ORIC: Benchmarking Object Recognition under Contextual Incongruity in Large Vision-Language Models

Anonymous CVPR submission

Paper ID ***

Abstract

Large Vision-Language Models (LVLMs) excel at captioning, visual question answering, and robotics by combining vision and language, yet they often miss obvious objects or hallucinate nonexistent ones in atypical scenes. We examine these failures through the lens of uncertainty, focusing on contextual incongruity*, where objects appear unexpectedly or fail to appear in expected contexts, and show that such cases increase recognition difficulty for state-of-the-art LVLMs. To study this regime, we introduce the* ***Object Recognition in Incongruous Context (ORIC) framework****, which constructs incongruous object-context pairs through two complementary strategies:* ***(1) LLM-guided sampling*** *to identify hard-to-recognize objects present in the image and* ***(2) CLIP-guided sampling*** *to mine plausible but absent ones. Applied to MSCOCO, ORIC produces ORIC-Bench and ORIC-style training data. Evaluating 18 LVLMs and 2 open-vocabulary detectors reveals substantial performance drops and bias patterns under incongruous contexts. Fine-tuning Qwen3-VL-8B-Instruct with Visual Reinforcement Fine-Tuning on 600 ORIC-style samples improves results on ORIC-Bench, AMBER, and HallusionBench. Overall, we show that contextual incongruity is a key source of uncertainty and provide tools for more reliable LVLMs.*

1. Introduction

Large Vision-Language Models (LVLMs) have achieved remarkable progress across image captioning [16], visual question answering (VQA) [60], robotics [22], and embodied AI [73], driven by their ability to integrate visual and textual modalities. A core skill underlying these advances is accurate object recognition [12], essential for reliable perception and high-level reasoning [83]. However, despite strong benchmark scores, LVLMs remain vulnerable to two key failures: **(1) object misidentification**, where existing objects are missed [49]; and **(2) object hallucination**, where nonexistent objects are falsely recognized [15, 58], which undermine downstream reliability [20, 36]. A particularly challenging regime that amplifies these issues is **contextual incongruity**, where objects appear in unexpected settings or are absent from expected ones. Under such conditions, LVLMs often misread visual evidence, either overlooking valid objects or hallucinating contextually plausible ones. For instance, as shown in the left side of Fig. 1, GPT-5 [53] correctly identifies a mouse but fails to recognize a prominent train in an office; in the right side of Fig. 1, it correctly denies a car but hallucinates a sports ball on a baseball field. These observations echo cognitive findings that unexpected contexts disrupt recognition [30, 54, 70].

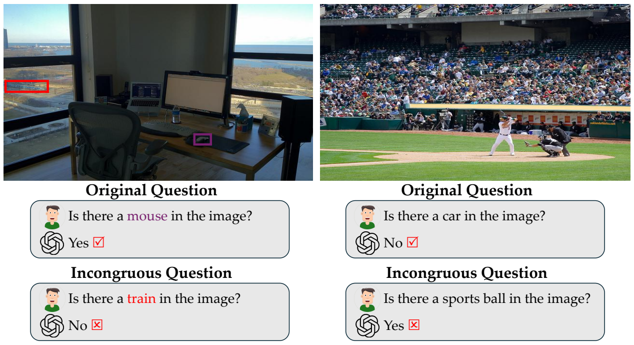

Original Question
Is there a mouse in the image?
Yes ☑
Incongruous Question
Is there a train in the image?
No ☒
Original Question
Is there a car in the image?
No ☑
Incongruous Question
Is there a sports ball in the image?
Yes ☒

Figure 1. **Contextual Incongruity Leads to Recognition Failures.** This figure illustrates how incongruous contexts cause two primary errors: misidentification of present objects and hallucination of absent ones. **Left (Misidentification):** In an office, GPT-5 identifies the expected "mouse" (purple) but fails to recognize the out-of-context "train" (red). **Right (Hallucination):** On a baseball court, the model correctly denies an unrelated "car" but hallucinates a plausible yet non-existent "sports ball."

Recent theory attributes language model errors to learning under uncertainty with binary scoring, which rewards guessing over abstaining [31]. In our setting, answering a binary existence question can be formalized as estimating $P(a \mid q, I)$, where $a \in \{\text{yes}, \text{no}\}$, q denotes the question, and $I = (\text{ROI}, \text{context})$ represents the image composed of a ROI containing the queried object and its surrounding scene. As illustrated in the left side of Fig. 1, the train area serves as the ROI, while the office environment represents the context. When evidence from the ROI is weak, contextual priors $P(a \mid q, \text{context})$ tend to dominate the inference. If the context strongly implies that an object should exist (e.g., a sports ball on a baseball field), the model is biased toward answering "yes," resulting in hallucinations. Conversely, when the context implies that the object is unlikely to appear (e.g., a train in an office), the model confidently predicts "no," causing misidentification. In both scenarios, contextual incongruity heightens uncertainty by opposing weak local evidence with strong scene-level priors, leading to recognition errors.

From this uncertainty perspective, existing benchmarks mainly target other sources while keeping object-context consistency. POPE [35] tests recognition under strong statistical or textual priors. AMBER [66] evaluates discriminative tasks involving object existence, attributes, and relations. HallusionBench [24] examines visual-dependent questions that require image context, such as visual illusions and figures. However, across these benchmarks, queried objects remain context-consistent with their scenes, leaving the high-uncertainty regime where weak local evidence opposes strong contextual priors largely unexplored.

