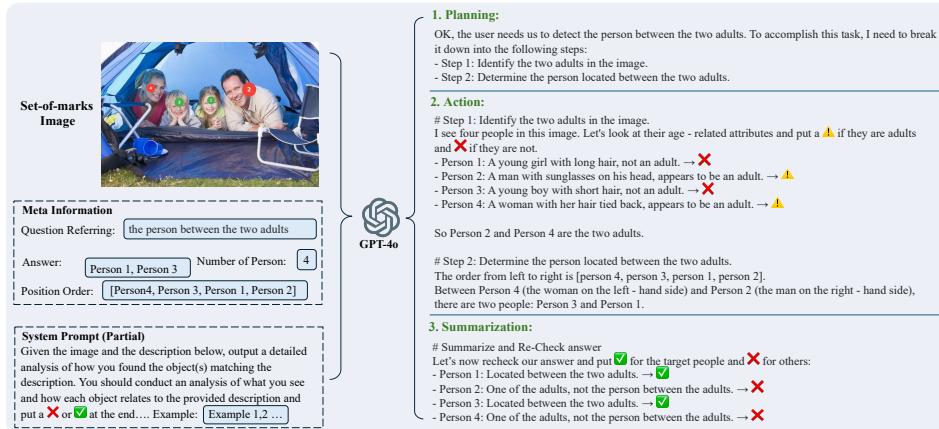


Figure 2: Overview of the proposed CoT reasoning referring data engine. We prompt GPT-4o to generate a three-step CoT reasoning process, including planning, action, and summarization.

### 3.1 COT FORMULATION

The core idea behind our CoT formulation for REC is to transform the task into a structured, grounded reasoning process over a set of candidate objects. Rather than directly predicting the referred object, the model evaluates each candidate in sequence, guided by input box hints that localize specific regions in the image. We decompose this CoT process into three key stages:

- **Planning:** The model analyzes the complexity of the referring expression and determines how many reasoning steps are needed. For simple expressions, it may plan a single step to directly match an attribute such as color or size. For more complex expressions, the model generates a multi-step plan, where each step focuses on resolving a specific sub-aspect.
- **Action:** Based on the reasoning plan, the model checks whether each candidate region, grounded via its input box hint, satisfies the current subgoal. This makes the reasoning clear and directly tied to specific regions in the image.
- **Summarization:** Finally, the model reviews the reasoning results across all steps and determines which objects best match the overall expression and outputs the final prediction.

This structured CoT process improves both interpretability and verifiability. Each candidate is evaluated corresponding to the input box hints, allowing every reasoning step to be explicitly grounded to a specific region of the image. This makes the model’s decisions transparent and easy to trace. Additionally, breaking complex expressions into sub-tasks enables step-by-step reasoning, which enhances accuracy and reflects how humans typically process such tasks.

### 3.2 DATA ENGINE PIPELINE

Building on the structured CoT formulation, we develop a data engine that leverages GPT-4o (Hurst et al. (2024)) to generate high-quality CoT annotations tailored to the referring task.

**Data Acquisition** We construct our CoT dataset based on HumanRef (Jiang et al. (2025b)), a recently proposed dataset specifically designed for REC in human-centric scenarios. Unlike prior REC datasets such as RefCOCO/+g (Mao et al. (2016); Yu et al. (2016)), HumanRef emphasizes multi-instance referring expressions, where a single expression may refer to multiple target persons. It also categorizes expressions into six distinct subsets: attribute, position, interaction, reasoning, celebrity recognition, and rejection. Since the HumanRef dataset provides all person boxes in an image, it can be directly used in our CoT annotation pipeline.

**GPT-4o Annotation** To generate high-quality CoT annotations, we employ in-context prompting with GPT-4o (Hurst et al. (2024)) as shown in Figure 2. Given an image and the bounding boxes of all persons within it, we apply the Set-of-Mark (Yang et al. (2023a)) strategy: each individual is labeled with an indexed visual marker, where ground-truth targets are marked in green and others in red. This design grounds the answer and guides GPT-4o to reason along the correct path. The prompt includes three key components: 1) meta-information such as the referring question, the number of people, their left-to-right spatial order, and the correct answer; 2) a system prompt specifying the desired planning–action–summarization structure; and 3) several in-context examples written by

Subset	GPT-4o Annotated Data	Removed Data	Removal Rate
Celebrity	6,775	294	4.3%
Interaction	5,875	72	1.2%
Position	15,950	533	3.3%
Reasoning	9,488	719	7.6%
Attribute	37,648	1,184	3.1%
Rejection	19,096	1,184	6.2%

Table 1: Statistics of our two-stage automated filtering process on the generated HumanRef-CoT data. This process removes samples with logical inconsistencies or incorrect final predictions.

<image>. A conversation between User and Assistant. The user asks a question, and the Assistant solves it. The assistant first thinks about the reasoning process in the mind and then provides the user with the answer. The reasoning process and answer are enclosed within <think> </think> and <answer> </answer> tags, respectively, i.e., <think> reasoning process here </think> <answer> answer here </answer>. Hint: Object and its coordinates in this image: **Box Hint**. User: Locate **Referring**. Assistant:

Table 2: Prompt Template for Rex-Thinker. **Box Hint** and **Referring** will be replaced with the input candidate boxes and the referring expression, respectively.

humans to illustrate the expected reasoning format. In essence, we provide GPT-4o with both the referring expression and its ground-truth answer, and prompt it to generate step-by-step reasoning in our CoT format.

**Quality Control:** To ensure the quality and logical consistency of these annotations, we implement a rigorous two-stage filtering process that goes beyond simply matching the final answer. First, we enforce internal logical coherence by verifying that the intermediate conclusions in the “Action” phase are consistent with the final “Summarization” phase. Second, we ensure final accuracy by requiring the “Summarization” outcome to perfectly match the ground-truth labels. This automated check is performed by parsing the correct and incorrect emojis generated by the model at each step. This filtering process is crucial for data quality, and as shown in Table 1, it identifies and removes a significant number of initially generated samples due to inconsistencies, especially in complex cases like the “Reasoning” subset (7.6% removal rate).

To further validate the quality of the filtered data, we conducted a manual human evaluation study (full details in Appendix A.2.2). The study confirmed the high quality of the final dataset, finding zero logical or summarization errors in a 600-sample review, with only a minor factual error rate of 1.2% in intermediate reasoning steps. This confirms our dataset provides a reliable training signal.

In total, we construct a total of 90,824 high-quality CoT annotations based on the HumanRef dataset, which we refer to as HumanRef-CoT. This diverse and large-scale dataset serves as the foundation for both our initial cold-start SFT and GRPO-based post-training.

## 4 METHOD

### 4.1 RETRIEVAL-BASED OBJECT REFERRING

To leverage the CoT-style referring data, we present Rex-Thinker, a retrieval-based model that performs object referring through explicit Chain-of-Thought reasoning.

To support explicit Chain-of-Thought (CoT) reasoning, we reformulate referring expression comprehension as a retrieval-based task. As shown in Figure 3, rather than directly regressing bounding boxes, we first use an open-vocabulary detector (Liu et al. (2024)) to extract a set of candidate object boxes corresponding to the referred object category. These candidate boxes serve as *box hints* to guide both the reasoning path and final decision of the model. This retrieval-based formulation brings two key advantages. First, during the reasoning phase, the model evaluates each candidate region in the order they appear in the input box hints (e.g., “Person 1” corresponds to the first input box). This alignment ensures that each step in the CoT trace is explicitly grounded to a specific region in the image, making the reasoning process interpretable and visually verifiable. Second, during the prediction phase, the model can directly select from the input box hints when producing the final output, thereby easing the challenge of precise coordinate regression.

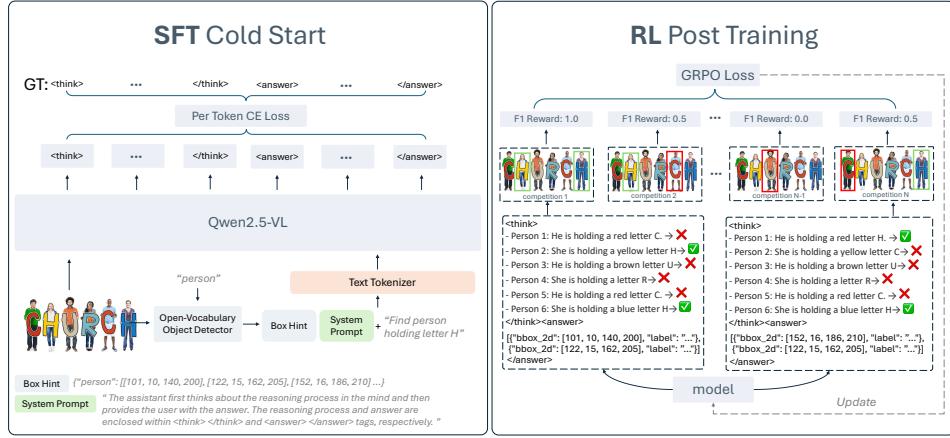


Figure 3: Overview of the Rex-Thinker architecture and our two-stage training methods

We build Rex-Thinker on top of Qwen2.5-VL-7B (Bai et al. (2025)), preserving its original architecture and using JSON-format bounding box coordinates as the final output. The model input includes the image, the box hint, the referring expression, and a system prompt that guides the reasoning process. The input prompt format is shown in Table 2.

#### 4.2 TRAINING

Our training strategy is a two-stage process, consisting of supervised fine-tuning (SFT) for cold start and GRPO-based reinforcement learning (RL) for post-training. We adopt this methodology, inspired by DeepSeek-R1 (Guo et al. (2025a)), because it has proven highly effective for teaching models to follow complex, structured reasoning paths and for subsequently refining these behaviors through reward-guided optimization.

**SFT Cold Start** We begin by fine-tuning Rex-Thinker on the HumanRef-CoT dataset to instill the ability to perform structured reasoning following our defined planning, action, and summarization format. We apply cross-entropy loss at the token level to both the reasoning trace and the final answer, providing strong supervision across the entire generation process. This stage teaches the model how to reason step-by-step in a CoT manner and also how to utilize the provided box hints to guide its final predictions.

**GRPO Post Training** While SFT teaches the model to follow our grounded CoT format, its strict token-level supervision may constrain the model to explore alternative reasoning traces and generalize beyond the training data. To enhance generalization beyond the limitations of supervised learning, we employ GRPO-based (Shao et al. (2024)) reinforcement learning for post-training. GRPO optimizes model performance by 1) sampling multiple candidate responses for each question and, 2) selectively reinforcing responses that achieve higher task-level rewards.

In our setting, given an image and a referring expression  $(I, x)$ , the model generates a group of  $G$  complete responses  $o_1, o_2, \dots, o_G$  from the current model  $\pi_\theta$ . Each response contains a full reasoning trace and a final predicted bounding box set. For each  $o_i$ , we compute a scalar reward  $r_i$  (detailed in below), and normalize these rewards to estimate group-relative advantages:

$$A_i = (r_i - \text{mean}(r_1, \dots, r_G)) / \text{std}(r_1, \dots, r_G). \quad (1)$$

Define the token-level advantage estimates  $\hat{A}_{i,t} = A_i$ , and the importance ratio at each decoding step as follows,

$$\rho_{i,t} = \frac{\pi_\theta(o_{i,t} | (I, x), x, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} | (I, x), x, o_{i,<t})}, \quad (2)$$

where  $\pi_{\theta_{\text{old}}}$  is the model before the current update. Then, the GRPO objective is given as follows,

$$\mathcal{J}_{\text{GRPO}}(\theta) = \frac{1}{G} \sum_{i=1}^G \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \left[ \min \left( \rho_{i,t} \hat{A}_{i,t}, \text{clip}(\rho_{i,t}, 1 - \epsilon, 1 + \epsilon) \hat{A}_{i,t} \right) - \beta \mathbb{D}_{\text{KL}} [\pi_\theta \| \pi_{\text{ref}}] \right] \quad (3)$$

$$\mathbb{D}_{\text{KL}} [\pi_\theta \| \pi_{\text{ref}}] = \frac{\pi_\theta(o_{i,t} | (I, x), x, o_{i,<t})}{\pi_{\text{ref}}(o_{i,t} | (I, x), x, o_{i,<t})} - \log \frac{\pi_\theta(o_{i,t} | (I, x), x, o_{i,<t})}{\pi_{\text{ref}}(o_{i,t} | (I, x), x, o_{i,<t})} - 1, \quad (4)$$

324 where  $\epsilon$  is a hyperparameter controlling the clipping range,  $\pi_{\text{ref}}$  is the model fixed after SFT stage,  
 325 and  $\beta$  is the KL penalty coefficient.  
 326

327 We argue that this formulation is suited to policy exploration in our reasoning-driven task. Given  
 328 that the model is already capable of producing structured reasoning traces after SFT, GRPO allows it  
 329 to freely explore different reasoning paths. In each iteration, the model generates diverse reasoning  
 330 strategies that may lead to different predicted object sets. The reward function then guides the model  
 331 to reinforce reasoning paths that yield accurate predictions.  
 332

333 *Accuracy Reward*: We use the F1 score to jointly evaluate the precision and recall of the model’s  
 334 predictions. Given a set of predicted boxes  $\hat{B}$  and the ground-truth set  $B^*$ , since box hints are  
 335 provided as input, we define a match only when a predicted box exactly overlaps with a ground-  
 336 truth box (i.e.,  $\text{IoU} = 1$ ), which encourages the model to select final outputs directly from the box  
 337 hints. Let  $M = \hat{B} \cap B^*$  denote the set of matched box pairs under this criterion. We compute  
 338 precision, recall, and the F1 reward as:  
 339

$$\text{Precision} = \frac{|M|}{|\hat{B}|}, \quad \text{Recall} = \frac{|M|}{|B^*|}, \quad r^{\text{F1}} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (5)$$

340 *Format Reward*: To encourage interpretable and well-structured output, we define a format reward  
 341  $r^{\text{fmt}}$  that equals 1 if the output follows the required structure: the reasoning must be enclosed in  
 342 `<think>...</think>` and the final result in `<answer>...</answer>`, and 0 otherwise.  
 343

344 The total reward is a weighted combination of the accuracy and format rewards, i.e.,  $r_i = \lambda \cdot r_i^{\text{F1}} +$   
 345  $(1 - \lambda) \cdot r_i^{\text{fmt}}$ , where  $\lambda = 0.9$  to emphasize correct detection while still enforcing output structure.  
 346

## 347 5 EXPERIMENTS

348 In this section, we evaluate the effectiveness of our CoT-based reasoning approach for object refer-  
 349 ring. We first introduce the experimental setup, then present in-domain results on the HumanRef  
 350 benchmark, followed by out-of-domain evaluation on the RefCOCOg benchmark. Lastly, we con-  
 351 duct ablation studies to analyze key design choices.  
 352

### 353 5.1 EXPERIMENTAL SETUP

355 **Model Setting.** We use Qwen2.5-VL-7B as our base model. Qwen2.5-VL outputs absolute bound-  
 356 ing box coordinates rather than quantized tokens, which provides better localization accuracy for  
 357 detection tasks. We adopt this native decoding format for final bounding box predictions.  
 358

359 **SFT Training.** We fine-tune the model on the full HumanRef-CoT dataset using supervised learn-  
 360 ing. We use a learning rate of 2e-5, weight decay of 0.01, and cosine decay scheduling. The max-  
 361 imum generation length is set to 2048 tokens. During SFT, the vision encoder and MLP projector  
 362 are frozen, and we update only the LLM parameters. For each training instance, we use all person  
 363 bounding boxes in the image as box hints.  
 364

365 **GRPO Training.** After SFT, we apply GRPO for reward-driven post-training. We continue train-  
 366 ing on HumanRef-CoT, but randomly shuffle the box hint order in each training data to create novel  
 367 input configurations. This leads the model to explore different reasoning paths than those seen dur-  
 368 ing SFT. During this phase, we train only the LLM. We use a learning rate of 1e-6, 8 rollout samples  
 369 per input, a batch size of 8, and gradient accumulation steps of 2. The KL penalty coefficient  $\beta$  is  
 370 set to 0.04, the sampling temperature to 1.0, and the output length remains 2048 tokens.  
 371

372 **Evaluation Protocol.** For in-domain evaluation, we evaluate our model on the HumanRef bench-  
 373 mark, which consists of six subsets: attribute, position, interaction, reasoning, celebrity recogni-  
 374 tion, and rejection. Following (Jiang et al. (2025b)), we report Recall (R), Precision (P), and DensityF1  
 375 (DF1) scores averaged over IoU thresholds from 0.5 to 0.95. For the rejection subset, we report  
 376 the rejection score, defined as the proportion of 1,000 images where the model correctly outputs no  
 377 bounding box when the object described by the referring expression is not present in the image. For  
 378 out-of-domain evaluation, we evaluate our model on the RefCOCOg dataset and report accuracy  
 379 at an IoU threshold of 0.5. We compare three variants: 1) Rex-Thinker-Plain, which is trained on  
 380

Method	Attribute			Position			Interaction			Reasoning			Celebrity			Average			Rejection Score
	R	P	DF1																
DINOX (Ren et al. (2024a))	59.5	28.8	20.9	78.8	28.1	17.6	67.3	28.5	18.9	76.2	32.1	22.2	94.1	48.0	37.0	75.2	33.1	23.3	36.0
InternVL-2.5-8B (Chen et al. (2025b))	23.5	39.0	27.1	23.0	28.0	24.3	27.8	40.1	31.3	17.5	22.8	18.9	57.4	59.3	58.0	29.8	37.8	31.9	54.9
Perfet-7B (You et al. (2023))	27.9	44.4	30.4	30.2	36.2	29.8	30.8	41.8	31.2	19.7	33.7	22.8	63.2	60.0	57.5	34.4	43.2	34.3	2.0
Gromma-7B (Ma et al. (2024))	67.5	47.8	38.6	63.2	43.1	37.2	66.6	48.1	40.6	59.1	41.4	34.8	73.2	63.3	59.1	65.9	48.7	42.1	0.0
ChatRex-7B (Jiang et al. (2024b))	44.3	78.0	51.8	48.0	66.7	52.5	49.6	74.8	56.5	36.6	65.4	42.8	73.7	76.5	74.2	50.4	72.2	55.6	0.0
DeepSeek-VL2-small (Wu et al. (2024))	52.3	78.0	57.7	56.4	66.1	58.1	55.4	75.7	60.7	46.6	61.7	50.1	85.9	74.3	70.7	59.3	71.2	59.5	3.1
Molimo-7B-D (Deitke et al. (2024))	82.7	86.4	76.3	78.0	80.6	72.4	69.9	77.7	66.1	72.1	80.4	65.5	85.9	87.5	82.9	77.7	82.5	72.6	<b>68.6</b>
ReSeek-7B (Jiang et al. (2025b))	87.2	86.8	81.5	86.1	86.3	83.8	<b>84.8</b>	<b>84.6</b>	<b>80.7</b>	<b>87.8</b>	<b>84.7</b>	<b>81.5</b>	83.4	86.5	84.2	<b>85.9</b>	<b>85.8</b>	<b>82.3</b>	54.1
Rex-Thinker-Plain	83.0	<b>88.7</b>	81.4	82.5	83.9	81.3	80.1	<b>85.6</b>	<b>80.2</b>	80.5	82.2	77.3	86.7	88.7	86.8	82.6	85.8	81.4	53.5
Rex-Thinker-CoT	86.6	87.7	<b>82.7</b>	<b>86.5</b>	<b>87.0</b>	<b>84.3</b>	79.6	81.7	77.2	85.7	83.8	80.3	<b>87.6</b>	<b>89.5</b>	<b>87.2</b>	85.2	<b>85.9</b>	<b>82.3</b>	67.3
Rex-Thinker-GRPO	<b>88.5</b>	<b>88.7</b>	<b>84.1</b>	<b>87.2</b>	<b>87.1</b>	<b>84.6</b>	<b>81.5</b>	83.5	79.1	<b>87.7</b>	<b>85.4</b>	<b>82.3</b>	<b>88.4</b>	<b>89.3</b>	<b>87.2</b>	<b>86.6</b>	<b>86.8</b>	<b>83.5</b>	<b>68.2</b>

Table 3: In-domain evaluation results on the HumanRef benchmark. R, P, and DF1 represent Recall, Precision, and DensityF1. The **bold** and underline fonts indicate the best and second numbers.