Motivated by this gap, we systematically examine how contextual incongruity affects object recognition in LVLMs. To analyze this effect under controlled conditions, we introduce the **Object Recognition in Incongruous Context (ORIC) framework**, which constructs incongruous object-context pairs for both evaluation and training. ORIC integrates two complementary strategies: (1) *LLM-guided sampling*, where GPT-5 identifies existing objects that are difficult to recognize in atypical contexts; and (2) *CLIP-guided sampling*, where CLIP [56] mines plausible yet nonexistent objects. Applied to the MSCOCO validation set, ORIC produces a balanced binary benchmark, **ORIC-Bench**, while applying the same pipeline to the training split yields ORIC-style samples. Evaluating 18 LVLMs and two open-vocabulary detectors on **ORIC-Bench** reveals that even top-performing models on standard benchmarks fail under contextual incongruity, exposing persistent recognition gaps. To mitigate these uncertainty-driven errors, we fine-tune Qwen3-VL-8B-Instruct [3, 4] using Visual Reinforcement Fine-Tuning (Visual-RFT) [44] on 600 ORIC-style samples, improving performance on not only ORIC, but also AMBER and HallusionBench, with responses more aligned with human reasoning. Overall, our main contributions are:

- **Problem Identification.** We identify *contextual incongruity* as an overlooked cause of visual uncertainty in LVLMs, which degrades recognition performance.
- **ORIC Framework.** We introduce ORIC, which builds incongruous object-context pairs via LLM- and CLIP-guided sampling for evaluation and training.
- **Model Evaluation.** We test 18 LVLMs and 2 detectors on ORIC, showing that the task is difficult and reveals clear bias patterns.
- **ORIC-driven Uncertainty Mitigation.** Visual-RFT of Qwen3-VL-8B-Instruct on ORIC-style data lowers uncertainty-driven errors and yields more human-aligned performance across benchmarks.

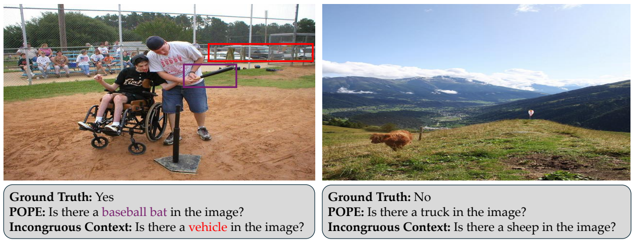

Ground Truth: Yes
POPE: Is there a baseball bat in the image?
Incongruous Context: Is there a vehicle in the image?
Ground Truth: No
POPE: Is there a truck in the image?
Incongruous Context: Is there a sheep in the image?

Figure 2. **Comparison of POPE and Incongruous Context Questions.** Both examples use the same image but differ in target objects. **Left:** In a baseball field, POPE targets a baseball bat (purple), while ours targets a large vehicle (red), which is less related to the scene and thus more incongruous. Both labels are "yes." **Right:** In a rural scene with a cow, POPE targets a truck, while our question targets a sheep—more contextually plausible but still absent, increasing incongruity. Both labels are "no."

2. Contextual Incongruity and Uncertainty

This section examines how contextual incongruity affects object recognition under uncertainty and provides empirical evidence that it significantly degrades model performance.

2.1. Theoretical Formulation

Mentioned on Sec. 1, answering a binary existence query is estimating $P(a \mid q, I)$ for $a \in \{\text{yes}, \text{no}\}$, with the image represented as $I = (\text{ROI}, \text{context})$. Let o be the queried object class and c the scene context (e.g., *baseball field*, *office*). Training data induce a joint $P(o, c)$ over object-context pairs. Existing benchmarks mostly sample head regions of this distribution, where pairs are frequent and consistent; both $P(a_{\text{gt}} \mid q, \text{ROI})$ and $P(a_{\text{gt}} \mid q, \text{context})$ are high for the ground-truth a_{gt}, yielding low uncertainty and allowing co-occurrence heuristics to perform well.

However, we focus on the *high-uncertainty regime* induced by contextual incongruity, where ROI evidence and contextual priors disagree. Typical examples include an unusual object in a familiar scene (e.g., a train in an office) or a missing object that the scene strongly suggests (e.g., no ball on a baseball field). In such cases, the posterior based on the ROI alone is diffuse, with $P(\text{yes} \mid q, \text{ROI})$ and $P(\text{no} \mid q, \text{ROI})$ being similar in magnitude, while the context strongly favors one of them. Theory [31] suggests that binary supervision rewarding guesses drives models toward contextual priors instead of uncertainty, causing hallucinations of plausible objects or overconfident rejections in incongruous contexts.

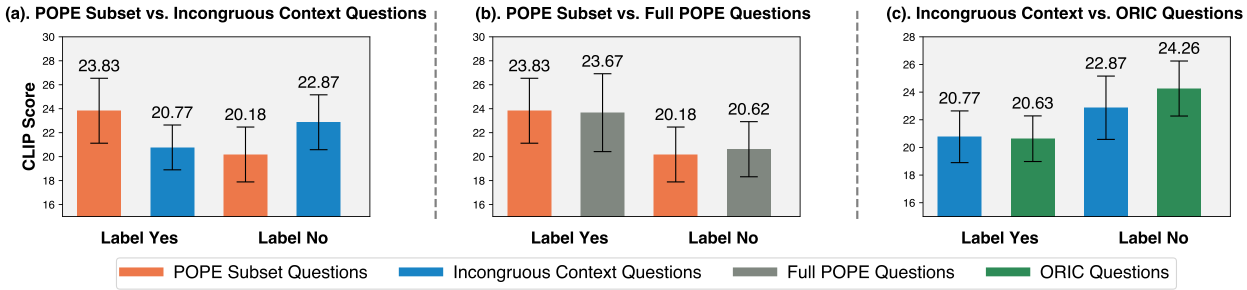


Figure 3. **Object–Context Congruity via CLIPScore.** CLIPScore quantifies alignment between queried objects and scene context. **(a)** For "yes" questions, POPE subset yields higher scores than incongruous variants (23.83 vs. 20.77); for "no" questions, the reverse holds (22.87 vs. 20.18), indicating stronger misleading cues. **(b)** The sampled POPE subset shows consistent CLIPScore distribution with the full dataset, confirming its representativeness. **(c)** ORIC questions exhibit even higher incongruity (e.g., 24.26 for "no"), reinforcing the contextual challenge. Subplots (a) and (c) share images but differ in queried objects. Error bars show 95% confidence intervals.