HumanRef-CoT using SFT only on the final detection outputs, without reasoning supervision; 2) Rex-Thinker-CoT, which is trained with SFT on both the reasoning process and the final answer; and 3) Rex-Thinker-GRPO, which is initialized from Rex-Thinker-CoT and optimized with GRPO.

## 5.2 IN-DOMAIN EVALUATION RESULTS

We begin by evaluating in-domain performance on the HumanRef benchmark to assess referring accuracy within the person domain. As shown in Table 3, Rex-Thinker-CoT, trained with structured CoT supervision, consistently outperforms Rex-Thinker-Plain across most evaluation subsets. Specifically, it achieves average improvements of +2.6 Recall, +0.1 Precision, and +0.9 DensityF1, confirming that step-by-step reasoning leads to more accurate and well-grounded predictions. Most notably, the CoT-trained model shows a remarkable 13.8 point improvement in terms of Rejection Score on the rejection subset. This metric directly quantifies the model’s trustworthiness by assessing its ability to correctly abstain when no valid target exists. This significant gain indicates substantially reduced hallucination rates, a critical capability for real-world applications requiring high reliability.

Further performance gains are realized through GRPO-based reinforcement learning. Rex-Thinker-GRPO demonstrates consistent improvements over Rex-Thinker-CoT, achieving gains of +1.4 Recall, +0.9 Precision, and +1.2 DensityF1. These results demonstrate the effectiveness of reward-based optimization in improving both the model’s reasoning process and prediction accuracy. While supervised CoT training provides strong guidance, GRPO enables dynamic exploration of alternative reasoning strategies that better optimize for task-level objectives, leading to improved performance.

## 5.3 OUT-OF-DOMAIN EVALUATION RESULTS

To evaluate the generalization of Rex-Thinker to unseen object categories, we conduct experiments on the out-of-domain RefCOCOg dataset. We first adopt a zero-shot setting: the model is trained only on HumanRef-CoT and directly evaluated on RefCOCOg. Given a referring expression and its original COCO category label, we use Grounding DINO (Liu et al. (2024)) to detect all instances of the target object category and use the detected results as box hints to Rex-Thinker.

As shown in Table 4, the CoT-trained model already performs competitively without any task-specific tuning. Further gains are achieved by applying GRPO for post training, demonstrating that reward-driven training enhances the model’s ability to generalize beyond the training domain. Interestingly, we find that Rex-Thinker maintains its structured CoT behavior even on novel categories. As illustrated in Figure 4, Rex-Thinker-GRPO successfully generalizes to detect the fish of manta ray (with “fish” bounding boxes as hints) while adhering to its *planning-action-summarization* reasoning paradigm. Notably, the model demonstrates self-correction ability: when provided with an incorrect hint label (e.g., a whale was incorrectly labeled as a “fish” in hint boxes), Rex-Thinker rectifies the error through logical reasoning and explicitly rejects the misclassification.

To further explore the upper bound of the model, we fine-tune Rex-Thinker-CoT using GRPO directly on RefCOCOg. This leads to additional performance improvements, achieving results comparable to state-of-the-art referring models. The experiment results highlight the adaptability of our reasoning paradigm across domains and the effectiveness of reward-based optimization in extending CoT reasoning to unseen categories.

432  
 433  
 434  
 435  
 436  
 437  
 438  
 439  
 440  
 441  
 442  
 443  
 444  
 445  
 446  
 447

Question: Please detect the fish of manta ray.

Answer:

<think>OK, the user needs us to detect the manta ray. In order to accomplish this task, I need to break it down into the following steps:

- Step 1: Identify each fish-like creature based on their appearance and determine if they are a manta ray. Let's start our analysis:

---

# Step 1: Identify each fish-like creature based on their appearance and determine if it is a manta ray. I see four creatures in this image. Let's look at them one by one and put a ▲ if it is a manta ray and ✗ if it is not.

- Fish 1: It is an orange and white striped fish, which is a clown fish. ✗
- Fish 2: It is a brown fish with spiky features, which is a puffer fish. ✗
- Fish 3: It has a large, wing-like structure on its back and a flat body shape, which are characteristic features of a manta ray. ▲
- Fish 4: It is a blue-colored fish with a long tail, which is a humpback whale (not a fish). ✗

---

# Summarize and Re-Check answer

Let's now recheck our answer and put ✓ for the target fish and ✗ for others.

- Fish 1: It is a clown fish, not a manta ray. ✗
- Fish 2: It is a puffer fish, not a manta ray. ✗
- Fish 3: It is a manta ray. ✓
- Fish 4: It is a humpback whale, not a fish. ✗

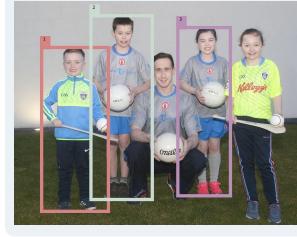
</think>

<answer>```json\n[{"bbox\_2d": [83, 894, 500, 911], "label": "fish of manta ray"}]```</answer>

Figure 4: The out-of-domain result. We use Rex-Thinker-GPRO trained on HumanRef-CoT to infer an unseen category (i.e., fish), resulting in a strong generalization. Boxes in the image denote hints.

Model	RefCOCOg	
	val	test
RexSeek-7B (Jiang et al. (2023b))	84.0	84.4
Grounding DINO (Liu et al. (2024))	86.1	87.0
QwenVL-2.5-7B (Bai et al. (2025))	87.2	87.2
ChatRex-7B (Jiang et al. (2024b))	89.8	90.0
Rex-Thinker-CoT	81.2	80.3
Rex-Thinker-GPRO	83.2	83.3
Rex-Thinker-GPRO*	89.2	88.8

Table 4: Out-of-domain evaluation results on RefCOCOg.  
 \*Fine-tuned on RefCOCOg using GPRO.



Question: Please detect the person holding a small ball

Answer:

<think>

There is **one person** holding a small ball in the image. This person is standing on the left side of the group.

</think>

<answer>

[100, 174, 369, 821],[290, 51, 538, 775],[625, 97, 839, 761]

</answer>

Figure 5: Predictions from a model that was trained with GPRO only, without CoT-based supervised fine-tuning as cold-start initialization. Boxes in the image denote answers.

#### 5.4 ABLATIONS

**Necessity of the Two Stage Architecture.** This study directly justifies our two stage detector-reasoner architecture by quantifying the inherent limitations of standalone components for REC. As shown in Table 5, neither open-set detectors nor MLLMs alone achieve robust REC performance. While powerful for open-vocabulary detection, general-purpose detectors like Grounding DINO struggle with the nuanced language comprehension required for complex REC. Even after fine-tuning on HumanRef, its performance remains significantly low (Avg. Precision: 25.7), confirming that standalone detectors are insufficient for the detailed language understanding essential for REC. Conversely, powerful MLLMs like Qwen2.5-VL-7B possess advanced reasoning but struggle with a lower recall rate, caused by pixel-imperfect object localization. For instance, fine-tuned Qwen2.5-VL-7B yields an Avg. Recall of 69.4, a substantial drop compared to that of Rex-Thinker. This demonstrates that while MLLMs are powerful reasoners, their native localization ability is a key bottleneck for REC. Thus, Rex-Thinker rigorously combines the perceptual strength of detectors (for initial candidate generation) with the cognitive strength of MLLMs (for complex reasoning and verification over these candidates), thereby addressing the individual limitations of the detector and MLLM for robust REC.

**Impact of CoT-based Cold Start on GPRO.** In Rex-Thinker, we adopt a two-stage training strategy where the model is first supervised using CoT-annotated data, followed by GPRO-based reinforcement learning. To assess the importance of this CoT-based initialization, we compare GPRO training with and without the cold-start SFT stage. We find that the model with CoT-based SFT significantly outperforms the direct GPRO model. Specifically, for models trained with GPRO, those initialized with **CoT SFT** (Avg. Precision: 86.8, Avg. Recall: 86.8, Avg. DF1: 83.5, Rejection: 68.2, please refer to the Appendix for details) achieve substantially higher final performance

Type	Method	Fine-tuned	With Box Hint	Avg. Precision	Avg. Recall	Avg. DF1	
486 Open-set Detector	Grounding DINO	No	No	15.2	87.3	9.7	
		Yes	No	25.7	92.0	14.8	
487 MLLM	Qwen2.5-VL-7B	No	No	52.5	68.5	56.2	
		Yes	No	74.1	69.4	69.6	
488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539	Two Stage (ours)	Rex-Thinker-Plain	Yes	Yes	85.7	82.6	81.4

Table 5: Limitations of standalone detectors and MLLMs for REC on the HumanRef benchmark, justifying the necessity of our two stage architecture.

compared to those trained **without CoT SFT** (Avg. Precision: 82.0, Avg. Recall: 81.2, Avg. DF1: 77.8, Rejection: 66.4). This indicates that the initial exposure to structured reasoning patterns provides a more effective starting point for reward-driven learning. Furthermore, as illustrated in Figure 5, models trained without CoT supervision tend to generate unstructured or incoherent reasoning traces, lacking the verifiable and trustworthy qualities we aim to promote. In contrast, CoT-pretrained models produce well-formed thinking steps aligned with our planning, action, and summarization framework.

## 6 CONCLUSION

We have presented Rex-Thinker, a novel framework that has reformulated the object referring problem as an explicit Chain-of-Thought reasoning process to achieve grounded and interpretable predictions. Unlike conventional approaches that have treated referring as direct bounding box prediction, our model has first detected candidate objects and then performed step-by-step verification against the referring expression through structured planning-action-summarization reasoning. To support this paradigm, we have constructed HumanRef-CoT, a large-scale dataset with reasoning traces that have enabled learning decomposed and interpretable reasoning patterns. Through a two-stage training approach combining SFT and GRPO-based RL, Rex-Thinker has demonstrated superior performance over prior works in both referring accuracy and rejection.

## REFERENCES

Pravesh Agrawal, Szymon Antoniak, Emma Bou Hanna, Devendra Chaplot, Jessica Chudnovsky, Saurabh Garg, Theophile Gervet, Soham Ghosh, Amélie Héliou, Paul Jacob, et al. Pixtral 12b. *arXiv preprint arXiv:2410.07073*, 2024.

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

Guo Chen, Zhiqi Li, Shihao Wang, Jindong Jiang, Yicheng Liu, Lidong Lu, De-An Huang, Wonmin Byeon, Matthieu Le, Tuomas Rintamaki, et al. Eagle 2.5: Boosting long-context post-training for frontier vision-language models. *arXiv preprint arXiv:2504.15271*, 2025a.

Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal llm’s referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023.

Ting Chen, Saurabh Saxena, Lala Li, David J Fleet, and Geoffrey Hinton. Pix2seq: A language modeling framework for object detection. *arXiv preprint arXiv:2109.10852*, 2021.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, Lixin Gu, Xuehui Wang, Qingyun Li, Yimin Ren, Zixuan Chen, Jiapeng Luo, Jiahao Wang, Tan Jiang, Bo Wang, Conghui He, Botian Shi, Xingcheng Zhang, Han Lv, Yi Wang, Wenqi Shao, Pei Chu, Zhongying Tu, Tong He, Zhiyong Wu, Huipeng Deng, Jiaye Ge, Kai Chen, Kaipeng Zhang, Limin Wang, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhui Wang. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling, 2025b. URL <https://arxiv.org/abs/2412.05271>.

540 Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. Yolo-world:  
 541 Real-time open-vocabulary object detection. In *Proceedings of the IEEE/CVF Conference on*  
 542 *Computer Vision and Pattern Recognition*, pp. 16901–16911, 2024.

543

544 Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V  
 545 Le, Sergey Levine, and Yi Ma. Sft memorizes, rl generalizes: A comparative study of foundation  
 546 model post-training. *arXiv preprint arXiv:2501.17161*, 2025.

547 Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Moham-  
 548 madreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open  
 549 weights and open data for state-of-the-art multimodal models. *arXiv preprint arXiv:2409.17146*,  
 550 2024.

551

552 Yihe Deng, Hritik Bansal, Fan Yin, Nanyun Peng, Wei Wang, and Kai-Wei Chang. Openvlthinker:  
 553 An early exploration to complex vision-language reasoning via iterative self-improvement. *arXiv*  
 554 *preprint arXiv:2503.17352*, 2025.

555 Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Benyou  
 556 Wang, and Xiangyu Yue. Video-r1: Reinforcing video reasoning in mllms. *arXiv preprint*  
 557 *arXiv:2503.21776*, 2025.

558

559 Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu,  
 560 Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms  
 561 via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025a.

562 Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang,  
 563 Jianyu Jiang, Jiawei Wang, et al. Seed1. 5-vl technical report. *arXiv preprint arXiv:2505.07062*,  
 564 2025b.

565

566 Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Os-  
 567 trow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint*  
 568 *arXiv:2410.21276*, 2024.

569

570 Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec  
 571 Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv*  
 572 *preprint arXiv:2412.16720*, 2024.

573

574 Qing Jiang, Feng Li, Tianhe Ren, Shilong Liu, Zhaoyang Zeng, Kent Yu, and Lei Zhang. T-rex:  
 Counting by visual prompting. *arXiv preprint arXiv:2311.13596*, 2023.

575

576 Qing Jiang, Feng Li, Zhaoyang Zeng, Tianhe Ren, Shilong Liu, and Lei Zhang. T-rex2: Towards  
 577 generic object detection via text-visual prompt synergy. In *European Conference on Computer*  
 578 *Vision*, pp. 38–57. Springer, 2024a.

579

580 Qing Jiang, Yuqin Yang, Yuda Xiong, Yihao Chen, Zhaoyang Zeng, Tianhe Ren, Lei Zhang,  
 581 et al. Chatrex: Taming multimodal llm for joint perception and understanding. *arXiv preprint*  
 582 *arXiv:2411.18363*, 2024b.

583

584 Qing Jiang, Feng Li, Zhaoyang Zeng, Tianhe Ren, Shilong Liu, and Lei Zhang. T-rex2: Towards  
 generic object detection via text-visual prompt synergy. In *European Conference on Computer*  
 585 *Vision*, pp. 38–57. Springer, 2025a.

586

587 Qing Jiang, Lin Wu, Zhaoyang Zeng, Tianhe Ren, Yuda Xiong, Yihao Chen, Qin Liu, and Lei  
 Zhang. Referring to any person, 2025b. URL <https://arxiv.org/abs/2503.08507>.

588

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

592

593 Feng Li, Qing Jiang, Hao Zhang, Tianhe Ren, Shilong Liu, Xueyan Zou, Huaizhe Xu, Hongyang Li,  
 Jianwei Yang, Chunyuan Li, et al. Visual in-context prompting. In *Proceedings of the IEEE/CVF*  
*Conference on Computer Vision and Pattern Recognition*, pp. 12861–12871, 2024a.

594 Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li.  
 595 Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv*  
 596 *preprint arXiv:2407.07895*, 2024b.

597

598 Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Li-  
 599 juan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training.  
 600 In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.  
 601 10965–10975, 2022.

602 Zhiqi Li, Guo Chen, Shilong Liu, Shihao Wang, Vibashan VS, Yishen Ji, Shiyi Lan, Hao Zhang,  
 603 Yilin Zhao, Subhashree Radhakrishnan, et al. Eagle 2: Building post-training data strategies from  
 604 scratch for frontier vision-language models. *arXiv preprint arXiv:2501.14818*, 2025.

605 Yue Liao, Si Liu, Guanbin Li, Fei Wang, Yanjie Chen, Chen Qian, and Bo Li. A real-time cross-  
 606 modality correlation filtering method for referring expression comprehension. In *Proceedings of*  
 607 *the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10880–10889, 2020.

608

609 Zhenyi Liao, Qingsong Xie, Yanhao Zhang, Zijian Kong, Haonan Lu, Zhenyu Yang, and  
 610 Zhijie Deng. Improved visual-spatial reasoning via r1-zero-like training. *arXiv preprint*  
 611 *arXiv:2504.00883*, 2025.

612 Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*,  
 613 2023.

614

615 Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan  
 616 Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training  
 617 for open-set object detection. In *European Conference on Computer Vision*, pp. 38–55. Springer,  
 618 2024.

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

622

623 Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi  
 624 Wang. Visual-rft: Visual reinforcement fine-tuning. *arXiv preprint arXiv:2503.01785*, 2025b.

625 Chuofan Ma, Yi Jiang, Jiannan Wu, Zehuan Yuan, and Xiaojuan Qi. Groma: Localized visual  
 626 tokenization for grounding multimodal large language models. *arXiv preprint arXiv:2404.13013*,  
 627 2024.

628

629 Xinyu Ma, Ziyang Ding, Zhicong Luo, Chi Chen, Zonghao Guo, Derek F Wong, Xiaoyi Feng,  
 630 and Maosong Sun. Deepperception: Advancing r1-like cognitive visual perception in mllms for  
 631 knowledge-intensive visual grounding. *arXiv preprint arXiv:2503.12797*, 2025.

632

633 Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L. Yuille, and Kevin Mur-  
 634 phy. Generation and comprehension of unambiguous object descriptions. In *CVPR*, pp. 11–20,  
 2016.

635

636 Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter,  
 637 Dhruti Shah, Xianzhi Du, Futang Peng, Floris Weers, Anton Belyi, Haotian Zhang, Karanjeet  
 638 Singh, Doug Kang, Ankur Jain, Hongyu Hè, Max Schwarzer, Tom Gunter, Xiang Kong, Aonan  
 639 Zhang, Jianyu Wang, Chong Wang, Nan Du, Tao Lei, Sam Wiseman, Guoli Yin, Mark Lee,  
 640 Zirui Wang, Ruoming Pang, Peter Grasch, Alexander Toshev, and Yinfei Yang. MM1: methods,  
 641 analysis & insights from multimodal LLM pre-training. *arXiv: 2403.09611*, 2024.

642

643 Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey  
 644 Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. Sim-  
 645 ple open-vocabulary object detection. In *European conference on computer vision*, pp. 728–755.  
 Springer, 2022.

646

647 Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke  
 648 Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time  
 scaling, 2025. URL <https://arxiv.org/abs/2501.19393>.

648 OpenAI. Gpt-4v(ision) system card. [https://cdn.openai.com/papers/GPTV\\_System\\_Card.pdf](https://cdn.openai.com/papers/GPTV_System_Card.pdf), 2023.

649

650

651 OpenAI, :, Ahmed El-Kishky, Alexander Wei, Andre Saraiva, Borys Minaiev, Daniel Selsam,  
652 David Dohan, Francis Song, Hunter Lightman, Ignasi Clavera, Jakub Pachocki, Jerry Tworek,  
653 Lorenz Kuhn, Lukasz Kaiser, Mark Chen, Max Schwarzer, Mostafa Rohaninejad, Nat McAleese,  
654 o3 contributors, Oleg Mürk, Rhythm Garg, Rui Shu, Szymon Sidor, Vineet Kosaraju, and  
655 Wenda Zhou. Competitive programming with large reasoning models, 2025. URL <https://arxiv.org/abs/2502.06807>.

656

657 Yi Peng, Xiaokun Wang, Yichen Wei, Jiangbo Pei, Weijie Qiu, Ai Jian, Yunzhuo Hao, Jiachun  
658 Pan, Tianyidan Xie, Li Ge, et al. Skywork r1v: Pioneering multimodal reasoning with chain-of-  
659 thought. *arXiv preprint arXiv:2504.05599*, 2025.

660

661 Yanyuan Qiao, Chaorui Deng, and Qi Wu. Referring expression comprehension: A survey of meth-  
662 ods and datasets. *IEEE Transactions on Multimedia*, 23:4426–4440, 2020.

663

664 Tianhe Ren, Yihao Chen, Qing Jiang, Zhaoyang Zeng, Yuda Xiong, Wenlong Liu, Zhengyu Ma,  
665 Junyi Shen, Yuan Gao, Xiaoke Jiang, et al. Dino-x: A unified vision model for open-world object  
666 detection and understanding. *arXiv preprint arXiv:2411.14347*, 2024a.

667

668 Tianhe Ren, Qing Jiang, Shilong Liu, Zhaoyang Zeng, Wenlong Liu, Han Gao, Hongjie Huang,  
669 Zhengyu Ma, Xiaoke Jiang, Yihao Chen, et al. Grounding dino 1.5: Advance the" edge" of  
670 open-set object detection. *arXiv preprint arXiv:2405.10300*, 2024b.

671

672 Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang,  
673 Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathe-  
674 matical reasoning in open language models, 2024. URL <https://arxiv.org/abs/2402.03300>.

675

676 Haozhan Shen, Peng Liu, Jingcheng Li, Chunxin Fang, Yibo Ma, Jiajia Liao, Qiaoli Shen, Zilun  
677 Zhang, Kangjia Zhao, Qianqian Zhang, et al. Vlm-r1: A stable and generalizable r1-style large  
678 vision-language model. *arXiv preprint arXiv:2504.07615*, 2025.

679

680 Mingjie Sun, Jimin Xiao, and Eng Gee Lim. Iterative shrinking for referring expression grounding  
681 using deep reinforcement learning. In *Proceedings of the IEEE/CVF conference on computer  
vision and pattern recognition*, pp. 14060–14069, 2021.

682

683 Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun  
684 Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with  
685 llms. *arXiv preprint arXiv:2501.12599*, 2025.

686

687 Omkar Thawakar, Dinura Dissanayake, Ketan More, Ritesh Thawkar, Ahmed Heakl, Noor Ahsan,  
688 Yuhao Li, Mohammed Zumri, Jean Lahoud, Rao Muhammad Anwer, et al. Llamav-o1: Rethink-  
689 ing step-by-step visual reasoning in llms. *arXiv preprint arXiv:2501.06186*, 2025.

690

691 Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu,  
692 Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the  
693 world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

694

695 Weihang Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang,  
696 Lei Zhao, Xixuan Song, et al. Cogvlm: Visual expert for pretrained language models. *arXiv  
697 preprint arXiv:2311.03079*, 2023.

698

699 Yichen Wei, Yi Peng, Xiaokun Wang, Weijie Qiu, Wei Shen, Tianyidan Xie, Jiangbo Pei, Jianhao  
700 Zhang, Yunzhuo Hao, Xuchen Song, et al. Skywork r1v2: Multimodal hybrid reinforcement  
701 learning for reasoning. *arXiv preprint arXiv:2504.16656*, 2025.

702

703 Size Wu, Wenwei Zhang, Sheng Jin, Wentao Liu, and Chen Change Loy. Aligning bag of regions  
704 for open-vocabulary object detection. In *Proceedings of the IEEE/CVF conference on computer  
705 vision and pattern recognition*, pp. 15254–15264, 2023.

702 Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang  
 703 Ma, Chengyue Wu, Bingxuan Wang, et al. Deepseek-vl2: Mixture-of-experts vision-language  
 704 models for advanced multimodal understanding. *arXiv preprint arXiv:2412.10302*, 2024.

705  
 706 Violet Xiang, Charlie Snell, Kanishk Gandhi, Alon Albalak, Anikait Singh, Chase Blagden, Duy  
 707 Phung, Rafael Rafailov, Nathan Lile, Dakota Mahan, Louis Castricato, Jan-Philipp Franken, Nick  
 708 Haber, and Chelsea Finn. Towards system 2 reasoning in llms: Learning how to think with meta  
 709 chain-of-thought, 2025. URL <https://arxiv.org/abs/2501.04682>.

710 Wei Xiong, Hanning Zhang, Chenlu Ye, Lichang Chen, Nan Jiang, and Tong Zhang. Self-  
 711 rewarding correction for mathematical reasoning, 2025. URL <https://arxiv.org/abs/2502.19613>.

712  
 713 Guowei Xu, Peng Jin, Li Hao, Yibing Song, Lichao Sun, and Li Yuan. Llava-o1: Let vision language  
 714 models reason step-by-step. *arXiv preprint arXiv:2411.10440*, 2024.

715  
 716 Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark  
 717 prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*,  
 718 2023a.

719  
 720 Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark  
 721 prompting unleashes extraordinary visual grounding in gpt-4v, 2023b. URL <https://arxiv.org/abs/2310.11441>.

722  
 723 Sibei Yang, Guanbin Li, and Yizhou Yu. Dynamic graph attention for referring expression com-  
 724 prehension. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp.  
 725 4644–4653, 2019.

726  
 727 Yi Yang, Xiaoxuan He, Hongkun Pan, Xiyan Jiang, Yan Deng, Xingtao Yang, Haoyu Lu, Dacheng  
 728 Yin, Fengyun Rao, Minfeng Zhu, et al. R1-onevision: Advancing generalized multimodal rea-  
 729 soning through cross-modal formalization. *arXiv preprint arXiv:2503.10615*, 2025.

730  
 731 Lewei Yao, Jianhua Han, Youpeng Wen, Xiaodan Liang, Dan Xu, Wei Zhang, Zhenguo Li, Chunjing  
 732 Xu, and Hang Xu. Detclip: Dictionary-enriched visual-concept paralleled pre-training for open-  
 733 world detection. *Advances in Neural Information Processing Systems*, 35:9125–9138, 2022a.

734  
 735 Yuan Yao, Ao Zhang, Zhengyan Zhang, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. Cpt: Col-  
 736 orful prompt tuning for pre-trained vision-language models, 2022b. URL <https://arxiv.org/abs/2109.11797>.

737  
 738 Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao,  
 739 Shih-Fu Chang, and Yinfei Yang. Ferret: Refer and ground anything anywhere at any granularity.  
 740 *arXiv preprint arXiv:2310.07704*, 2023.

741  
 742 En Yu, Kangheng Lin, Liang Zhao, Jisheng Yin, Yana Wei, Yuang Peng, Haoran Wei, Jianjian Sun,  
 743 Chunrui Han, Zheng Ge, et al. Perception-r1: Pioneering perception policy with reinforcement  
 744 learning. *arXiv preprint arXiv:2504.07954*, 2025.

745  
 746 Licheng Yu, Patrick Poirson, Shan Yang, Alexander C. Berg, and Tamara L. Berg. Modeling context  
 747 in referring expressions. In *ECCV*, volume 9906, pp. 69–85, 2016.

748  
 749 Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. Mat-  
 750 ttnet: Modular attention network for referring expression comprehension. In *Proceedings of the*  
 751 *IEEE conference on computer vision and pattern recognition*, pp. 1307–1315, 2018.

752  
 753 Alireza Zareian, Kevin Dela Rosa, Derek Hao Hu, and Shih-Fu Chang. Open-vocabulary object  
 754 detection using captions. In *Proceedings of the IEEE/CVF conference on computer vision and*  
 755 *pattern recognition*, pp. 14393–14402, 2021.

756  
 757 Yufei Zhan, Yousong Zhu, Hongyin Zhao, Fan Yang, Ming Tang, and Jinqiao Wang. Griffon v2:  
 758 Advancing multimodal perception with high-resolution scaling and visual-language co-referring.  
 759 *arXiv preprint arXiv:2403.09333*, 2024.

756 Yufei Zhan, Yousong Zhu, Zhiyang Chen, Fan Yang, Ming Tang, and Jinqiao Wang. Griffon:  
757 Spelling out all object locations at any granularity with large language models. In *European*  
758 *Conference on Computer Vision*, pp. 405–422. Springer, 2025.

759

760 Chao Zhang, Weiming Li, Wanli Ouyang, Qiang Wang, Woo-Shik Kim, and Sunghoon Hong. Re-  
761 ferring expression comprehension with semantic visual relationship and word mapping. In *Pro-*  
762 , pp. 1258–1266, 2019.

763

764 Haotian Zhang, Haoxuan You, Philipp Dufter, Bowen Zhang, Chen Chen, Hong-You Chen, Tsu-Jui  
765 Fu, William Yang Wang, Shih-Fu Chang, Zhe Gan, et al. Ferret-v2: An improved baseline for  
766 referring and grounding with large language models. *arXiv preprint arXiv:2404.07973*, 2024.

767

768 Jingyi Zhang, Jiaxing Huang, Huanjin Yao, Shunyu Liu, Xikun Zhang, Shijian Lu, and Dacheng  
769 Tao. R1-vl: Learning to reason with multimodal large language models via step-wise group  
770 relative policy optimization. *arXiv preprint arXiv:2503.12937*, 2025.

771

772 Zhipeng Zhang, Zhimin Wei, Zhongzhen Huang, Rui Niu, and Peng Wang. One for all: One-stage  
773 referring expression comprehension with dynamic reasoning. *Neurocomputing*, 518:523–532,  
774 2023.

775

776

777

778

779

780

781

782

783

784

785

786

787

788

789

790

791

792

793

794

795

796

797

798

799

800

801

802

803

804

805

806

807

808

809

810 A APPENDIX  
811812 A.1 USE OF LARGE LANGUAGE MODELS  
813

814 In preparing this manuscript, we made limited use of a large language model (LLM) to assist with  
815 language editing. Specifically, the LLM was employed only to improve grammar, clarity, and word-  
816 ing of sentences. No part of the scientific content, analysis, or claims was generated by the LLM.  
817 The authors take full responsibility for all aspects of the paper’s content.

818 A.2 MORE DETAILS ON CONSTRUCTING HUMANREF-CoT  
819820 A.2.1 PROMPT FOR GPT-4o  
821

822 To annotate HumanRef-CoT dataset using GPT-4o, we designed a two-part prompting strategy that  
823 addresses the diverse reasoning requirements across different subsets. This strategy consists of a  
824 **unified system prompt** and a set of **subset-specific in-context examples**.

825 The system prompt is shared across all subsets and instructs the model on how to interpret the input,  
826 which includes an image, a referring expression, and candidate bounding boxes. It also defines  
827 the expected format of the response, including the use of structured reasoning and answer tags. In  
828 addition to the system prompt, each of the six subsets in HumanRef-CoT namely attribute, position,  
829 interaction, reasoning, celebrity recognition, and rejection, is paired with a collection of in-context  
830 examples. These examples are carefully curated to reflect the specific annotation challenges and  
831 reasoning patterns required for each subset. They guide GPT-4o in producing chain-of-thought  
832 (CoT) rationales that are consistent with human annotations in both style and logic.

833 In the following sections, we first present the shared system prompt. Then, for each subset, we  
834 provide the corresponding in-context examples and visualization results.  
835

836 **Unified System Prompt.** The system prompt instructs the model to perform detailed visual rea-  
837 soning based on either positional or attribute-based referring expressions. It emphasizes step-by-step  
838 analysis, beginning with predefined reasoning steps (first attributes, then orientation), and requires  
839 the model to explicitly evaluate each candidate object. Special symbols are also used to denote  
840 matching, non-matching, and reference entities during analysis.

841 **Subset-Specific In-Context Examples.** After the system prompt, we provide in-context examples  
842 to guide the model toward producing outputs aligned with our CoT structure. These examples help  
843 reinforce consistent reasoning patterns. HumanRef-CoT includes six subsets: attribute, position,  
844 interaction, reasoning, celebrity recognition, and rejection. Each subset uses its own set of in-context  
845 examples tailored to its specific reasoning needs.

846 We show the in-context prompts used for each subset, along with representative outputs generated  
847 by GPT-4o.  
848

Subset	attribute	position		interaction		reasoning		celebrity	rejection
		inner position	outer position	inner interaction	outer interaction	inner reasoning	attribute reasoning		
Prompt Example	Figure 34 Figure 35	Figure 36 Figure 37	Figure 38 Figure 39	Figure 40 Figure 41	Figure 42 Figure 43	Figure 44 Figure 45	Figure 46 Figure 47	Figure 48 Figure 49	Figure 50 Figure 51

850 Table 6: Ablation study on the retrieval-based design of our model. We compare performance with  
851 and without box hints to assess their impact on referring accuracy.  
852

## 853 A.2.2 HUMAN EVALUATION OF HUMANREF-CoT DATASET QUALITY

854 To empirically validate the quality of our HumanRef-CoT dataset and assess the effectiveness of our  
855 automated filtering pipeline, we conducted a comprehensive manual evaluation study. This study  
856 was designed to verify the logical consistency and factual correctness of the generated Chain-of-  
857 Thought annotations that were used for model training.

858 **Methodology:** 1) *Data Sampling:* We randomly sampled 600 instances from the final, filtered  
859 HumanRef-CoT dataset. To ensure representative coverage, we drew 100 samples from each of the

Given the image and the description below, output a detailed analysis of how you found the object(s) matching the description about position or attribute.

The position description is something like “the third man to the left of the boy wearing a red shirt”. Then you first need to find the reference person, i.e. the boy wearing a red shirt, determine for each person of what you see and whether fulfill this reference condition, and put a ▲ if he is the reference person or object, and ✗ if he doesn't. If this question is about attribute, you should provide a description of what you see and how each object relates to the provided description, and put a ✗ or ✓ at the end. You first need to do your analysis in a tuple like (analysis)(analysis), and then output your answer in a tuple like (answer)(answer). Note that in your analysis, you need to start by listing your action steps. The action steps must be about attribute first, then orientation. And your first step doesn't need to be to find all the people in the diagram, since I've already provided you with all of them. For example:

...

To find the woman in red dress, I will need to excuse the following steps:

- Step1: Find all woman
- Step2: From the person in step1, I need to find all the person wearing red dress

...

Each step you initially planned must be strictly enforced, and you cannot omit a step or modify the execution of each step.

Note that I will tell you which objects are the ground truth that fit this description, you need to use the answer I give as a reference. But you can't refer to the answer I gave in your answer, pretend that you are thinking about it yourself. I will highlight all such objects as marks in the diagram. Each mark has a circle and a number. The number represents the serial number of the object, and the color of the circle represents whether or not the current object matches the description, with green circles representing objects that match the description, and red circles representing objects that do not match the description. Note that you can't mention this mark in your answer, this mark is just for you to go and specify the corresponding person with the corresponding serial number, you can say person 1, but not mark 1.

Your final answer must be consistent with the analysis, e.g. if you say in the analysis that Person 1 and Person 2 satisfy the condition, then you must also say in your final answer that it is Person 1 and Person 2. In the rare case that I give a reference answer that is incorrect, you need to trust your own judgment.

Additionally I'll calculate the xy coordinates of each mark and then sort them in order from left to right to give you a positional reference if the question is about position. But you can't mention this order, pretend that you figure out the order all by yourself. Every useful information will be provided in METAINFO

INCONTEXT EXAMPLES

Figure 6: The system prompt used to instruct GPT-4o on visual reasoning for HumanRef-CoT. It specifies output format, reasoning steps, symbol conventions, and the expected alignment between intermediate analysis and final answers.

six distinct subsets (Celebrity, Interaction, Position, Reasoning, Attribute, and Rejection) 2) *Annotators*: The annotation task was performed by five Ph.D. students with expertise in computer vision and natural language processing. All annotators were given detailed instructions and calibration examples to ensure consistent evaluation criteria. 3) *Annotation Interface*: e developed a custom HTML-based annotation interface to facilitate the review process. As shown in Figure 7, the interface presented annotators with the source image (including the Set-of-Mark visual markers), the referring expression, and the complete, generated CoT trace (Planning, Action, and Summarization). This allowed for a holistic review of each data point in its full context.

**Error Categories:** Annotators were instructed to identify three specific types of errors, ordered from most to least severe:

- **Wrong Summarization:** The final set of person indices identified in the <answer> block does not perfectly match the ground-truth labels.
- **Action-Summarization Inconsistency:** A logical contradiction exists between the step-by-step evaluation in the “Action”phase and the final conclusion in the “Summarization” phase. For example, the Action phase concludes a person is a non-match, but the Summarization phase incorrectly includes them.
- **Wrong Action Result:** A specific reasoning step in the “Action” phase is factually incorrect with respect to the visual evidence. For example, the model incorrectly identifies the color of a person's clothing or misjudges a spatial relationship.

**Results and Analysis** The results of our comprehensive human evaluation on the 600 samples are summarized in Table 7. First, our automated, two-stage filtering process is extremely effective at eliminating high-level logical and summarization errors. The human evaluators found zero instances of Wrong Summarization or Action-Summarization Inconsistency. This provides strong evidence that the dataset is reliable in its final conclusions and overall logical structure. Second, we findw

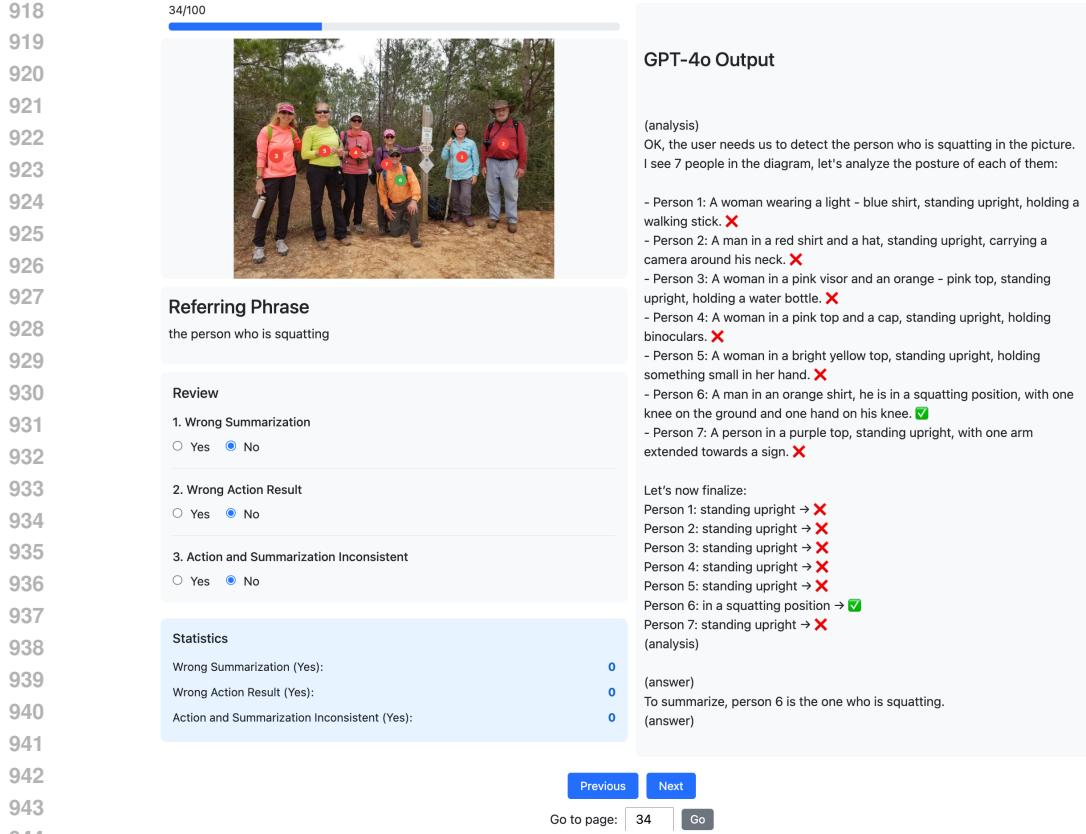


Figure 7: A screenshot of our custom human evaluation interface. The tool displays the image with visual markers, the referring expression, and the full CoT trace, enabling annotators to efficiently verify each reasoning step.

Error Type	Celebrity	Interaction	Position	Reasoning	Attribute	Rejection
Wrong Summarization	0	0	0	0	0	0
Action and Summarization Inconsistent	0	0	0	0	0	0
Wrong Action Result	2	2	0	3	0	0

Table 7: Results of the human evaluation on 600 randomly sampled instances from the HumanRef-CoT dataset. No high-level logical or summarization errors were found.

that a small number of low-level Wrong Action Result errors persist in the final dataset (7 out of 600 total samples, or 1.2%). Upon manual inspection of these cases, we found they typically occur when a referred person is very small, heavily occluded, or when the visual marker itself obscures a key attribute, forcing GPT-4o to make a reasonable but incorrect guess. Despite these minor, infrequent imperfections in intermediate steps, the overall quality of the dataset is very high. The significant performance improvement of our model when trained with this data demonstrates that it provides a valuable and effective training signal for learning grounded, step-by-step reasoning.