2.2. Empirical Analysis of Contextual Incongruity

To assess how contextual incongruity affects LVLMs, we conduct a controlled study based on the POPE benchmark [35]. We sample 25 "yes" and 25 "no" context-consistent questions, then keep each image and label fixed while replacing the queried object, creating paired context-incongruous questions. For example, in the left side of Fig. 2, the baseball-field question "Is there a baseball bat in the image?" is changed to "Is there a vehicle in the image?". In the right side of Fig. 2, the rural-scene question "Is there a truck in the image?" becomes "Is there a sheep in the image?" even though the image contains only a cow. We evaluate four representative LVLMs including GPT-5-08-07 [28], Janus-Pro-7B [10], InternVL3-9B [85], and Qwen3-VL-8B-Instruct using macro accuracy, precision, recall, and F1 (see formulas in Appendix A.4).

Model	POPE Subset			Incongruous Context		
	Prec.	Rec.	F1.	Prec.	Rec.	F1.
Janus-Pro-7B	96.30	96.00	95.99	58.01	58.00	**57.98**
InternVL3-9B	96.30	96.00	95.99	56.16	56.00	**58.00**
Qwen3-VL-8B-Instruct	98.08	98.00	98.00	61.90	60.00	**58.33**
GPT-5-08-07	100.00	100.00	100.0	61.27	60.32	**60.79**

Table 1. **Model Performance on POPE vs. Incongruous Context Questions.** This table reports macro precision (Prec.), recall (Rec.), and F1 score (F1) for four LVLMs on the POPE benchmark and a set of manually curated questions. Although all models perform well on the POPE subset, they struggle with incongruous context questions.

Table 1 reports results on the original context-consistent questions and their context-incongruous counterparts. All four models achieve near-perfect performance on the original subset (macro F1 between 96.0 and 100.0), indicating that these questions are easy for current LVLMs. However, macro F1 drops dramatically to around 60 on the incongruous questions, despite the images being identical. This sharp degradation cannot be attributed to low-level visual difficulty and instead points to failures induced purely by breaking object–context compatibility.

To quantify how our modifications alter object–background associations, we further analyze CLIPScores between each image and the textual description of the queried object. Given an image I and a question-related object name O, we use CLIP [56] to extract visual and textual embeddings $f_I, f_O \in \mathbb{R}^d$, normalize them as $\hat{f}_I = f_I/\|f_I\|$ and $\hat{f}_O = f_O/\|f_O\|$, and compute

$$\text{CLIPScore}(I, O) = \hat{f}_I^\top \hat{f}_O = \frac{f_I^\top f_O}{\|f_I\|\,\|f_O\|} \times 100. \quad (1)$$

Fig. 3(a) plots CLIPScores for 50 pairs of original and context-incongruous questions. For "yes" questions, original objects show a higher mean score (23.83) than their incongruous replacements (20.77), indicating weaker contextual alignment. For "no" questions, the trend reverses: context-incongruous objects score higher (22.87 vs. 20.18), suggesting that the background strongly implies the presence of objects that are actually absent. The middle subplot in Fig. 3(b) exhibits the same patterns as the full benchmark, confirming that our subset is representative. Together, these results show that contextual incongruity creates a high-uncertainty regime for LVLMs, where models that perform reliably on standard questions experience substantial accuracy drops. This motivates ORIC as a framework that systematically constructs data with incongruous context for both evaluation and training.

3. The ORIC Framework

This section introduces ORIC, which generates object-recognition questions under contextual incongruity, each framed as a binary "yes" or "no" label of object presence.

3.1. ORIC Construction Method

Positive Questions (Existing Objects): Contextual incongruity arises when objects appear in unexpected settings, creating high uncertainty. Therefore, our objective is to generate questions that deliberately minimize