### A.2.3 EVALUATE GPT-4O ON HUMANREF

Since we use GPT-4o to annotate HumanRef-CoT, a natural question is how well GPT-4o performs directly on the HumanRef benchmark when prompted in a similar style. To investigate this, we adopt a setup similar to the annotation phase, using the same SoM-style prompt and a set of visual marks (with all marks shown in red). However, we remove any hint indicating which objects are correct. We then evaluate GPT-4o on the HumanRef-Benchmark without prompting with ground-truth answers. As shown in Table 8, GPT-4o achieves an average DF1 score of 53.2 without any hint supervision. This result suggests that while GPT-4o can be used to generate annotations when given

972 the correct answer as reference, its standalone performance without answer supervision remains  
 973 limited.  
 974

Method	Attribute			Position			Interaction			Reasoning			Celebrity			Average			Rejection Score
	R	P	DF1																
GPT4o-CoT	50.2	56.2	50.9	56.1	56.8	55.1	52.8	56.8	53.2	53.3	52.9	51.1	54.9	54.3	53.2	54.3	55.2	53.2	14.8
Rex-Thinker-GRPO	<b>88.5</b>	<b>88.7</b>	<b>84.1</b>	<b>87.2</b>	<b>87.1</b>	<b>84.6</b>	<b>81.5</b>	<b>83.5</b>	<b>79.1</b>	<b>87.7</b>	<b>85.4</b>	<b>82.3</b>	<b>88.0</b>	<b>89.3</b>	<b>87.2</b>	<b>86.6</b>	<b>86.8</b>	<b>83.5</b>	<b>68.2</b>

975  
 976  
 977  
 978 Table 8: Evaluation of GPT-4o on the HumanRef-Benchmark test set using SoM-style prompts  
 979 without answer hints. The model achieves 53.2 average DF1 score, indicating limited standalone  
 980 performance.  
 981

#### A.2.4 ETHICAL CONSIDERATIONS AND DATASET AVAILABILITY

982 This section details the ethical safeguards employed during the creation of the HumanRef-CoT  
 983 dataset and outlines our commitment to its public release to the research community.  
 984

985 **Ethical Considerations and Safeguards** We acknowledge the critical importance of responsible  
 986 and ethical data creation. Our data generation pipeline was designed with a multi-layered approach  
 987 to mitigate the risk of generating biased or problematic content.  
 988

- 989 • **Filtered Image Source:** The source images for HumanRef-CoT are from the public HumanRef  
 990 dataset. These images have undergone prior NSFW (Not Safe For Work) filtering, ensuring that  
 991 the visual content is appropriate and does not contain sensitive material.  
 992
- 993 • **Use of a Moderated Large Language Model:** Our data generation process utilizes OpenAI’s  
 994 GPT-4o to produce the reasoning traces. As a state-of-the-art commercial model, GPT-4o is  
 995 subject to rigorous safety protocols and content moderation filters developed by its provider.  
 996 These built-in guardrails are designed to prevent the generation of offensive, biased, or otherwise  
 997 harmful content and served as a primary safeguard in our pipeline.  
 998
- 999 • **Final Data Review:** Throughout our automated and manual quality control stages (detailed in  
 1000 Appendix A.5), where we verified the logical and factual correctness of the reasoning traces, we  
 1001 also remained vigilant for any inappropriate content. We can confirm that we did not encounter  
 1002 any instances of ethically questionable language in the final, curated HumanRef-CoT dataset.  
 1003

1004 **Dataset Availability** To ensure full reproducibility and to foster future research in grounded reasoning  
 1005 and model interpretability, we are committed to making our dataset publicly available. The  
 1006 HumanRef-CoT dataset will be released upon the publication of this work.  
 1007

### A.3 EXPERIMENT DETAILS

#### A.3.1 COT SFT SETTINGS

1008 Table 9 summarizes the full training hyperparameters and computational cost used during the CoT  
 1009 SFT stage. These settings were applied in the cold-start phase without prior instruction tuning.  
 1010

batch size	4	maximum gradient norm	1	precision	bf16
gradient accumulation	4	learning rate scheduler	cosine	epochs	2
learning rate	2e-5	max length	2048	times	10.1h
optimizer	AdamW	deepspeed	zero3	GPU	8xA100
warm up ratio	0.03	weight decay	0.01	trainable module	LLM

1011  
 1012 Table 9: Training settings and cost statistics for CoT SFT.  
 1013

#### A.3.2 GRPO SETTINGS

1014 We provide the training configurations used during the GRPO stage in Table 15. We did not run  
 1015 full GRPO training on the entire HumanRef-CoT dataset. Instead, training was terminated when the  
 1016 reward signal plateaued, indicating convergence.  
 1017

#### A.3.3 GRPO TRAINING ANALYSIS

1018 We analyze the training logs of the GRPO stage. As shown in Figure 8, we visualize the changes in  
 1019 both reward signals and completion length throughout training.  
 1020

1026	batch size	8	num of rollout	8	precision	bf16
1027	gradient accumulation	2	$\beta$	0.04	epochs	0.25
1028	learning rate	1e-6	temperature	1.0	times	112h
1029	optimizer	AdamW	deepspeed	zero3	GPU	8xA100
1030	warm up ratio	0.03	weight decay	0.01	trainable module	LLM

Table 10: Hyperparameters used during the GRPO training stage.

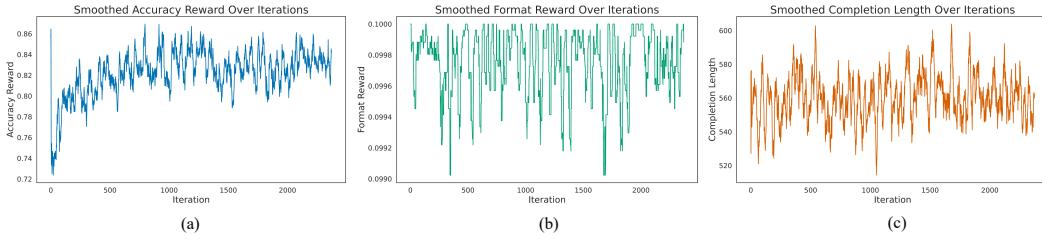


Figure 8: GRPO training curves showing accuracy reward, format reward, and completion length over time.

Thanks to the cold-start CoT initialization, the model achieves a reasonably high accuracy reward at the beginning of GRPO training. At the same time, the format reward is nearly saturated from the start, indicating that the model has already learned to follow the correct output structure after CoT supervision. Meanwhile, the completion length remains stable at around 560 tokens throughout training. We attribute this to the model having already acquired the basic reasoning skills required for the referring task during the CoT fine-tuning phase, resulting in consistent output lengths with minimal fluctuation.

### A.3.4 DETAILED EVALUATION SETTINGS

This section provides a detailed breakdown of the experimental settings and evaluation contexts for all baseline models.

**In-domain Evaluation Settings:** For the in-domain evaluation on the HumanRef benchmark, as presented in Table 3 of the main paper, the baseline models fall into two distinct categories:

- **Zero-Shot Evaluation:** The majority of the compared models (e.g., DINOX, InternVL-2.5-8B, Ferret-7B, Groma-7B, ChatRex-7B, Qwen2.5-VL-7B, etc.) were evaluated in a zero-shot setting. We used their publicly available, official checkpoints to generate predictions directly on the HumanRef test set without any fine-tuning on our dataset. This setup is designed to measure their out-of-the-box generalization capability to our challenging, human-centric scenarios.
- **State-of-the-Art In-Domain Model:** The RexSeek-7B model is the only baseline that was specifically trained on the HumanRef training set. Crucially, it was not trained using any form of Chain-of-Thought (CoT) reasoning, as it is a direct-prediction model. We report the performance numbers directly from the original RexSeek paper, as this represents the current state-of-the-art for non-CoT methods on this benchmark.

**Out-of-domain Evaluation Settings:** For the out-of-domain evaluation on the RefCOCOg dataset, as presented in Table 4 of the main paper, the comparison settings are:

- **Baseline Models (Supervised):** The reported numbers for all baseline models (RexSeek-7B, Grounding DINO, QwenVL-2.5-7B, ChatRex-7B) are taken directly from their original papers. As per their publications, all of these models were trained on the RefCOCOg training set. Furthermore, none of them are based on a CoT framework. They represent the state-of-the-art for direct-prediction or retrieval-based methods on this standard benchmark.
- **Our Model (Zero-Shot):** It is critical to note that our Rex-Thinker models are evaluated in a strict zero-shot setting on this dataset. They were trained only on the HumanRef-CoT dataset and had no exposure to RefCOCOg data or its object categories during training. This setup specifically tests the generalization of our learned reasoning framework.

1080 A.3.5 DETAILED METRICS FOR ABLATION STUDIES  
1081

1082 As shown in Table 11, the without-hint model is naive Qwen2.5-VL-7B, while the with-hint model  
1083 is Rex-Thinker-Plain. It is seen that the box hint plays an important role in REC accuracy. As  
1084 shown in Table 12, CoT-based SFT as a cold start is an important pre-procedure for RL exploration,  
1085 leading to significant improvement in REC accuracy and rejection. This is also evidence that our  
1086 planning-action-summarization paradigm is helpful for REC.

With Box Hint	Attribute			Position			Interaction			Reasoning			Celebrity			Average			Rejection Score
	R	P	DF1																
No	66.4	74.3	67.2	69.3	71.9	69.5	65.2	72.1	66.4	63.6	67.5	62.2	82.4	84.6	82.7	69.4	74.1	69.6	71.7
Yes	<b>83.0</b>	<b>88.7</b>	<b>81.4</b>	<b>82.5</b>	<b>83.9</b>	<b>81.3</b>	<b>80.1</b>	<b>85.6</b>	<b>80.2</b>	<b>80.5</b>	<b>82.2</b>	<b>77.3</b>	<b>86.7</b>	<b>88.7</b>	<b>86.8</b>	<b>82.6</b>	<b>85.8</b>	<b>81.4</b>	53.5

1091 Table 11: Ablation study on the retrieval-based design of our model. We compare performance with  
1092 and without box hints to assess their impact on referring accuracy.

With Cold Start	Attribute			Position			Interaction			Reasoning			Celebrity			Average			Rejection Score
	R	P	DF1																
No	81.4	85.8	78.1	80.2	80.2	77.5	79.6	82.6	78.0	77.6	75.0	70.6	87.3	86.5	84.8	81.2	82.0	77.8	66.4
Yes	<b>88.5</b>	<b>88.7</b>	<b>84.1</b>	<b>87.2</b>	<b>87.1</b>	<b>84.6</b>	<b>81.5</b>	<b>83.5</b>	<b>79.1</b>	<b>87.7</b>	<b>85.4</b>	<b>82.3</b>	<b>88.0</b>	<b>89.3</b>	<b>87.2</b>	<b>86.6</b>	<b>86.8</b>	<b>83.5</b>	<b>68.2</b>

1093 Table 12: Ablation on the impact of CoT-based cold start on final performance after GRPO training.

1094 A.3.6 COMPARISON WITH THE THINK-WITH-IMAGE PARADIGM  
1095

Method	Attribute			Position			Interaction			Reasoning			Celebrity			Average			Rejection Score
	R	P	DF1	R	P	DF1	R	P	DF1	R	P	DF1	R	P	DF1	R	P	DF1	
Qwen2.5-VL-7B	49.1	71.3	54.4	50.2	61.7	52.8	48.2	66.3	53.2	34.6	61.2	40.3	80.3	81.9	80.1	52.5	68.5	56.2	7.1
DeepEyes-7B	36.7	60.1	41.6	34.6	43.7	36.1	40.2	56.9	44.6	28.4	40.7	30.8	40.2	40.9	40.2	36.0	48.4	38.7	27.3
Rex-Thinker-GRPO	88.5	88.8	84.1	87.3	87.8	85.1	82.1	83.6	79.4	87.0	84.3	81.3	88.6	90.4	88.1	86.7	87.0	83.6	67.8

1096 Table 13: Detailed performance comparison on the HumanRef benchmark. Our Rex-Thinker-GRPO  
1097 significantly outperforms DeepEyes, a representative "Think-with-image" model.

1098 We evaluated DeepEyes-7B, a representative model from the "Think-with-image" paradigm, on the  
1099 HumanRef benchmark. DeepEyes is designed for complex visual reasoning, where its Chain-of-  
1100 Thought process can invoke tools to interactively modify the image, such as by zooming in or crop-  
1101 ping specific regions to gather more detailed information. The detailed results of our comparison  
1102 are presented in Table 13.

1103 The quantitative results reveal a significant performance gap. Our Rex-Thinker-GRPO outperforms  
1104 DeepEyes by a large margin across all categories, most notably in Average Recall (86.7 vs. 36.0)  
1105 and Average DF1 (83.6 vs. 38.7). This disparity points to a fundamental limitation in applying a  
1106 general visual reasoning paradigm to the specific problem of REC.

1107 Our investigation into the performance gap reveals that it stems from a core methodological dif-  
1108 ference between our framework and the "Think-with-image" approach. As illustrated in Figure 9,  
1109 DeepEyes employ a holistic, tool-augmented reasoning strategy. The model identifies a few salient  
1110 regions of interest and may invoke tools (e.g., cropping, zooming) to inspect these specific regions  
1111 in greater detail before generating an answer. While this interactive approach is powerful for general  
1112 visual reasoning tasks, it is ill-suited for REC, especially for multi-instance referring. We observed  
1113 that after investigating only one promising candidate, the model frequently concludes its reasoning  
1114 process prematurely, leading to a high number of false negatives and thus very low recall.

1115 A.3.7 FINE-GRAINED PERFORMANCE ANALYSIS  
1116

1117 To provide a more nuanced understanding of our model's performance and the specific impact of the  
1118 Chain-of-Thought framework, we conducted a fine-grained analysis on the HumanRef benchmark.  
1119 This analysis compares our full model, Rex-Thinker-GRPO, against a baseline version trained with  
1120 supervised fine-tuning but without the structured CoT data (Rex-Thinker-Plain)

1121 The primary finding of this analysis, detailed in Table 14, is that the CoT framework provides a sub-  
1122 stantial and consistent boost in performance, particularly in Recall. This effect is most pronounced  
1123 in the benchmark's most complex categories, such as Reasoning (+5.29 Recall) and Attribute (+4.48  
1124 Recall).

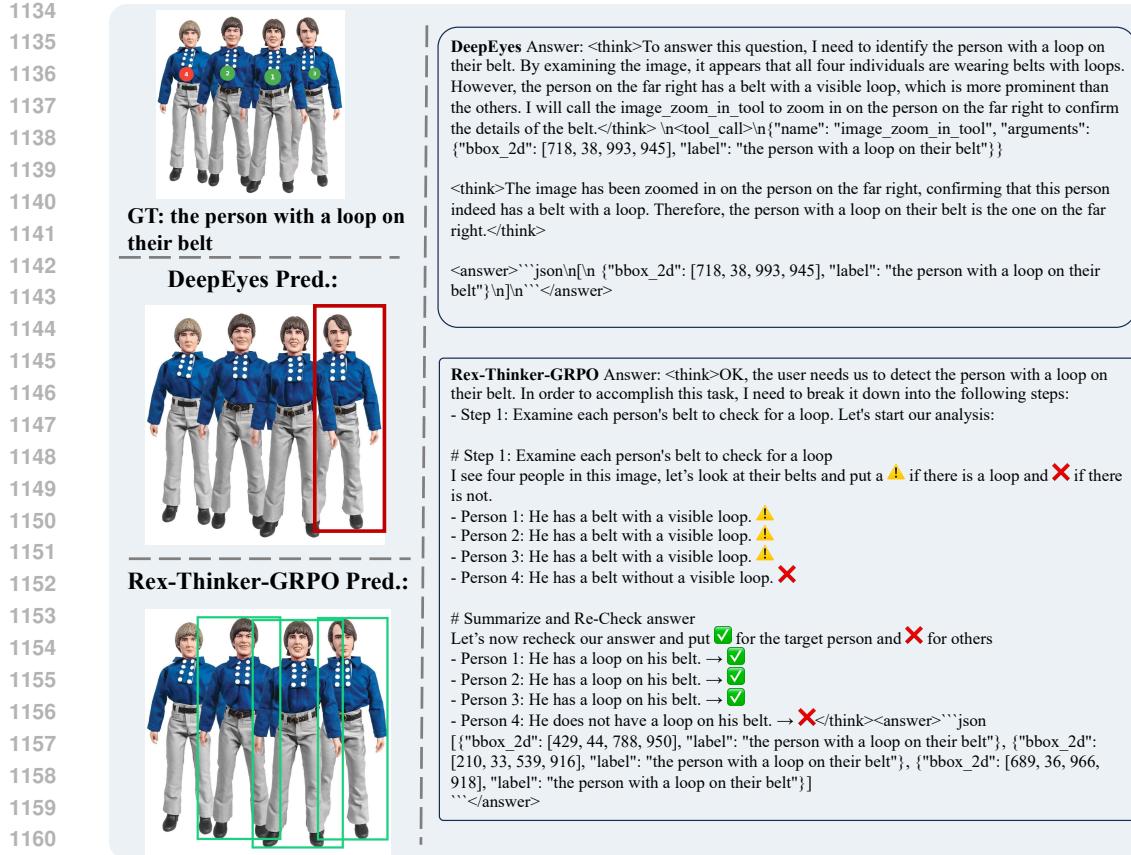


Figure 9: Qualitative comparison of Chain-of-Thought styles between DeepEyes and Rex-Thinker-GRPO. DeepEyes uses a holistic, tool-augmented approach, identifying only one of multiple targets after interactively inspecting it. Our model uses a systematic, exhaustive framework, evaluating each candidate to correctly identify all matching targets.

Method	Attribute		Position		Interaction		Reasoning		Celebrity	
	R	P	R	P	R	P	R	P	R	P
Rex-Thinker-Plain	84.17	89.89	85.48	87.18	82.16	87.80	82.63	85.34	88.58	90.58
Rex-Thinker-GRPO	88.65	91.98	87.42	88.68	81.95	88.11	87.92	87.71	89.41	91.30
	(+4.48)	(+2.09)	(+1.94)	(+1.50)	(-0.21)	(+0.31)	(+5.29)	(+2.37)	(+1.13)	(+0.72)

Table 14: Detailed performance comparison between our full model (Rex-Thinker-GRPO) and a non-CoT baseline (Rex-Thinker-Plain) across HumanRef subsets. The results highlight that the CoT framework significantly improves recall.

This improvement in recall stems directly from our core design. The structured Planning-Action-Summarization process forces the model to systematically evaluate each candidate box against the expression's sub-goals. Unlike methods that might holistically reason and identify only the most salient object, our approach's step-by-step verification minimizes the chance of overlooking a potential target. This inherently reduces false negatives (missed detections) and leads to significantly higher recall, as illustrated in the qualitative examples in Figure 10, Figure 11

This detailed analysis also allows us to transparently identify a key limitation of our current model. The "Interaction" category is the only subset where recall experiences a minor decrease (-0.21%). Upon manual inspection of failure cases (e.g., Figure 12), we hypothesize this is due to the inherent difficulty of grounding complex, multi-entity relationships in crowded scenes with significant occlusion. "Interaction" expressions (e.g., "the person talking to the woman in red") require the model to reason about a relationship between at least two individuals. In crowded scenes, these individuals may be physically close, overlapping, or occluding each other. This can lead to the initial candidate boxes being imprecise, with a single box ambiguously containing parts of multiple people. Our

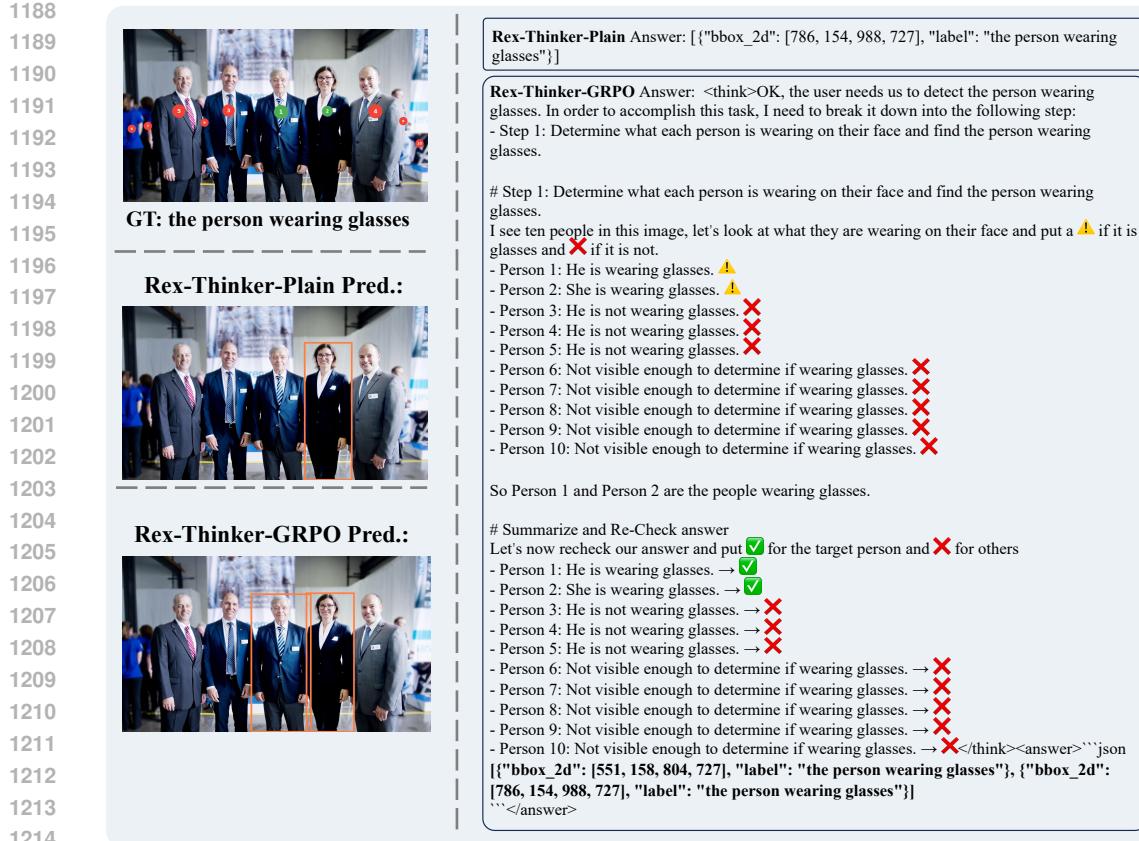


Figure 10: Comparsion between Rex-Thinker-GRPO and Rex-Thinker-Plain. Chain-of-Thought improves attribute-based referring.

current CoT framework evaluates one candidate region at a time. When presented with a heavily occluded or merged bounding box, the model struggles to definitively confirm the "interaction" based on the features within that single region. This can lead the model to conservatively (and sometimes incorrectly) reject the candidate, thus slightly lowering the recall rate for this specific, challenging scenario.

#### A.4 LIMITATIONS AND BROADER IMPACTS

##### A.4.1 INFERENCE SPEED

While the CoT-based design improves both interpretability and performance, it also introduces additional computational overhead at inference time. To quantify this, we randomly selected 100 images from the HumanRef-Benchmark test set and compared the average inference time per image between RexThinker-Plain and RexThinker-GRPO.

All experiments were conducted using the vLLM framework on a single NVIDIA A100 GPU. As shown in Table 3, RexThinker-GRPO exhibits slower inference due to its longer CoT-style outputs. This observation aligns with the general principle of test-time computation, where improved interpretability and accuracy often come at the cost of slower response time.

model	Rex-Thinker-Plain	Rex-Thinker-GRPO
average inference time	1.13s	6.68s

Table 15: Comparison of average inference time between RexThinker-Plain and RexThinker-GRPO. The CoT design in GRPO leads to slower inference.

1242

1243

1244

1245

1246

1247

1248

1249

1250

1251

1252

1253

1254

1255

1256

1257

1258

1259

1260

1261

1262

1263

1264

1265

1266

1267

1268

1269

1270

1271

1272



**Rex-Thinker-Plain** Answer: [ {"bbox\_2d": [378, 120, 803, 158]} ]

**Rex-Thinker-Plain Pred.:**



**Rex-Thinker-GRPO Pred.:**



**Rex-Thinker-GRPO** Answer<think>OK, the user needs us to detect the person in the second row from the bottom. In order to accomplish this task, I need to break it down into the following steps:

- Step 1: Identify the rows of people in the image.
- Step 2: Determine which row is the second-row from the bottom.
- Step 3: List all the people in the second-row from the bottom.

# Step 1: Identify the rows of people in the image

There are two rows of people in the image. The bottom row consists of children sitting on the floor holding papers, and the top row consists of children standing behind them also holding papers.

# Step 2: Determine which row is the second-row from the bottom

The second-row from the bottom is the row of children standing behind the sitting children.

# Step 3: List all the people in the second-row from the bottom

Let's check each person and see if they are in the second-row from the bottom:

- Person 1: He is in the bottom row (sitting), not in the second-row from the bottom → X
- Person 2: She is in the bottom row (sitting), not in the second-row from the bottom → X
- Person 3: She is in the second-row from the bottom → ✓
- Person 4: She is in the second-row from the bottom → ✓
- Person 5: She is in the second-row from the bottom → ✓
- Person 6: She is in the second-row from the bottom → ✓
- Person 7: He is in the bottom row (sitting), not in the second-row from the bottom → X
- Person 8: He is in the bottom row (sitting), not in the second-row from the bottom → X
- Person 9: He is in the bottom row (sitting), not in the second-row from the bottom → X
- Person 10: He is in the bottom row (sitting), not in the second-row from the bottom → X
- Person 11: She is in the second-row from the bottom → ✓
- Person 12: He is in the bottom row (sitting), not in the second-row from the bottom → X
- Person 13: He is in the bottom row (sitting), not in the second-row from the bottom → X
- Person 14: She is in the second-row from the bottom → ✓

# Summarize and Re-Check answer

Let's now recheck our answer and put ✓ for the target person and X for others:

...</think><answer>...</answer>

```
[ {"bbox_2d": [378, 478, 759, 540]} ]
```

Figure 11: Comparsion between Rex-Thinker-GRPO and Rex-Thinker-Plain. Chain-of-Thought improves position-based referring.

#### A.4.2 INCONSISTENT REASONING

We observe occasional inconsistencies between the model’s reasoning process and its final output. For example, as shown in Figure 13, the model identifies nine candidate objects in the reasoning phase but only includes eight in the final predicted coordinates.

We attribute such inconsistencies to the lack of explicit supervision enforcing alignment between the reasoning chain and the final answer during GRPO training. While most predictions remain consistent, these rare cases highlight a potential gap in our current framework. One possible solution is to introduce a consistency reward, which evaluates whether the number of objects summarized in the reasoning matches the number of bounding boxes in the final answer. This direction remains open for future exploration.

#### A.4.3 BROADER IMPACTS

Referring models enable more flexible and natural interaction with visual scenes compared to traditional object detection, as they can understand complex, context-dependent language. This makes them valuable in applications such as surveillance, smart cities, and smart homes.

However, these models also raise concerns. In particular, they may expose privacy risks when deployed in sensitive environments and can inherit biases from training data or user input. Although CoT reasoning improves interpretability, ensuring its consistency and robustness remains an open challenge. Future work should address these risks to ensure safe and responsible deployment.

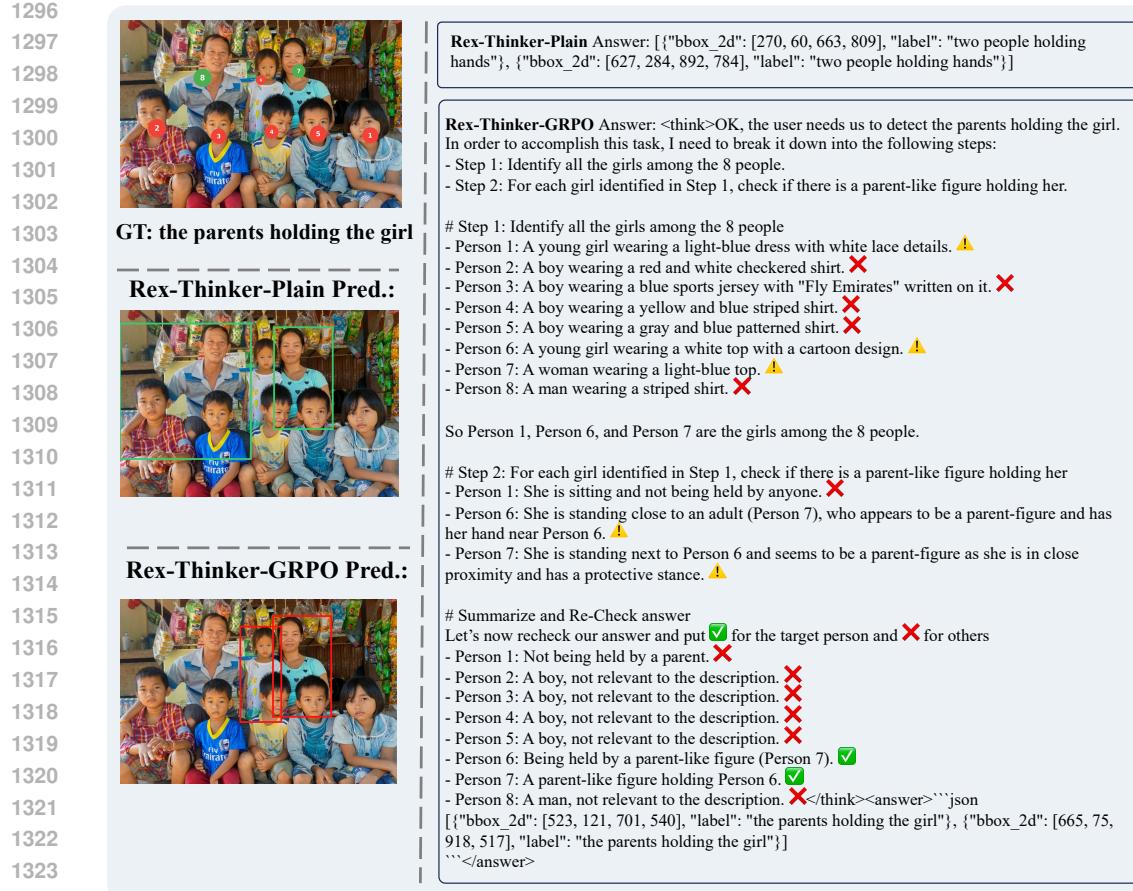


Figure 12: A failure case of Rex-Thinker-GRPO, where CoT leads to a wrong answer.

## A.5 VISUALIZATION RESULTS

In this section, we present qualitative results of the RexThinker model across different scenarios, as shown in Figures 14–33. All visualizations are generated by the RexThinker-GRPO model, which is trained solely on the HumanRef-CoT dataset.



1404

1405

1406

1407

1408

1409

1410

1411

1412

1413

1414

1415

1416

1417

1418

1419

1420

1421

1422

1423

1424

1425

1426

1427

1428

1429

1430

1431

1432

1433

1434

1435

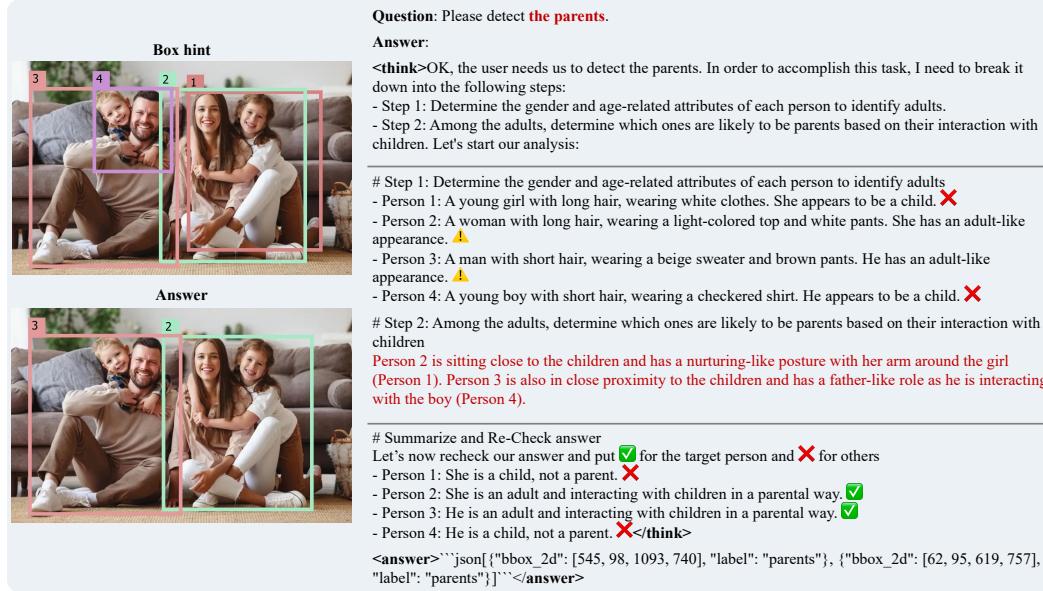


Figure 15: Attribute referring example.

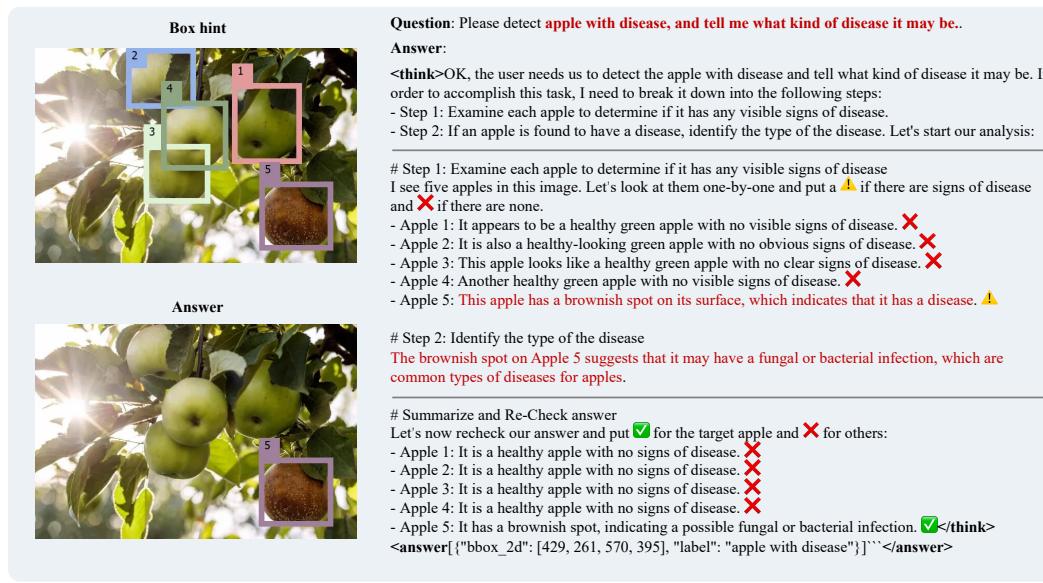


Figure 16: Attribute referring example.

1454

1455

1456

1457

1458

1459

1460

1461

1462

1463

1464

1465

1466

1467

1468

1469

1470

1471

1472

1473

1474

1475

1476

1477

1478

1479

1480

1481

1482

1483

1484

1485

1486

1487

1488

1489

1490

1491

1492

1493

1494

1495

1496

1497

1498

1499

1500

1501

1502

1503

1504

1505

1506

1507

1508

1509

1510

1511

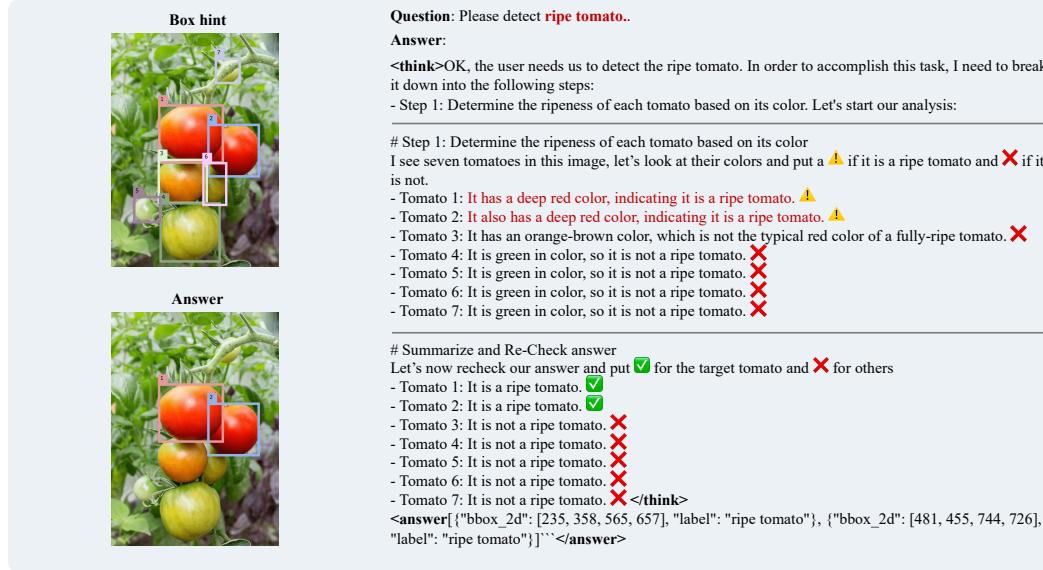


Figure 17: Attribute referring example.

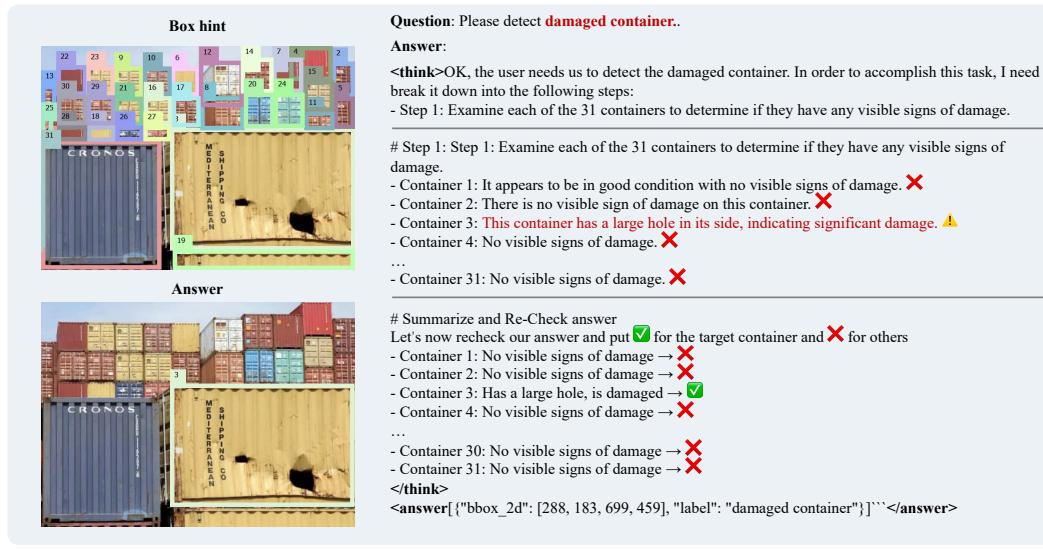


Figure 18: Attribute referring example.

1512

1513

1514

1515

1516

1517

1518

1519

1520

1521

1522

1523

1524

1525

1526

1527

1528

1529

1530

1531

1532

1533

1534

1535

1536

1537

1538

1539

1540

1541

1542

1543

1544

1545

1546

1547

1548

1549

1550

1551

1552

1553

1554

1555

1556

1557

1558

1559

1560

1561

1562

1563

1564

1565

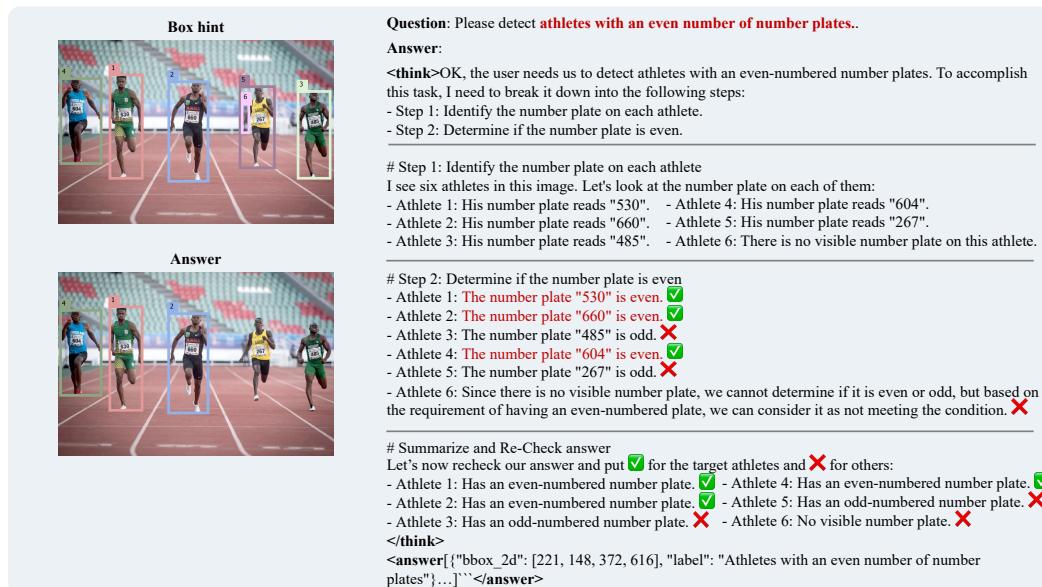


Figure 19: Attribute referring example.