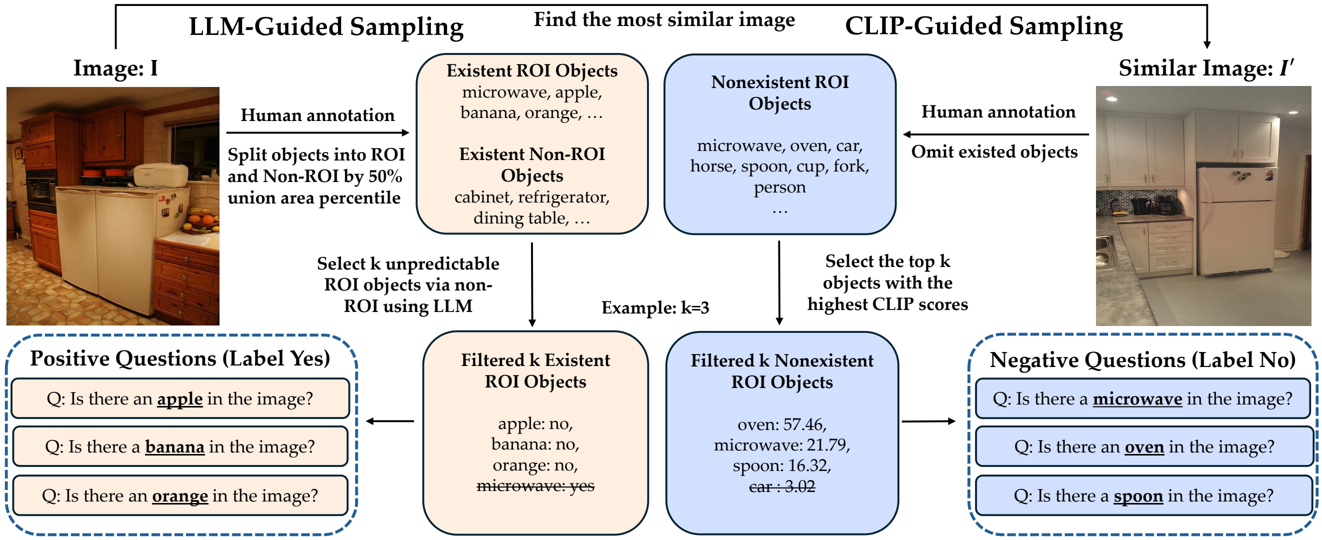

LLM-Guided Sampling
Find the most similar image
CLIP-Guided Sampling
Image: I
Similar Image: I′
Human annotation
Split objects into ROI and Non-ROI by 50% union area percentile
Existent ROI Objects
microwave, apple, banana, orange, …
Existent Non-ROI Objects
cabinet, refrigerator, dining table, …
Nonexistent ROI Objects
microwave, oven, car, horse, spoon, cup, fork, person
…
Human annotation
Omit existed objects
Select k unpredictable ROI objects via non-ROI using LLM
Example: k=3
Select the top k objects with the highest CLIP scores
Positive Questions (Label Yes)
Q: Is there an apple in the image?
Q: Is there a banana in the image?
Q: Is there an orange in the image?
Filtered k Existent ROI Objects
apple: no,
banana: no,
orange: no,
~~microwave: yes~~
Filtered k Nonexistent ROI Objects
oven: 57.46,
microwave: 21.79,
spoon: 16.32,
~~car : 3.02~~
Negative Questions (Label No)
Q: Is there a microwave in the image?
Q: Is there an oven in the image?
Q: Is there a spoon in the image?

Figure 4. **ORIC Method Overview.** This figure shows two construction methods of the ORIC. **LLM-Guided Sampling (Positive Question Construction):** First, given an image I, objects are classified as ROI if their combined bounding box area is under 50%; otherwise, they are non-ROI. Next, we query the LLM (GPT-5) with textual categories of non-ROI objects to predict the existence of each ROI object based on common sense and co-occurrence. Finally, we select the top k unpredictable ROI objects (e.g., $k = 3$) for which the LLM predicts "no" (e.g., apple, banana, and orange). **CLIP-Guided Sampling (Negative Question Construction):** A similar image I' is identified using cosine distance from I. We then compute the CLIPScore for each nonexistent ROI object against I' and select the top k nonexistent ROI objects based on their scores. For example, the top three are an oven (57.46), a microwave (21.79), and a spoon (16.32).

Category	HallusionBench	POPE	MM-Vet v2	AMBER	Hallu-PI	ORIC-Bench
Image Count	346	500	517	1k	1.2k	**1k**
Contextual Incongruity	✗	✗	✗	✗	✗	✓
Missed / Hallucinated Recognition	Hallucinated only	Both	Both	Both	Hallucinated only	**Both**

Table 2. **Benchmark Comparison.** Benchmarks compared by image count, contextual incongruity, and error types.

background-object associations, utilizing **LLM-guided sampling**. We define the objects targeted for recognition as ROI, while background contexts consist of non-ROI elements. Formally, as illustrated on the left side of Fig. 4, given an image I containing objects $\mathcal{O} = \{o_i = (n_i, \{B_{ij}\}_{j=1}^{m_i})\}_{i=1}^{N}$, where n_i is the object's name and B_{ij} denotes the j-th bounding box associated with object o_i, we categorize objects into ROI and non-ROI based on their bounding box coverage. We then select k ROI objects as positive question candidates, where k is the desired number of selected objects. The total area covered by each object's bounding boxes is calculated as:

$$A_i = \text{area}\Big(\bigcup_{j=1}^{m_i} B_{ij}\Big), \tag{2}$$

where the function $\text{area}(\cdot)$ computes pixel area, and then we split $\mathcal{O}$ into two disjoint sets based on the 50^{th} percentile: $\mathcal{O}_{\text{ROI}} = \{o_{(i)} \mid A_{(i)} < M_{50}(A)\}$ and $\mathcal{O}_{\text{nonROI}} = \{o_{(i)} \mid A_{(i)} \geq M_{50}(A)\}$, where $M_{50}(A)$ denotes the median area of the union of bounding boxes (i.e., the 50th-percentile area of the union of bounding boxes among all objects). We then use GPT-5 to filter ROI candidates. Specifically, the LLM is queried to determine whether each ROI object is logically consistent with the provided non-ROI object categories. The verification function is defined as:

$$f(o) = \begin{cases} 1, & \text{if LLM}(o, \mathcal{O}_{\text{nonROI}}) = \text{“no”}, \\ 0, & \text{otherwise}. \end{cases} \tag{3}$$

The function $\text{LLM}(o, \mathcal{O}_{\text{nonROI}})$ returns "no" if the ROI object is unexpected based on common sense and typical co-occurrence. Objects receiving a ”no” from GPT-5 form the positive candidate set $\mathcal{C}$. Positive questions are generated by randomly selecting k objects from $\mathcal{C}$. For detailed pseudocode and prompts, refer to Appendix A.1.