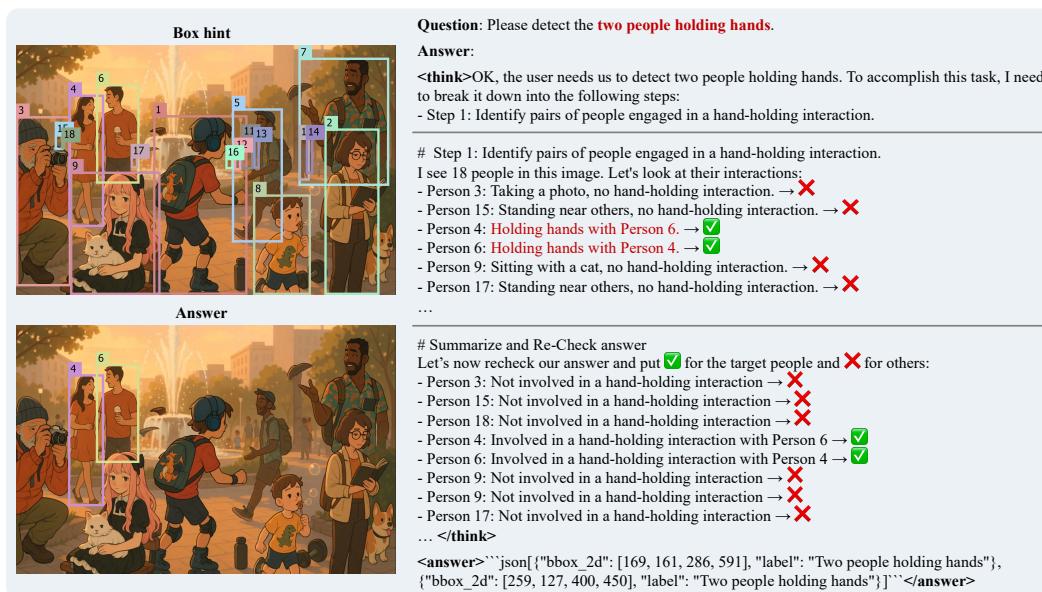
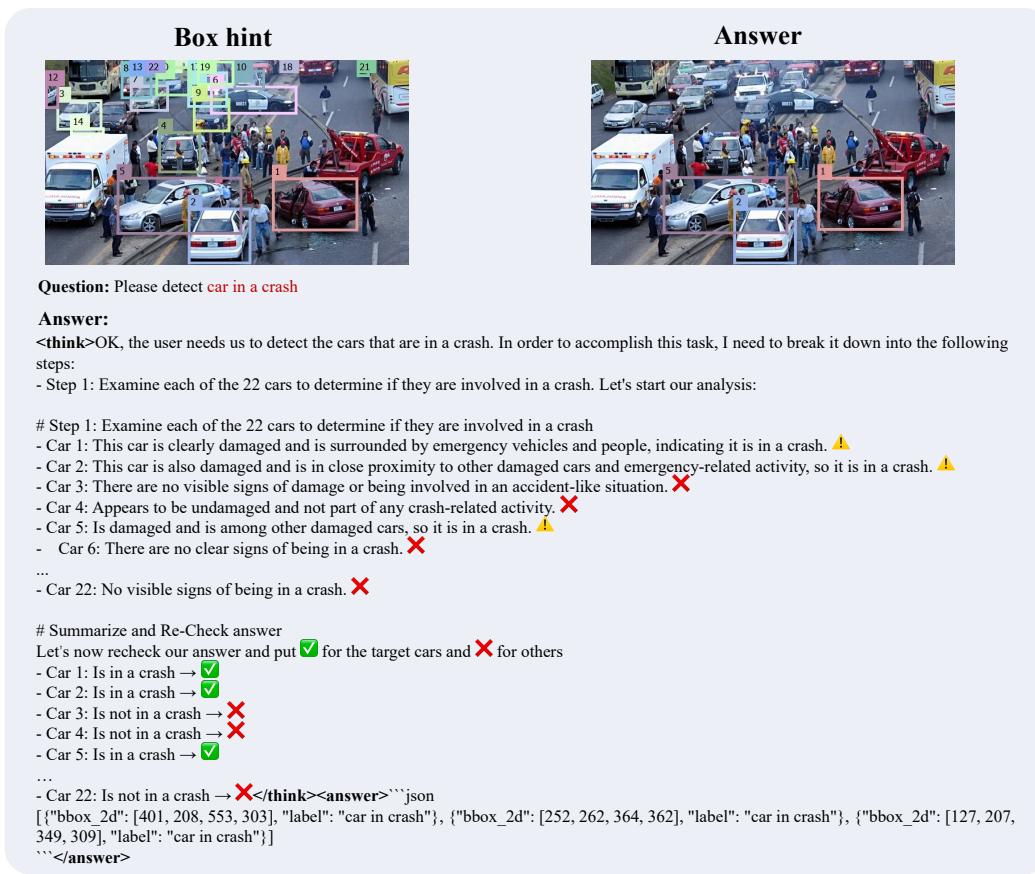
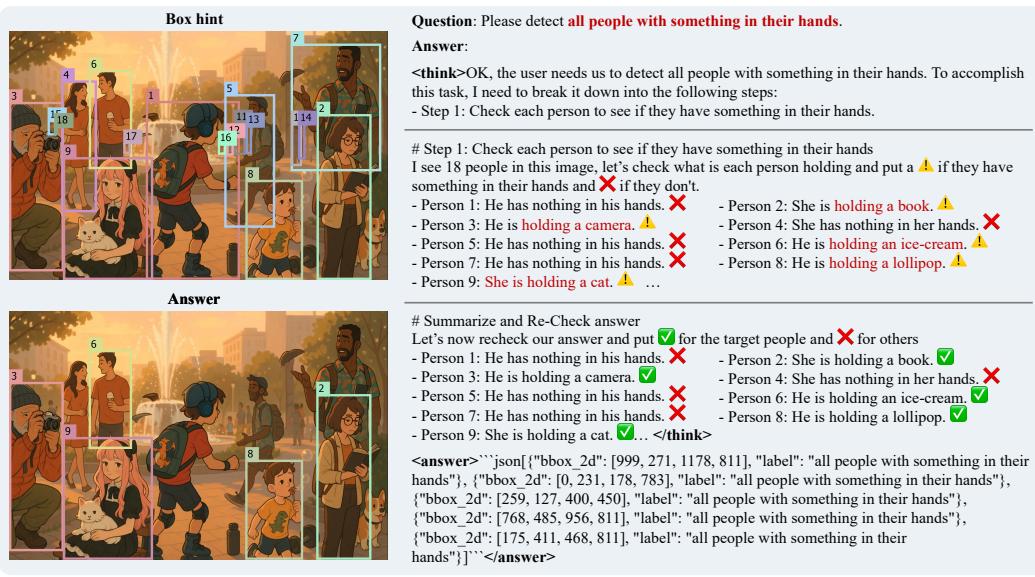


Figure 20: Interaction referring example.

1566  
1567  
1568  
1569  
1570  
1571  
1572  
1573  
1574  
1575



1595  
1596  
1597  
1598  
1599



1610  
1611  
1612  
1613  
1614  
1615  
1616  
1617

Figure 21: Interaction referring example.

1620

1621

1622

1623

1624

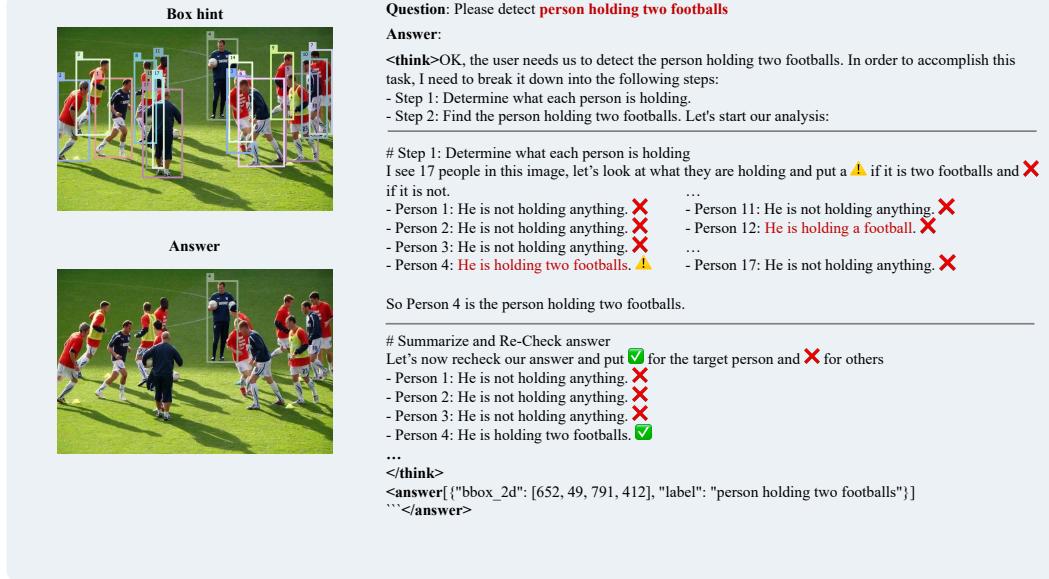


Figure 23: Interaction referring example.

1643

1644

1645

1646

1647

1648

1649

1650

1651

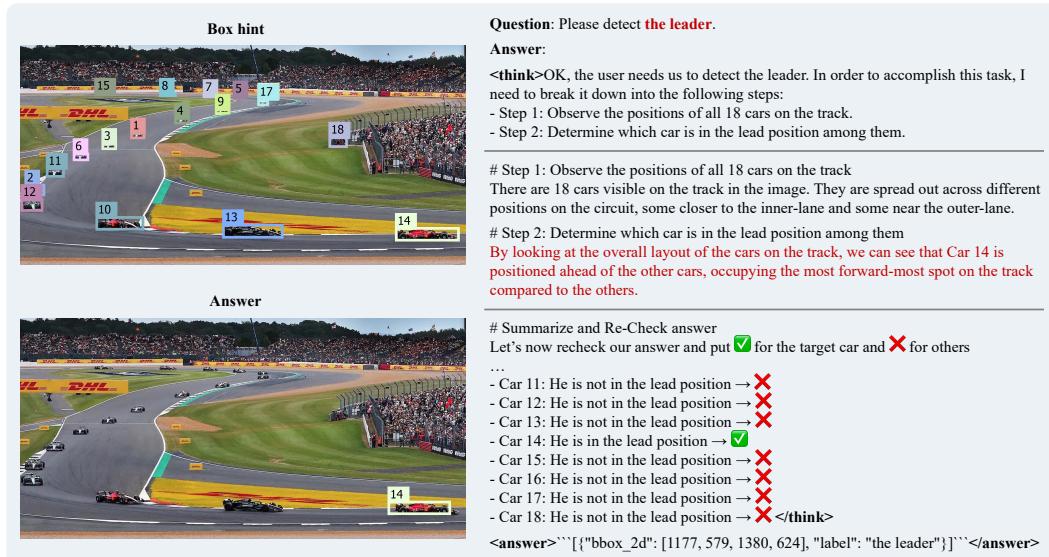


Figure 24: Position referring example.

1660

1661

1662

1663

1664

1665

1666

1667

1668

1669

1670

1671

1672

1673

1674  
 1675  
 1676  
 1677  
 1678  
 1679  
 1680  
 1681  
 1682  
 1683  
 1684  
 1685  
 1686  
 1687  
 1688  
 1689  
 1690  
 1691  
 1692  
 1693  
 1694  
 1695  
 1696  
 1697  
 1698  
 1699  
 1700  
 1701  
 1702  
 1703  
 1704  
 1705  
 1706  
 1707  
 1708  
 1709  
 1710  
 1711  
 1712  
 1713  
 1714  
 1715  
 1716  
 1717  
 1718  
 1719  
 1720  
 1721  
 1722  
 1723  
 1724  
 1725  
 1726  
 1727

**Box hint**



**Answer**



**Question:** Please detect **all person to the right of the person wearing a yellow tie**

**Answer:**

<think> OK, the user needs us to detect all person to the right of the person wearing a yellow tie. In order to accomplish this task, I need to break it down into the following steps:

- Step 1: Find the person wearing a yellow tie.
- Step 2: Identify all the people to the right of the person found in Step 1

---

# Step 1: Find the person wearing a yellow tie  
 I see seven people in this image. Let's check their ties and put a ▲ if it is a yellow tie and ✗ if it is not.

- Person 1: She is wearing a red-colored uniform and no visible tie. ✗
- Person 2: He is wearing a red-colored uniform and no visible tie. ✗
- Person 3: He is wearing a dark suit with a red tie. ✗
- Person 4: He is wearing a dark suit with a red tie. ✗
- Person 5: **He is wearing a dark suit with a yellow tie.** ▲
- Person 6: He is wearing a dark suit with a green tie. ✗
- Person 7: She is wearing a red-colored uniform and no visible tie. ✗

So Person 5 is the person wearing a yellow tie.

# Step 2: Identify all the people to the right of Person 5  
 The order from left to right is [Person 7, Person 5, Person 3, Person 4, Person 6, Person 2, Person 1].  
 People to the right of Person 5 are Person 3, Person 4, Person 6, Person 2, Person 1.

---

# Summarize and Re-Check answer  
 Let's now recheck our answer and put ✓ for the target people and ✗ for others

- Person 1: She is to the right of the person wearing a yellow tie. ✓
- ...
- Person 5: He is the person wearing a yellow tie (reference person). ▲
- Person 6: He is to the right of the person wearing a yellow tie. ✓
- Person 7: She is to the left of the person wearing a yellow tie. ✗

</think>

<answer> {"bbox\_2d": [838, 323, 1028, 783], "label": "all person to the right of the person wearing a yellow tie"}</answer>

Figure 25: Position referring example.

1709  
 1710  
 1711  
 1712  
 1713  
 1714  
 1715  
 1716  
 1717  
 1718  
 1719  
 1720  
 1721  
 1722  
 1723  
 1724  
 1725  
 1726  
 1727

**Box hint**



**Answer**



**Question:** Please detect **Hot dogs on the grill**

**Answer:**

<think> OK, the user needs us to detect the hot dogs on the grill. In order to accomplish this task, I need to break it down into the following steps:

- Step 1: Determine the position of each hot-dog in relation to the grill

---

# Step 1: Determine the position of each hot-dog in relation to the grill  
 Let's analyze the position of each hot-dog:

- Hot dog 1: It is placed on a piece of paper on the table, not on the grill. ✗
- Hot dog 2: It is also on a piece of paper on the table, not on the grill. ✗
- Hot dog 3: It is placed on the metal grill. ▲
- Hot dog 4: It is placed on the metal grill. ▲

---

# Summarize and Re-Check answer  
 Let's now recheck our answer and put ✓ for the target hot-dogs and ✗ for others:

- Hot dog 1: Not on the grill. ✗
- Hot dog 2: Not on the grill. ✗
- Hot dog 3: On the grill. ✓
- Hot dog 4: On the grill. ✓

</think>

<answer> {"bbox\_2d": [782, 118, 1028, 451], "label": "Hot dogs on the grill"}, {"bbox\_2d": [901, 281, 1028, 550], "label": "Hot dogs on the grill"}</answer>

Figure 26: Position referring example.

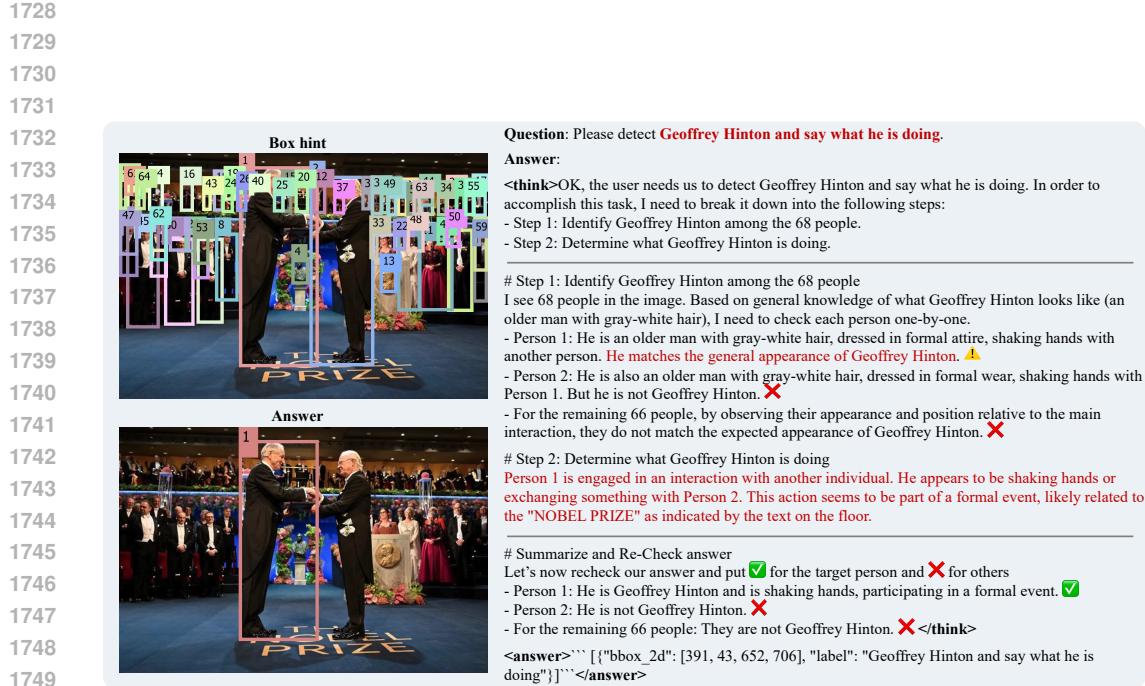


Figure 27: Celebrity referring example.

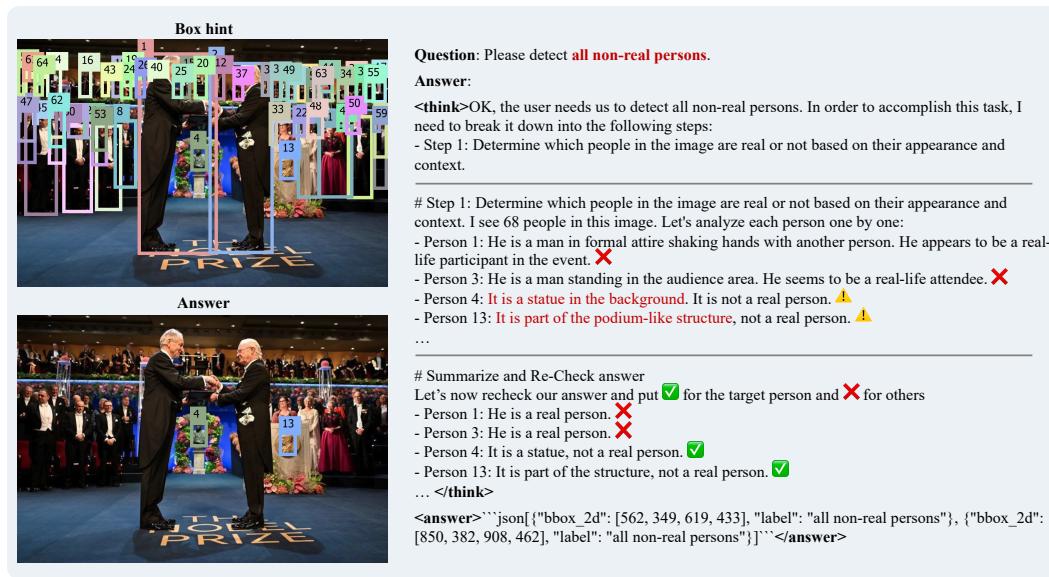


Figure 28: Reasoning referring example.

1782

1783

1784

1785

1786

1787

1788

1789

1790

1791

1792

1793

1794

1795

1796

1797

1798

1799

1800

1801

1802

1803

1804

1805

1806

1807

1808

1809

1810

1811

1812

1813

1814

1815

1816

1817

1818

1819

1820

1821

1822

1823

1824

1825

1826

1827

1828

1829

1830

1831

1832

1833

1834

1835



Figure 29: Reasoning referring example.