Negative Questions (Nonexistent Objects): LVLMs often hallucinate objects when strong contextual cues make nonexistent items seem plausible, reflecting the high uncertainty created by incongruous contexts. Therefore, our goal is to generate questions that enhance the correlation between nonexistent ROI objects and non-ROI elements by leveraging **CLIP-guided sampling**. As depicted on the right side of Fig. 4, we first identify the most visually similar image I' to a query image I using the CLIP model's image encoder, which helps curate a more diverse set of retrieved images. Formally, given images $\{I_1, \ldots, I_n\}$ and a query image I_q, visual embeddings are extracted via ViT: $e = ViT(I)$. The image similarity is measured using cosine distance:

$$D(I_q, I_i) = 1 - \frac{\mathbf{e}_q \cdot \mathbf{e}_i}{\|\mathbf{e}_q\|\|\mathbf{e}_i\|}, \tag{4}$$

where $\mathbf{e}_q$ and $\mathbf{e}_i$ represent embeddings of image I_q and I_i, respectively. The most similar image I' minimizes this distance. Next, given the most similar image I' and a set of nonexistent ROI objects $\mathcal{O}_{\text{non}} = \{n_i\}_{i=1}^{M}$, where n_i represents an individual nonexistent ROI object and M is the total number of nonexistent ROI objects considered in the set $\mathcal{O}_{\text{non}}$. For each n_i, a text description T_i is generated in the form of "*an image contains* n_i." We compute the similarity score for each object as $s_i = \text{CLIPScore}(I', T_i)$. The objects are then sorted by s_i, and the top k nonexistent ROI objects are selected to form $\mathcal{O}_{\text{non}}$ for negative question generation. See Appendix A.2 for the detailed algorithm.

3.2. ORIC Statistics

Human Evaluation: We sampled 150 "yes" and 150 "no" questions using ORIC framework and manually verified **(1)** object labeling accuracy and **(2)** contextual incongruity. The low 2% error rate confirms the robustness of our pipeline. Appendix E.1 provides six error cases, and additional correct examples are shown in Appendix E.2.

CLIPScore for ROI–Background Analysis: We compared ORIC-generated questions with incongruous context questions in Sec. 2 using a CLIPScore-based method. Specifically, we generated 50 ORIC questions (25 for each label, "yes" and "no") corresponding to the same images used in the previous incongruous context questions. As illustrated in Fig. 3(c), CLIP scores for "yes" questions were nearly identical between ORIC (20.77) and incongruous context questions (20.63), suggesting similar contextual alignment. However, for "no" questions, ORIC achieved higher CLIP scores (24.25 vs. 22.87), indicating a stronger correlation between the nonexistent object and the visual context, thereby creating a more incongruous context.

4. ORIC-Bench Experiments and Analysis

We evaluate **18** LVLMs and 2 open-vocabulary detectors on ORIC-Bench under contextual incongruity, analyzing performance, architecture, class bias, and object-size effects. The 11-LVLM summary is in Table 3, and the full 18-LVLM results are in Appendix Table 10. Ablations and POPE comparisons in Appendices B.3.1 and B.3.2 show that ORIC-Bench is more challenging and discriminative for LVLMs.

4.1. Experimental Setup

ORIC-Bench Setup and Evaluated Models. We evaluate on ORIC-Bench, built with the ORIC using 1,000 MSCOCO [40] validation images (avoiding leakage). Each image pair yields two present-object and two absent-object queries, resulting in 1,000 "yes" and 1,000 "no" questions. As shown in Table 2, ORIC-Bench uniquely introduces contextual incongruity and jointly tests both missed and hallucinated recognition. We evaluate **18** LVLMs (vision-encoder-based, vision-encoder-free, and closed-source) and **2** open-vocabulary detectors (Grounding DINO 1.5 Pro [57] and OWLv2 [50]). Detailed model specifications are provided in Appendix B.1.

Evaluation Protocol and Metrics. Ambiguous LVLM outputs are resolved using MMBench's two-step matching [43]: we first heuristically extract explicit "yes" or "no" labels from each output; if none are found, GPT-5-08-07 is prompted with the question, answer options, and the raw response to infer the label. All experiments are conducted on a single NVIDIA H100 with temperature 0 and a 1,024-token limit. Each LVLM is tested under four prompts, and results are averaged. Detectors jointly process present and absent objects: a detection with confidence ≥ 0.25 counts as "yes," otherwise "no." We report the yes-predictions proportion (YP), macro precision, recall, and F1, as well as class-wise precision, recall, and F1 for yes and no. See Appendix B.2 for prompt details and Appendix A.4 for metric details.

4.2. ORIC-Bench Results and Analysis

Table 3 presents the results of **11** LVLMs and 2 open-vocabulary detectors on ORIC-Bench. We analyze overall performance, architectural differences, and the impact of contextual incongruity.

Overall Performance: Qwen3-VL-8B-Instruct achieves the highest overall F1 of 79.55, surpassing GPT-5 (78.61) and strong vision-encoder models like InternVL3-9B (76.87) and Janus-Pro-7B (74.83). Open-vocabulary detectors perform slightly lower but remain competitive, with Grounding DINO 1.5 Pro at 72.48 and OWLv2 at 72.02. Most models fall between 60 and 77 F1, highlighting benchmark difficulty. Llama-3.2-11B-Vision (33.33, YP = 0.00%) shows extreme class bias, while GLM-4v-9B favors precision (missed objects). Qwen3-VL-8B-Instruct also leads per-class F1 for Yes (78.51) and No (80.59) with balanced YP = 44.94%, whereas GPT-5 remains similarly balanced (Yes 76.92, No 79.35, $YP = 42.12\%$). Despite potential data overlap, the 79.55 F1 ceiling shows LVLMs still struggle with incongruous cases.