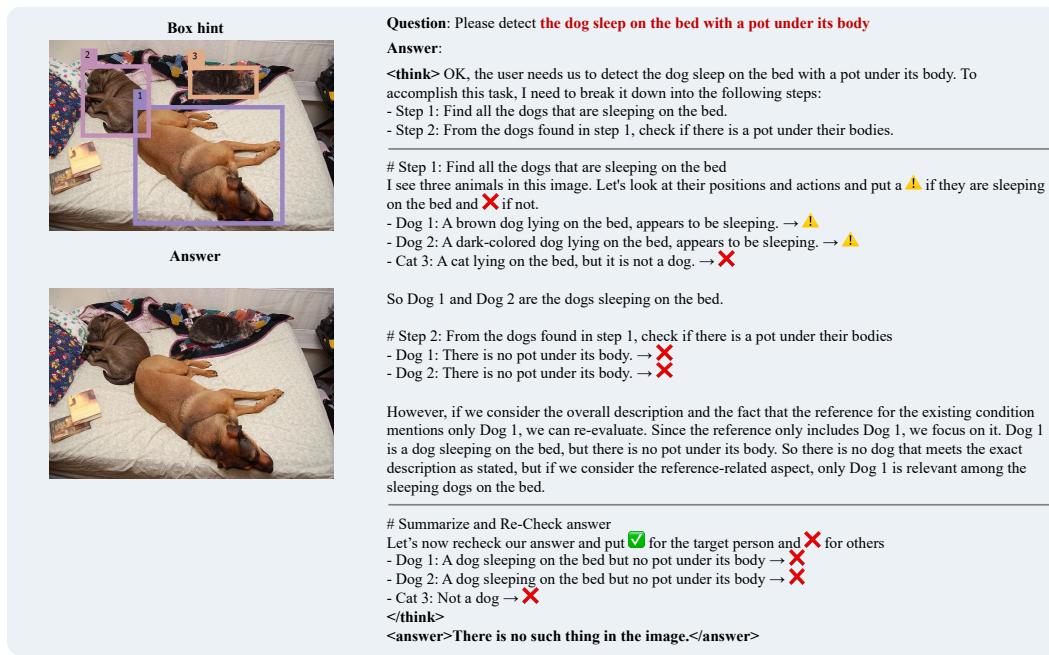


Figure 30: Rejection referring example.

1836  
1837  
1838  
1839  
1840  
1841  
1842  
1843  
1844  
1845  
1846  
1847  
1848  
1849  
1850  
1851  
1852  
1853  
1854  
1855  
1856  
1857  
1858  
1859  
1860

**Box hint**



**Question:** Please detect **person wearing cloth that has letter A**

**Answer:**

<think> OK, the user needs us to detect the person wearing cloth that has letter A. To accomplish this task, I need to break it down into the following steps:

- Step 1: Find all the people in the image.
- Step 2: Check the clothes of each person to see if there is the letter A on them.

---

# Step 1: Find all the people in the image.  
There are 8 people in the image, listed from left-to-right as person 4, person 6, person 5, person 2, person 3, person 7, person 1, person 8.

# Step 2: Check the clothes of each person to see if there is the letter A on them.

- Person 4: Wearing a pink dress with the word "VOL" on it, no letter A → X
- Person 6: Wearing an orange dress with the word "VOL" on it, no letter A → X
- Person 5: Wearing a blue shirt with the word "L" on it, no letter A → X
- Person 2: Wearing a dress with the word "U" on it, no letter A → X
- Person 3: Wearing a yellow shirt with the word "N" on it, no letter A → X
- Person 7: Wearing a brown shirt with the word "T" on it, no letter A → X
- Person 1: Wearing a blue shirt with the word "ER" on it, no letter A → X
- Person 8: There is a blue shirt inside the box, but no person 8 is visible in terms of wearing clothes with a letter A, and among the people visible, none have a letter A on their clothes.

We also note that person 4 and person 6 are part of the group mentioned in the additional information, but they still do not have the letter A on their clothes.

---

# Summarize and Re-Check answer  
Let's now recheck our answer and put ✓ for the target person and X for others

- Person 4: no letter A on clothes → X
- Person 6: no letter A on clothes → X
- Person 5: no letter A on clothes → X
- Person 2: no letter A on clothes → X
- Person 3: no letter A on clothes → X
- Person 7: no letter A on clothes → X
- Person 1: no letter A on clothes → X
- Person 8: no person 8 visible wearing relevant clothes and others don't have A either → X

<answer>There is no such thing in the image.</answer>

Figure 31: Rejection referring example.

1861  
1862  
1863  
1864  
1865  
1866  
1867  
1868  
1869  
1870  
1871  
1872  
1873  
1874  
1875  
1876  
1877  
1878  
1879  
1880  
1881  
1882  
1883  
1884  
1885  
1886  
1887  
1888  
1889

**Box hint**



**Question:** Please detect **Kobe Bryant**.

**Answer:**

<think>OK, the user needs us to detect Kobe Bryant. To accomplish this task, I need to break it down into the following steps:

- Step 1: Identify each person based on their appearance and basketball-related attributes.
- Step 2: Check if any of the identified people match the identity of Kobe Bryant.

---

# Step 1: Identify each person based on their appearance and basketball-related attributes.

- Person 1: A man wearing a white Dallas-branded basketball jersey, holding a basketball. He has short hair and a beard.
- Person 2: A man wearing a green Boston-branded basketball jersey with the number 0. He is in a shooting pose.
- Person 3: A man wearing a white Denver Nuggets-branded basketball jersey with the number 15. He is in a shooting pose.
- Person 4: A man wearing a black San Antonio Spurs-branded basketball jersey. He has a muscular build.
- Person 5: A man wearing a blue and yellow basketball jersey. He has a beard and is smiling.
- Person 6: Not clearly visible in terms of distinct features related to identification.
- Person 7: Partially visible, no clear distinct features for identification.
- Person 8: A man wearing a purple and yellow hoodie, not in a basketball-playing pose.
- Person 9: A man wearing a blue Thunder-branded basketball jersey with the number 2. He has a headband.

# Step 2: Check if any of the identified people match the identity of Kobe Bryant.  
Kobe Bryant is a well-known basketball player, but **none of the people in the image have the exact appearance and context that can confirm them as Kobe Bryant...**

---

# Summarize and Re-Check answer  
Let's now recheck our answer and put ✓ for the target person and X for others

- Person 1-9: Not Kobe Bryant → X

<answer>There is no such thing in the image.</answer>

Figure 32: Rejection referring example.

1890  
1891  
1892  
1893  
1894  
1895  
1896  
1897  
1898  
1899  
1900  
1901  
1902      **Question:** Please detect **all fish**, analyze their subcategories, and tell me if they are in a **normal or endangered state**.  
1903  
1904      **Answer:**  
1905      <think>OK, the user needs us to detect all fish, analyze their subcategories, and tell if they are in a  
1906      normal or endangered state. To accomplish this task, I need to break it down into the following steps:  
1907      - Step 1: Identify all the fish in the image.  
1908      - Step 2: Analyze the subcategories of the identified fish.  
1909      - Step 3: Determine the state (normal or endangered) of each fish.  
1910  
1911      # Step 1: Identify all the fish in the image  
1912      There are six sea-related characters in the image. Let's check which ones are fish:  
1913      - Fish 1: A **clown fish** with orange and white stripes. ⚠  
1914      - Fish 2: A **puffer fish** with a round body and spiky fins. ⚠  
1915      - Fish 3: A **stingray** with a flat, wing-like body. ⚠  
1916      - Fish 4: A **blue whale** with a long body and flippers. ⚠  
1917      - Fish 5: A **jellyfish** with a pink body and blue tentacles. ✗  
1918      - Fish 6: An **oyster** with a purple shell and an eye-like feature. ✗  
1919      So, Fish 1, Fish 2, Fish 3, and Fish 4 are fish.  
1920  
1921      # Step 2: Analyze the subcategories of the identified fish  
1922      - Fish 1: A clown fish, which is a type of fish.  
1923      - Fish 2: A puffer fish, which is a type of fish.  
1924      - Fish 3: A stingray, which is a type of fish.  
1925      - Fish 4: A blue whale, which is a type of fish.  
1926  
1927      # Step 3: Determine the state (normal or endangered) of each fish  
1928      - Fish 1: **The clown fish is a common and widely-distributed fish, so it is in a normal state.** ✓  
1929      - Fish 2: **The puffer fish is also a relatively common fish, so it is in a normal state.** ✓  
1930      - Fish 3: **The stingray is not as common as some other fish, but still not on the endangered list, so it is in a normal state.** ✓  
1931      - Fish 4: **The blue whale is an endangered species due to hunting and habitat loss.** ✗  
1932  
1933      # Summarize and Re-Check answer  
1934      Let's now recheck our answer and put ✓ for the target fish and ✗ for others:  
1935      - Fish 1: In a normal state. ✓  
1936      - Fish 2: In a normal state. ✓  
1937      - Fish 3: In a normal state. ✓  
1938      - Fish 4: Endangered. ✗  
1939      - Fish 5: Not a fish. ✗  
1940      - Fish 6: Not a fish. ✗

</think>

<answer>``` [{"bbox\_2d": [515, 115, 800, 255], "label": "all fish, analyze their subcategories, and tell me if they are in a normal or endangered state"}, {"bbox\_2d": [512, 457, 812, 687], "label": "all fish, analyze their subcategories, and tell me if they are in a normal or endangered state"}, {"bbox\_2d": [90, 855, 812, 890], "label": "all fish, analyze their subcategories, and tell me if they are in a normal or endangered state"}]```</answer>

Figure 33: Reasoning referring example with multi-task chat.

```

1944
1945
1946
1947
1948 ....
1949 Example 1 :
1950 METAINFO:
1951 {
1952     "total_num_of_person": 2,
1953     "answer": "Person 1",
1954     "order_from_left_to_right": [Person 1, Person 2]
1955 }
1956 Question: the person wearing a watch
1957 The output should be:
1958 (analysis)
1959 OK, the user needs us to detect the person wearing a watch. To accomplish this task, I need to break it down into the following steps:
1960 - Step 1: Determine what each person is wearing on their wrists to check for a watch.
1961 # Step 1: Determine what each person is wearing on their wrists to check for a watch
1962 - Person 1: She is wearing a watch on her left wrist. ⚠
1963 - Person 2: There is no visible watch on her wrists. ✗
1964 # Summarize and Re - Check answer
1965 Let's now recheck our answer and put ✓ for the target person and ✗ for others
1966 - Person 1: She is wearing a watch → ✓
1967 - Person 2: She is not wearing a watch → ✗
1968 (analysis)
1969 To summarize, Person 1 is the person wearing a watch
1970 (answer)
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997

```

Figure 34: In-context prompt for *attribute* subset in HumanRef-CoT.


```

1976 (analysis)
1977 OK, the user needs us to detect the person with brownish - black curly hair. In order to
1978 accomplish this task, I need to break it down into the following steps:
1979 - Step 1: Examine the hair color and texture of each person. Let's start our analysis:
1980 # Step 1: Examine the hair color and texture of each person
1981 I see 8 people in this image, let's look at their hair and put a ⚠ if it is brownish - black curly hair
1982 and ✗ if it is not.
1983 - Person 1: She has brownish - black curly hair. ⚠
1984 - Person 2: She has dark hair that appears to be straight. ✗
1985 - Person 3: He has short, light - colored hair. ✗
1986 - Person 4: She has brownish - black curly hair. ⚠
1987 - Person 5: He has short, dark hair. ✗
1988 - Person 6: He has short, dark hair. ✗
1989 - Person 7: He has short, dark hair. ✗
1990 - Person 8: She has brownish - black curly hair. ⚠
1991 # Summarize and Re - Check answer
1992 Let's now recheck our answer and put ✓ for the target person and ✗ for others
1993 - Person 1: She has brownish - black curly hair. ✓
1994 - Person 2: She has straight dark hair. ✗
1995 - Person 3: He has short, light - colored hair. ✗
1996 - Person 4: She has brownish - black curly hair. ✓
1997 - Person 5: He has short, dark hair. ✗
1998 - Person 6: He has short, dark hair. ✗
1999 - Person 7: He has short, dark hair. ✗
2000 - Person 8: She has brownish - black curly hair. ✓
2001 (analysis)
2002 (answer)
2003 To summarize, Person 1, Person 4 and Person 8 are the people with brownish - black curly hair
2004 (answer)

```

Figure 35: Visualization of GPT-4o's output on the *attribute* subset.

1998  
 1999  
 2000  
 2001  
 2002 Example 1 :  
 ````  
 METAINFO:  
 {  
 "total\_num\_of\_person": 4,  
 "answer": "person 3",  
 "order\_from\_left\_to\_right": [person 1, person 2, person 3, person 4]  
 }  
 2003  
 2004  
 2005  
 2006  
 2007 Question: The third person from the left to right  
 2008 The output should be:  
 (analysis)  
 2009 OK, the user needs us to detect the third person from the left to right. To accomplish this task, I need to break it down into the following steps:  
 - Step 1: Sort the people from left to right and find the third person from the left to right.  
 2010  
 # Step 1: Sort the people from left to right and find the third person from the left to right.  
 I need to sort the people from left to right. I see 4 people in this image, and their order from left to right is [Person 1, Person 2, Person 3, Person 4].  
 So Person 3 is the third person from the left to right.  
 2011  
 2012  
 2013  
 2014 # Summarize and Re-Check answer  
 Let's now recheck our answer and put  for the target person and  for others  
 2015 - Person 1: He is the first person from left to right →   
 - Person 2: He is the second person from left to right →   
 2016 - Person 3: He is the third person from left to right →   
 - Person 4: He is the forth person from left to right →   
 2017 (analysis)  
 2018 (answer)  
 2019 To summarize, person 3 is the third person from left to right  
 2020 ````

Figure 36: In-context prompt for *position (inner)* subset in HumanRef-CoT.

2021  
 2022  
 2023  
 2024  
 2025  
 2026  
 2027  
 2028  
 2029

2030  
 2031  
 2032  
 2033  
 2034  
 2035  
 2036  
 2037  
 2038  
 2039  
 2040  
 2041 Referring Phrase: the second person wearing gloves when  
 counting from right to left  
 2042  
 2043  
 2044  
 2045  
 2046  
 2047  
 2048  
 2049  
 2050  
 2051

````

(analysis)  
 OK, the user needs us to detect the second person wearing gloves when counting from right to left. To accomplish this task, I need to break it down into the following steps:  
 - Step 1: Determine what each person is wearing and find the people wearing gloves.  
 - Step 2: Sort the results of the first step by position from right to left.  
 # Step 1: find all the people wearing gloves  
 I see four people in this image, let's look at their hands and put a  if they are wearing gloves and  if they are not.  
 - Person 1: He is wearing black gloves.   
 - Person 2: She is wearing black gloves.   
 - Person 3: He is wearing black gloves.   
 - Person 4: He is wearing black gloves.   
 So Person 1, Person 2, Person 3, and Person 4 are all wearing gloves.  
 # Step 2: sort all the people wearing gloves from right to left  
 Now I need to sort their order from right to left. The order from right to left is [Person 1, Person 2, Person 3, Person 4].  
 So Person 2 is the second person wearing gloves when counting from right to left.  
 # Summarize and Re - Check answer  
 Let's now recheck our answer and put  for the target person and  for others.  
 - Person 1: He is the first person wearing gloves when counting from right to left.   
 - Person 2: She is the second person wearing gloves when counting from right to left.   
 - Person 3: He is the third person wearing gloves when counting from right to left.   
 - Person 4: He is the fourth person wearing gloves when counting from right to left.   
 (analysis)  
 (answer)  
 To summarize, Person 2 is the second person wearing gloves when counting from right to left  
 (answer)

Figure 37: Visualization of GPT-4o's output on the *position (inner)* subset.

2052

2053

2054

2055

2056

2057

2058

2059

2060

2061

2062

2063

2064

2065

2066

2067

2068

2069

2070

2071

2072

2073

2074

2075

2076

2077

2078



Referring Phrase: all the people to the right of the word 'muckle'

(analysis)

OK, the user needs us to detect all the people to the right of the word 'muckle'. To accomplish this task, I need to break it down into the following steps:

- Step 1: Locate the word 'muckle' in the image.
- Step 2: Identify all the people positioned to the right of the located word.

# Step 1: Locate the word 'muckle' in the image

The word 'muckle' is prominently displayed on a white counter - like structure in the center of the image.

- Word 'muckle': On a white counter - like structure in the center of the image ✓

# Step 2: Identify all the people positioned to the right of the located word

Let's analyze the position of each person relative to the word 'muckle':

- Person 1: He is sitting on the counter with the word 'muckle' and is not to the right of it → ✗
- Person 2: He is standing to the right of the word 'muckle' → ✓
- Person 3: He is standing to the right of the word 'muckle' → ✓
- Person 4: He is standing to the right of the word 'muckle' → ✓
- Person 5: He is standing to the left of the word 'muckle' → ✗

# Summarize and Re - Check answer

Let's now recheck our answer and put ✓ for the target people and ✗ for others:

- Person 1: Not to the right of the word 'muckle' → ✗
- Person 2: To the right of the word 'muckle' → ✓
- Person 3: To the right of the word 'muckle' → ✓
- Person 4: To the right of the word 'muckle' → ✓
- Person 5: Not to the right of the word 'muckle' → ✗

(analysis)

(answer)

To summarize, Person 2, Person 3, and Person 4 are all the people to the right of the word 'muckle'

(answer)

Figure 38: In-context prompt for *position (outer)* subset in HumanRef-CoT.

Example 1 :

...

METAINFO:

```
{
  "total_num_of_person": 5,
  "answer": "person 1",
  "order_from_left_to_right": [person 1, person 4, person 2, person 5, person 3]
}
```

Question: the person to the left of the green light bulb

The output should be:

(analysis)

OK, the user needs us to detect the person to the left of the green light-bulb. To accomplish this task, I need to break it down into the following steps:

Step 1: Find the green light-bulb in this image.

Step 2: Fine the person to the left of the green light-bulb

# Step 1: Find the green light-bulb in this image.

Clearly, I can see a green light - bulb near the lower - left part of the image.

- green light - bulb: a green light - bulb near the lower - left part of the image ✓

# Step2: Fine the person to the left of the green light-bulb

Let's analyze the position of each person next:

- Person 1: He is standing directly to the left of the green light-bulb → ✓
- Person 2: She is on a ladder near the top-right part of the image, not to the left of the green light - bulb → ✗
- Person 3: He is standing near the right side of the image holding a tablet, not to the left of the green light - bulb → ✗

# Summarize and Re-Check answer

Let's now recheck our answer and put ✓ for the target person and ✗ for others

- Person 1: He is standing directly to the left of the green light-bulb → ✓
- Person 2: She is on a ladder not the left of the green light-bulb → ✗
- Person 3: He is not on the left of the green light-bulb → ✗

(analysis)

(answer)

To summarize, person 1 is to the left of the green light bulb

(answer)

...

Figure 39: Visualization of GPT-4o's output on the *position (outer)* subset.

```

2106
2107
2108
2109 Example 1 :
2110 ...
2111 METAINFO:
2112 {
2113     "total_num_of_person": 3,
2114     "answer": "person 1",
2115     "order_from_left_to_right": [person 1, person 2, person 3, person 4]
2116 }
2117
2118 Question: The man who is hugging the woman from behind
2119
2120 The output should be:
2121 (analysis)
2122 OK, the user needs us to detect the man who is hugging the woman from behind
2123 . To accomplish this task, I need to break it down into the following steps:
2124 - Step 1: Find all the man in this image.
2125 - Step 2: Look at each man's action and find the man who is hugging the woman from behind
2126
2127 # Step 1: Find all the man in this image.
2128 I see four person in this image, let's look at their gender and put a ⚠ if they are man and ✗ if it is not.
2129 - Person 1: It is a man sitting on the fence → ⚠
2130 - Person 2: It is a man wearing a blue shirt → ⚠
2131 - Person 3: It is a woman in red dress → ✗
2132 - Person 4: It is a woman standing on the grass → ✗
2133
2134 # Step 2: Look at each man's interaction from step 1
2135 Person 1: He is man standing behind a woman (Person 3), with his arms around her, fulfilling the description of hugging the woman from behind → ✓
2136 Person 2: A man sitting on the fence, holding hands with person 4. He is not hugging a woman from behind. → ✗
2137 So Person 1 is the man hugging the woman (Person 3) from behind
2138
2139 # Summarize and Re-Check answer
2140 Let's now recheck our answer and put ✓ for the target person and ✗ for others
2141 - Person 1: man hugging the woman from behind → ✓
2142 - Person 2: man not hugging from behind → ✗
2143 - Person 3: a woman, not a man hugging from behind → ✗
2144 - Person 4: a woman, not a man hugging from behind → ✗
2145
2146 (analysis)
2147 To summarize, person 1 is the man who is hugging the woman from behind
2148 (answer)
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159

```

Figure 40: In-context prompt for *interaction (inner)* subset in HumanRef-CoT.