Model Architecture Comparison: Vision-encoder-based LVLMs dominate overall, with Qwen3-VL-8B-Instruct (79.55 F1), InternVL3-9B (76.87), and

Model	**Overall**				**Label Yes**			**Label No**		
	Pre.	**Rec.**	**F1**	**YP (%)**	**Pre.**	**Rec.**	**F1**	**Pre.**	**Rec.**	**F1**
Closed-source										
GPT-5-2025-08-07 [53]	79.50	78.75	**78.61**	42.12	84.14	70.88	**76.92**	71.84	88.62	**79.35**
Vision-encoder-based										
Llama-3.2-11B-Vision [13]	25.00	50.00	33.33	0.00	0.00	0.00	0.00	50.00	100.00	66.67
VILA1.5-13B [39]	65.19	62.40	60.41	28.95	71.44	41.35	51.86	58.92	83.45	68.96
GLM-4v-9B [23]	71.18	64.92	61.99	23.32	82.41	38.25	51.61	59.94	91.60	72.35
Phi-3.5-Vision-Instruct [1]	68.69	68.06	67.79	40.86	72.12	58.92	64.85	65.27	77.20	70.73
LLaVA-v1.6-Vicuna-13B [42]	75.29	74.56	74.37	56.94	71.76	81.50	76.19	78.82	67.62	72.55
Janus-Pro-7B [10]	76.60	75.22	74.83	56.42	73.30	81.65	76.71	79.90	68.80	72.95
InternVL3-9B [85]	77.33	76.95	76.87	44.60	80.27	71.55	75.60	74.39	82.35	78.13
Qwen3-VL-8B-Instruct [3, 4]	79.93	79.61	**79.55**	44.94	82.96	74.55	**78.51**	76.91	84.68	**80.59**
Vision-encoder-free										
EVE-7B-HD-v1.0 [18]	61.02	56.42	51.59	76.53	54.82	82.95	65.27	67.22	29.90	37.90
Emu3-Chat [68]	67.74	65.79	**64.78**	33.41	73.58	49.20	58.90	61.91	82.38	**70.67**
Open-vocabulary Detection										
OWLv2 [50]	73.02	72.25	72.02	40.85	77.23	63.10	69.46	68.81	81.40	**74.58**
Grounding DINO 1.5 Pro [57]	77.02	73.40	**72.48**	68.30	67.13	91.70	**77.51**	86.91	55.10	67.44

Table 3. **Main Experimental Results on ORIC.** Performance is broken down by model category and label type (Yes/No). We report macro precision (Prec.), recall (Rec.), F1 score, and the proportion of "yes" predictions (YP). Results for LVLMs are averaged over four prompts, while detection models use a single prompt. Full metric definitions are in Appendix A.4.

Janus-Pro-7B (74.83) notably outperforming encoder-free models, whose best, Emu3-Chat, reaches 64.78. The gap stems from ViT-style encoders providing structured visual features for fine-grained perception, whereas encoder-free models using raw pixels remain fragile in complex scenes. Among closed-source systems, GPT-5 (78.61) trails Qwen3-VL-8B-Instruct by only 0.94 points, showing open-source LVLMs can match or surpass proprietary ones. Open-vocabulary detectors like Grounding DINO 1.5 Pro (72.48) and OWLv2 (72.02) lag further, as their region–text alignment lacks holistic reasoning and explicit modeling of object absence, leading to more hallucinations in incongruous contexts.

Model	POPE-Bench			ORIC		
	Small	Medium	Large	Small	Medium	Large
Emu3-Chat	68.22	80.97	94.19	38.73	56.61	71.99
GPT-5-2025-08-07	78.24	88.48	94.30	67.85	71.69	84.34
InternVL3-9B	82.29	90.43	96.34	63.63	77.61	86.45
Qwen3-VL-8B-Instruct	79.96	89.71	96.40	69.96	77.67	85.24

Table 4. **Recall by Object Size on POPE vs. ORIC.** We report the recall for questions labeled "yes" across small, medium, and large objects in both the POPE and ORIC datasets for three LVLMs, illustrating how object scale affects model performance.

Influence of Incongruous Context (Class-Wise): Models exhibit distinct biases in incongruous contexts. Qwen3-VL-8B-Instruct and InternVL3-9B maintain balanced performance but lean conservative on "yes" predictions (YP $\approx$ 45%), yielding higher "no" F1 scores of 80.59 and 78.13. Their high "no" recall (84.68, 82.35) and lower "yes" recall suggest a preference for rejecting uncertainty over hallucinating presence. GLM-4v-9B and VILA1.5-13B show the opposite trend, underdetecting valid objects, while LLaVA-1.6-Vicuna-13B maintains a more even trade-off. Among detectors, Grounding DINO 1.5 Pro favors "yes" (recall $=$ 91.70, "no" recall $=$ 55.10), whereas OWLv2 is more balanced with the best "no" F1 (74.58). Overall, vision-encoder LVLMs handle contextual incongruity best, though a shared "yes"-conservatism bias reduces hallucinations but limits true-positive sensitivity.

Performance Comparison Across Object Sizes: Using COCO tiers—small ($< 24^2$ pt^2), medium (24^2–96^2 pt^2), and large ($\geq 96^2$ pt^2)—we compare 1,000 "yes"-labeled questions for POPE and ORIC-Bench. As shown in Table 4, all four models show lower recall on ORIC-Bench across sizes. Emu3-Chat drops most on small objects ($68.22 \rightarrow 38.73, -29.49$), while GPT-5 is comparatively stable on large ones ($94.30 \rightarrow 84.34, -9.96$). The large–small gap widens under incongruity for Emu3-Chat ($25.97 \rightarrow 33.26$) and InternVL3-9B ($14.05 \rightarrow 22.82$), remains roughly unchanged for GPT-5 ($16.06 \rightarrow 16.49$), and slightly narrows for Qwen3-VL-8B-Instruct ($16.44 \rightarrow 15.28$). Thus, while large objects remain easier, the consistent drop across all sizes shows that contextual incongruity, rather than scale, is the main source of uncertainty and performance drop.

Method	Overall				Label Yes			Label No		
	Precision	Recall	F1	YP (%)	Precision	Recall	F1	Precision	Recall	F1
(a) Standard ORIC-Bench Evaluation										
w 0-shot CoT	78.69	78.50	78.46	46.23	80.85	74.72	77.64	76.53	82.28	79.28
w/o 0-shot CoT	79.93	79.61	79.55	44.94	82.96	74.55	78.51	76.91	84.68	80.59
Visual-RFT	**83.55**	**82.88**	**82.79**	43.05	**88.21**	**75.92**	**81.59**	**78.88**	**89.83**	**83.99**
(b) Human-Labeled Ground Truth on ORIC-Bench										
w/o 0-shot CoT	78.70	78.63	78.63	47.14	79.73	76.52	78.08	77.69	80.75	79.17
Visual-RFT	**84.03**	**83.64**	**83.62**	44.72	**87.36**	**78.54**	**82.71**	**80.70**	**88.75**	**84.53**

Table 5. **Visual-RFT and Human-Referenced Results on ORIC-Bench.** (a) Standard evaluation comparing models with and without 0-shot CoT; (b) comparison against human-labeled ground truth. We report macro precision, recall, F1, and the proportion of "yes" predictions (YP). We find that visual-RFT produces outputs that better align with human thinking.

5. ORIC-driven Uncertainty Mitigation

Models trained on conventional data degrade on ORIC-Bench (macro-F1 79.55; Table 3). To mitigate these uncertainty-driven errors, we adopt Visual-RFT [44], which uses verifiable rewards to enforce evidence-grounded reasoning. We choose Visual-RFT over supervised fine-tuning because it is more data-efficient, more robust in few-sample regimes, and matches our ORIC setting, where rewards are naturally verifiable under the incongruous context.

We follow Visual-RFT [44], applying Group Relative Policy Optimization (GRPO) [59] to vision–language binary recognition with verifiable rewards. GRPO removes the PPO-style critic and compares candidates sampled from the same prompt, directly optimizing *relative* quality. Given a question q, we sample a group of G candidate responses $\{o_1, \dots, o_G\} \sim \pi_{\theta_{\text{old}}}(\cdot \mid q)$. Each sample receives two automatically checkable binary rewards: $r_{\text{acc}} \in \{0, 1\}$ for answer correctness and $r_{\text{fmt}} \in \{0, 1\}$ for format compliance (e.g., `<REASONING>...<\REASONING>` `<SOLUTION>...<\SOLUTION>`). Then, we define the per-sample reward as $r_i = r_{\text{acc},i} + r_{\text{fmt},i}$. Let $\{r_j\}_{j=1}^{G}$ denote the rewards of all candidates in the group. Since raw rewards may vary in scale across samples, we normalize them within each group (z-score) with a small constant ε:

$$\hat{r}_i = \frac{r_i - \text{mean}(\{r_j\}_{j=1}^{G})}{\text{std}(\{r_j\}_{j=1}^{G}) + \varepsilon}. \tag{5}$$

As rewards are one-step, token-level advantages are constant within a sample: $\hat{A}_{i,t} = \hat{r}_i, \ \forall t$. With the per-token ratio $\rho_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} \mid q, o_{i,<t})}$ GRPO maximizes the clipped, KL-regularized objective:

$$J_{\text{GRPO}}(\theta) = \mathbb{E}_q\Bigg[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\min\Big(\rho_{i,t}(\theta), \text{clip}\big(\rho_{i,t}(\theta), 1-\epsilon, 1+\epsilon\big)\Big)\hat{A}_{i,t}\Bigg] - \beta\, D_{\text{KL}}\big(\pi_\theta(\cdot|q)\,\big\|\,\pi_{\text{ref}}(\cdot|q)\big) \tag{6}$$

where ϵ is the clipping parameter and β controls a KL penalty to a frozen reference policy π_{ref}. In practice, we minimize $L_{\text{GRPO}} = -J_{\text{GRPO}}$. We adopt an R1-style, tag-constrained prompt to elicit explicit reasoning and a verifiable "yes" or "no" answer.

6. Uncertainty Mitigation Experiments and Analysis

6.1. Experimental Setup

To mitigate uncertainty-driven misjudgment and to strengthen evidence-grounded reasoning through verifiable reward optimization, we employ Visual-RFT. Specifically, we fine-tune Qwen3-VL-8B-Instruct [3, 4] on 600 ORIC-style binary questions (300 "yes"-label and 300 "no"-label questions) generated from the COCO-2014 training split, while ORIC-Bench uses disjoint validation images. We perform full-parameter Visual-RFT for 15 epochs with a group size $G = 8$ on 4×NVIDIA H100 GPUs using an R1-style tag-constrained prompt, which elicits explicit step-by-step reasoning and enforces verifiable yes/no outputs. Full hyper-parameters and prompts are provided in Appendix C. This setup enables reward signals based on reasoning correctness rather than label matching alone, reducing overreliance on uncertainty-driven errors. Inference follows the standard ORIC-Bench protocol, averaging predictions over four prompt variants.