2142 (analysis)  
2143 OK, the user needs us to detect two people who are holding hands. To accomplish  
2144 this task, I need to break it down into the following steps:  
2145 - Step 1: Identify the pair of people engaged in a hand - holding interaction.  
2146 # Step 1: Identify the pair of people engaged in a hand - holding interaction.  
2147 I see four people in this image. Let's look at their interactions:  
2148 - Person 1: He is walking with his arm around person 4, not holding hands. → ✗  
2149 - Person 2: He is holding hands with person 3. → ✓  
2150 - Person 3: She is holding hands with person 2. → ✓  
2151 - Person 4: She is being walked with an arm around by person 1, not holding hands.  
2152 → ✗  
2153 # Summarize and Re - Check answer  
2154 Let's now recheck our answer and put ✓ for the target people and ✗ for others:  
2155 - Person 1: Not involved in a hand - holding interaction → ✗  
2156 - Person 2: Involved in a hand - holding interaction with Person 3 → ✓  
2157 - Person 3: Involved in a hand - holding interaction with Person 2 → ✓  
2158 - Person 4: Not involved in a hand - holding interaction → ✗  
2159 (analysis)  
2160 (answer)  
2161 To summarize, person 2 and person 3 are the two people who are holding hands.  
2162 (answer)

Figure 41: Visualization of GPT-4o's output on the *interaction (inner)* subset.

2160  
 2161  
 2162  
 2163  
 2164 Example 1 :  
 2165 ...  
 2166 METAINFO:  
 2167 {  
 2168    "total\_num\_of\_person": 3,  
 2169    "answer": "person 1",  
 2170    "order\_from\_left\_to\_right": [person 2, person 3, person 1]  
 2171 }  
 2172 Question: The woman holding a red sniper rifle in her hand  
 2173  
 2174 The output should be:  
 2175 (analysis)  
 2176 OK, the user needs us to find the woman holding a red sniper-rifle in her hand. To accomplish this task, I need to break it down into the following steps:  
 2177 - Step 1: See if everyone is holding objects in their hands and what they're holding  
 2178  
 2179 # Step1: See if everyone is holding objects in their hands and what they're holding  
 2180 There are 5 people in the picture, and we need to check the objects held by each of them:  
 2181 - Person 1: A woman with dark hair and a red - themed outfit. She is holding a long, red firearm which appears to be a sniper - rifle. So she matches the description → ✓  
 2182 - Person 2: A woman with short orange hair, wearing a pink and white outfit. She is holding a gray and pink - colored weapon that does not resemble a red sniper - rifle → ✗  
 2183 - Person 3: A woman with long orange hair and a green and white dress. She is holding a weapon, but it is not a red sniper - rifle → ✗  
 2184  
 2185 # Summarize and Re-Check answer  
 2186 Let's now recheck our answer and put ✓ for the target person and ✗ for others  
 2187 Person 1: holding a red sniper - rifle → ✓  
 2188 Person 2: holding a non - red weapon → ✗  
 2189 Person 3: holding a non - red weapon → ✗  
 2190 (analysis)  
 2191  
 2192 (answer)  
 2193 To summarize, person 1 is the woman holding a red sniper rifle in her hand.  
 2194 (answer)

Figure 42: In-context prompt for *interaction (outer)* subset in HumanRef-CoT.

2194  
 2195  
 2196  
 2197  
 2198  
 2199  
 2200  
 2201  
 2202  
 2203  
 2204  
 2205 Referring Phrase: the person holding a microphone in the hand  
 2206  
 2207  
 2208  
 2209  
 2210  
 2211  
 2212  
 2213



(analysis)  
 OK, the user needs us to find the person holding a microphone in the hand. To accomplish this task, I need to break it down into the following steps:  
 - Step 1: Check the hands of each person to see if they are holding a microphone.

# Step 1: Check the hands of each person to see if they are holding a microphone

- Person 1: He is holding a microphone in his right hand. → ✓
- Person 2: His hands are on his hips, not holding a microphone. → ✗
- Person 3: His hands are gesturing, not holding a microphone. → ✗
- Person 4: He is holding a microphone in his left hand. → ✓
- Person 5: His hands are making a gesture, not holding a microphone. → ✗
- Person 6: His hands are positioned near his body, not holding a microphone. → ✗
- Person 7: His hands are gesturing, not holding a microphone. → ✗

# Summarize and Re - Check answer  
 Let's now recheck our answer and put ✓ for the target people and ✗ for others:

- Person 1: Holding a microphone → ✓
- Person 2: Not holding a microphone → ✗
- Person 3: Not holding a microphone → ✗
- Person 4: Holding a microphone → ✓
- Person 5: Not holding a microphone → ✗
- Person 6: Not holding a microphone → ✗
- Person 7: Not holding a microphone → ✗

(analysis)  
 (answer)  
 To summarize, person 1 and person 4 are the persons holding a microphone in the hand.  
 (answer)

Figure 43: Visualization of GPT-4o's output on the *interaction (outer)* subset.

```

2214
2215
2216 Example 1 :
2217 ...
2218 METAINFO:
2219 {
2220     "total_num_of_person": 3,
2221     "answer": "person 1",
2222     "order_from_left_to_right": [person 3, person 1, person 2]
2223 }
2224
2225 Question: The woman to the left of the woman in blue dress
2226
2227 The output should be:
2228 (analysis)
2229 OK, the user needs us to detect the woman to the left of the woman in the blue dress. In order to accomplish this task, I need to break it down into the following steps:
2230 Step 1: Find the woman in the blue dress.
2231 Step 2: Find the woman to her left.
2232 Let's start our analysis:
2233
2234 # Step 1: Find the woman in the blue dress
2235 I see three people in this image. Let's determine who is wearing a blue dress and put a ⚠ if it is and ✗ if it is not.
2236 - Person 1: She is wearing a pink dress. ✗
2237 - Person 2: She is wearing a blue dress. ⚠
2238 - Person 3: He is wearing a black suit with a pink tie. ✗
2239
2240 So Person 2 is the woman in the blue dress. Now I need to find the woman to her left.
2241
2242 # Step 2: Find the woman to the left of Person 2
2243 The order from left to right is [Person 3, Person 1, Person 2]. So Person 1 is to the left of Person 2. And Person 1 is a woman.
2244
2245 # Summarize and Re - Check answer
2246 Let's now recheck our answer and put ✓ for the target person and ✗ for others
2247 - Person 1: She is the woman to the left of the woman in the blue dress. ✓
2248 - Person 2: She is the woman in the blue dress. ✗
2249 - Person 3: He is a man, not the woman we are looking for. ✗
2250 (analysis)
2251 To summarize, Person 1 is the woman to the left of the woman in the blue dress.
2252 (answer)
2253

```

Figure 44: In-context prompt for *reasoning (inner position)* subset in HumanRef-CoT.


Referring Phrase: the person holding a microphone in the hand

```

2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267 (analysis)
2268 OK, the user needs us to detect the woman to the right of the man wearing a gold tie. In
2269 order to accomplish this task, I need to break it down into the following steps:
2270 Step 1: Find the man wearing a gold tie.
2271 Step 2: Determine the person to the right of the man found in Step 1 and check if it is a
2272 woman.
2273 Let's start our analysis:
2274
2275 # Step 1: Find the man wearing a gold tie
2276 I see five people in this image. Let's check their ties and put a ⚠ if it is a gold tie and ✗ if it
2277 is not.
2278 - Person 1: He is wearing a gold - patterned tie. ⚠
2279 - Person 2: She is a woman with long blonde hair and no tie. ✗
2280 - Person 3: She is a woman with short blonde hair and no tie. ✗
2281 - Person 4: Only part of his head is visible and no tie is seen. ✗
2282 - Person 5: Only a small part of his body is visible and no tie is seen. ✗
2283
2284 So Person 1 is the man wearing a gold tie.
2285
2286 # Step 2: Find the person to the right of Person 1 and check the gender
2287 The order from left to right is [Person 4, Person 1, Person 2, Person 3, Person 5].
2288 The person to the right of Person 1 is Person 2. Person 2 is a woman.
2289
2290 # Summarize and Re - Check answer
2291 Let's now recheck our answer and put ✓ for the target person and ✗ for others:
2292 - Person 1: He is the man wearing a gold tie (reference person). ⚠
2293 - Person 2: She is the woman to the right of the man wearing a gold tie. ✓
2294 - Person 3: She is a woman but not to the right of the man with the gold tie. ✗
2295 - Person 4: He is not the reference person and no gold tie is seen. ✗
2296 - Person 5: He is not relevant to the reference person and no gold tie is seen. ✗
2297 (analysis)
2298 (answer)
2299 To summarize, Person 2 is the woman to the right of the man wearing a gold tie.
2300 (answer)

```

Figure 45: Visualization of GPT-4o's output on the *reasoning (inner position)* subset.

```

2268 Example 1 :
2269 ...
2270 METAINFO:
2271 {
2272     "total_num_of_person": 4,
2273     "answer": "person 3, person 2, person 1",
2274     "order_from_left_to_right": [person 4, person 2, person 3, person 1]
2275 }
2276
2277 Question: Person wearing glasses but not sitting on a chair
2278
2279 The output should be:
2280 (analysis)
2281 OK, the user needs us to detect the person wearing glasses but not sitting on a chair. To accomplish this task, I need to break it down into the following steps:
2282 Step 1: Find all person wearing glasses.
2283 Step 2: Find the person in Step 1 that are sitting on a chair
2284 Step 3: Remove person in Step 2 from Step 1 to get the final result
2285
2286 # Step 1: Find all person wearing glasses.
2287 I see five people in this image, let's look at their face and put a ▲ if they are wearing glasses and ✗ if it is not.
2288 - Person 1: A man wearing a red shirt, he is wearing glasses → ▲
2289 - Person 2: A man wearing a light-blue shirt, he is wearing glasses → ✗
2290 - Person 3: A woman with blonde hair, she is not wearing glasses → ✗
2291 - Person 4: A woman in a white top, she is not wearing glasses → ▲
2292 - Person 5: A man wearing a light blue long sleeved shirt, he is wearing glasses → ▲
2293
2294 So Person 1, Person 4 and Person 5 are wearing glasses
2295
2296 # Step 2: Find the person in Step 1 that are sitting on a chair
2297 Let's look at the three people in Step 1 and put a ▲ if they are sitting on a chair and ✗ if it is not.
2298 - Person 1: He is sitting on a chair → ▲
2299 - Person 2: He is sitting on a chair → ▲
2300 Person 5: He is standing instead of sitting → ✗
2301
2302 # Step 3 Remove person in Step 2 from Step 1 to get the final result
2303 Remove Person 1 and Person 2 and we get Person 5 that meets the description
2304
2305 Let's now recheck our answer and put ✓ for the target person and ✗ for others
2306 Person 1: wearing glasses but sitting on a chair → ✗
2307 Person 2: wearing glasses but sitting on a chair → ✗
2308 Person 3: not wearing glasses → ✗
2309 Person 4: not wearing glasses → ✗
2310 Person 5: wearing glasses and not sitting on a chair → ✓
2311
2312 (answer)
2313 To summarize, person 5 is the third person from left to right
2314 (answer)
2315 ...

```

Figure 46: In-context prompt for *reasoning (attribute)* subset in HumanRef-CoT.



Referring Phrase: the person holding a microphone in the hand

(analysis)

OK, the user needs us to detect the person straddling someone's shoulders but not wearing long blue ripped jeans. To accomplish this task, I need to break it down into the following steps:

Step 1: Find all person straddling someone's shoulders.

Step 2: Find the person in Step 1 that are wearing long blue ripped jeans.

Step 3: Remove person in Step 2 from Step 1 to get the final result

# Step 1: Find all person straddling someone's shoulders.

I see four people in this image, let's look at their position and put a ▲ if they are straddling someone's shoulders and ✗ if it is not.

- Person 1: A boy straddling a man's shoulders → ▲
- Person 2: A woman not straddling anyone's shoulders → ✗
- Person 3: A boy straddling a woman's shoulders → ▲
- Person 4: A man not straddling anyone's shoulders → ✗

So Person 1 and Person 3 are straddling someone's shoulders

# Step 2: Find the person in Step 1 that are wearing long blue ripped jeans.

Let's look at the two people in Step 1 and put a ▲ if they are wearing long blue ripped jeans and ✗ if it is not.

- Person 1: He is wearing long blue ripped jeans → ▲
- Person 3: He is wearing dark-colored pants, not long blue ripped jeans → ✗

# Step 3: Remove person in Step 2 from Step 1 to get the final result

Remove Person 1 from Step 1 and we get Person 3 that meets the description

# Summarize and Re-Check answer

Let's now recheck our answer and put ✓ for the target person and ✗ for others

- Person 1: straddling someone's shoulders and wearing long blue ripped jeans → ✗
- Person 2: not straddling someone's shoulders → ✗
- Person 3: straddling someone's shoulders and not wearing long blue ripped jeans → ✓
- Person 4: not straddling someone's shoulders → ✗

(answer)

To summarize, the person is person 3

(answer)

...

Figure 47: Visualization of GPT-4o's output on the *reasoning (attribute)* subset..

```

2322
2323
2324
2325 Example:
2326 METAINFO:
2327 {
2328     "total_num_of_person": 4,
2329     "answer": "person 1",
2330     "order_from_left_to_right": [person 4, person 1, person 2, person 3]
2331 }
2332 (analysis)
2333 OK, the user needs us to detect Brad Pitt. In order to accomplish this task, I need to break it down into the following step:
2334 - Step 1: Identify Brad Pitt among the 8 people based on his appearance characteristics. Let's start our analysis:
2335 # Step 1: Identify Brad Pitt among the 8 people based on his appearance characteristics
2336 - Person 1: Has light - colored hair, similar facial features to Brad Pitt, and is wearing a black jacket over a white shirt, which matches
2337 the reference image of Brad Pitt. ▲
2338 - Person 2: A man with dark hair sitting on a red couch, does not match Brad Pitt's appearance. ✗
2339 - Person 3: A woman with long - brown hair, clearly not Brad Pitt. ✗
2340 - Person 4: A man with dark hair and a dark shirt, does not match Brad Pitt's appearance. ✗
2341 # Summarize and Re - Check answer
2342 Let's now recheck our answer and put ✓ for the target person and ✗ for others
2343 - Person 1: Matches Brad Pitt's appearance characteristics. ✓
2344 - Person 2: Does not match Brad Pitt's appearance. ✗
2345 - Person 3: Does not match Brad Pitt's appearance. ✗
2346 - Person 4: Does not match Brad Pitt's appearance. ✗
2347 (analysis)
2348 (answer)
2349 To summarize, Person 1 is Brad Pitt
2350 (answer)

```

Figure 48: In-context prompt for *celebrity recognition* subset in HumanRef-CoT.

```

2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364 Referring Phrase: Donald Sutherland
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375

```

(analysis)

OK, the user needs us to detect Donald Sutherland. To accomplish this task, I need to break it down into the following steps:

- Step 1: Recall what Donald Sutherland looks like and match his appearance among the people in the image.

# Step 1: Recall what Donald Sutherland looks like and match his appearance among the people in the image.

I know that Donald Sutherland is an older - looking man with white hair and a beard.

- Person 1: He is a younger - looking man with dark hair and no beard. He does not match the appearance of Donald Sutherland. ✗

- Person 2: He is a man with short, light - brown hair and a short beard. He does not match the appearance of Donald Sutherland. ✗

- Person 3: He is an older man with white hair and a beard. He matches the appearance of Donald Sutherland. ▲

- Person 4: He is a man with short, light - colored hair and no beard. He does not match the appearance of Donald Sutherland. ✗

# Summarize and Re - Check answer

Let's now recheck our answer and put ✓ for the target person and ✗ for others

- Person 1: Does not match the appearance of Donald Sutherland. ✗

- Person 2: Does not match the appearance of Donald Sutherland. ✗

- Person 3: Matches the appearance of Donald Sutherland. ✓

- Person 4: Does not match the appearance of Donald Sutherland. ✗

(analysis)

(answer)

To summarize, Person 3 is Donald Sutherland

(answer)


Figure 49: Visualization of GPT-4o's output on the *celebrity recognition* subset..

2376  
 2377  
 2378  
 2379  
 2380 Example:  
 2381 METAINFO:  
 2382 {  
 2383   "total\_num\_of\_person": 4,  
 2383   "reference\_answer": "there is not person that meets the description"  
 2384 }  
 2385 (analysis)  
 2385 OK, the user needs us to detect all the people with guns in the picture. I need to look at the objects held in each person's hands and determine if they are guns or not. I see 7 people in the diagram, let's analyze the objects held by each of them:  
 2386  
 2387 - Person 1: She is a woman with long black hair, her right fist is outstretched but she is not holding a gun, and we can't see her left hand, so we can tell that she is not holding a gun.  $\times$   
 2388 - Person 2: A female with brown hair wearing a thin red jacket. She has her left hand up and looks nervous, but she's not holding a gun either.  $\times$   
 2389 - Person 3: A bald policeman in a black jacket, we can't see his hands, so we don't think he's holding a gun either.  $\times$   
 2390 - Person 4: A bald black male with a big mouth, he has his hands open but he is also not holding a gun.  $\times$   
 2391 Let's now finalize:  
 2392 Person 1: not holding a gun  $\rightarrow \times$   
 2392 Person 2: not holding a gun  $\rightarrow \times$   
 2393 Person 3: not holding a gun  $\rightarrow \times$   
 2393 Person 4: not holding a gun  $\rightarrow \times$   
 2394 (analysis)  
 2395 (answer)  
 2396 To summarize, there is no person in the image that meets the description.  
 2396 (Answer)

Figure 50: In-context prompt for *rejection* subset in HumanRef-CoT.

2397 Referring Phrase: the person wearing sunglasses  
 2398  
 2399  
 2400  
 2401  
 2402  
 2403  
 2404  
 2405  
 2406  
 2407  
 2408  
 2409  
 2410  
 2411  
 2412  
 2413  
 2414  
 2415  
 2416  
 2417  
 2418  
 2419  
 2420  
 2421  
 2422  
 2423  
 2424  
 2425  
 2426  
 2427  
 2428  
 2429

(analysis)  
 OK, the user needs us to detect the person wearing sunglasses. To accomplish this task, I need to break it down into the following steps:  
 - Step 1: Examine the attire and accessories of each person to find those wearing sunglasses.

# Step 1: Examine the attire and accessories of each person to find those wearing sunglasses.  
 - Person 1: A young girl in a pink - colored coat. She is not wearing sunglasses.  
 - Person 2: A boy in a checkered shirt and a black tie. He is not wearing sunglasses.  
 - Person 3: A young girl in a beige - colored coat with a patterned collar. She is not wearing sunglasses.  
 - Person 4: A woman in a metallic - green jacket and a patterned dress. She is not wearing sunglasses.  
 - Person 5: A man in a dark suit and a light - colored shirt with a tie. He is not wearing sunglasses, but he is the person mentioned in the condition related content (implied from the hint).

# Summarize and Re - Check answer  
 Let's now recheck our answer and put  $\checkmark$  for the target person and  $\times$  for others  
 - Person 1: not wearing sunglasses  $\rightarrow \times$   
 - Person 2: not wearing sunglasses  $\rightarrow \times$   
 - Person 3: not wearing sunglasses  $\rightarrow \times$   
 - Person 4: not wearing sunglasses  $\rightarrow \times$   
 - Person 5: not wearing sunglasses  $\rightarrow \times$

(analysis)

(answer)  
 To summarize, there are no people that meet the requirements as none of the five people are wearing sunglasses.  
 (answer)

Figure 51: Visualization of GPT-4o's output on the *rejection* subset..