Our baselines include the base model without 0-shot Chain-of-Thought (CoT) [69] and a 0-shot CoT variant using the prompt shown in Appendix Fig. 12. We further assess how Visual-RFT shifts predictions toward human-like behavior using a small human-labeled subset of ORIC-Bench, and additionally report results on HallusionBench and AMBER to show that its benefits generalize beyond ORIC-style data.

6.2. Results and Analysis on ORIC-Bench

Standard ORIC-Bench Evaluation. Table 5(a) shows that Visual-RFT consistently improves Qwen3-VL-8B-

Instruct, with or without 0-shot CoT. Macro F1 rises to **82.79** (from 78.46/79.55), with clear F1 and recall gains for both "yes" (78.51 → 81.59; 74.55 → 75.92) and "no" (80.59 → 83.99; 84.68 → 89.83) questions. The slight drop in YP further suggests fewer spurious positives. Overall, training on ORIC-style data with Visual-RFT mitigates uncertainty-driven errors and strengthens LVLM performance under contextual incongruity.

Comparison with Human Preferences. To evaluate alignment with human reasoning, we annotate 200 ORIC-Bench questions (100 "yes"-label and 100 "no"-label questions) as the alternative ground truth. As shown in Tab. 5(b), Visual-RFT improves macro F1 from 78.63 to **83.62**, indicating closer agreement with human judgments under ambiguous contexts. F1 increases for both labels (78.08→82.71 for "yes" and 79.17→84.53 for "no"), with particularly strong gains on "no" questions, where recall rises from 80.75 to 88.75. This shows that training on ORIC-style data with Visual-RFT reduces missed negatives and better aligns model predictions with human patterns.

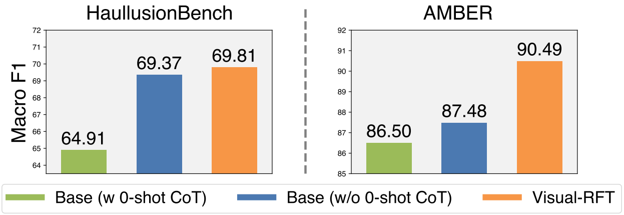


Figure 5. **Performance across Benchmarks.** Macro F1 on HallusionBench and AMBER under three settings: with/without zero-shot CoT and Visual-RFT fine-tuning.

Cross-benchmark Evaluation. We further assess generalization on HallusionBench and AMBER (Fig. 5). Visual-RFT improves robustness on both benchmarks. On HallusionBench, which contains visual illusions and abstract figures, performance remains stable (69.37 → 69.81), showing that RFT does not overfit to ORIC-style data. On AMBER, which requires compositional reasoning over existence, attributes, and relations, the gains are substantial (87.48 → **90.49**). These results show that training on ORIC-style data with Visual-RFT improves generalization beyond ORIC-Bench and enhances robustness to both visual and semantic distribution shifts.

7. Related Work

Large Vision-Language Models: Recent advances in large vision-language models (LVLMs) have greatly enhanced text-image processing for visual understanding [1, 28, 67, 84]. These models fall into two categories: vision-encoder-based approaches [2, 3, 23, 34, 41], which use pretrained visual encoders like Vision Transformer (ViT) [19], and vision-encoder-free methods [6, 18, 68], which tokenize image patches for joint text-image processing. LVLMs are widely used in tasks such as image captioning [16], visual question answering [60], robotics [22, 27, 51], and embodied AI [73, 82]. Despite progress, they still struggle with fine-grained perception [55].

Benchmarking Large Vision-Language Models: As LVLMs evolve, benchmarking is crucial for guiding their development [8, 37, 38]. Many benchmarks focus on fine-grained perception, including counting, relations, attributes, and reasoning [9, 21, 33, 43, 47, 71, 75, 78], or on commonsense and knowledge-intensive tasks [7, 79]. Others target object hallucination and recognition [26, 35, 58, 66], with some emphasizing textual influences or visual semantics [24, 64, 65]. However, these benchmarks largely preserve object–context compatibility and rarely test recognition under incongruous contexts. ORIC-Bench fills this gap by explicitly evaluating object existence in such settings.

Reinforcement Learning: Recent RL-based post-training methods directly optimize verifiable reasoning outcomes. OpenAI o1 and DeepSeek-R1 demonstrate that large-scale RL and GRPO can strengthen chain-of-thought reasoning in both closed- and open-source models [25, 52], while subsequent work improves GRPO stability and efficiency [11, 14, 46, 76]. In multimodal settings, RL reduces hallucinations through fine-grained visual feedback, as in RLHF-V [77], and enables efficient visual reinforcement tuning via Visual-RFT [45]. Building on this line of work, we attach verifiable rewards directly to object existence under contextual incongruity using a Visual-RFT–style GRPO scheme that enforces evidence-grounded decisions.

8. Conclusion and Limitations

This paper presents the first systematic study of how contextual incongruity, viewed through the lens of uncertainty, affects LVLM object recognition, showing that state-of-the-art models still struggle in such settings. To investigate this gap, we introduce ORIC, a framework built with LLM-guided and CLIP-guided sampling to generate challenging, context-aware recognition tasks for both evaluation and training. Experiments across 20 models reveal that handling incongruous contexts remains a substantial weakness. We further fine-tune the LVLM with reinforcement learning under the Visual-RFT framework using ORIC-style data, which improves robustness to incongruity, boosts both in-distribution and out-of-distribution performance, and yields outputs more aligned with human reasoning. While our study establishes a foundation, it is limited to a single dataset. Future work should explore more diverse contexts and develop stronger methods for reliable recognition under incongruity.

References

[1] Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024. 6, 8, 2

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022. 8

[3] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 1(2):3, 2023. 2, 6, 7, 8

[4] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 2, 6, 7

[5] Rohan Bavishi, Erich Elsen, Curtis Hawthorne, Maxwell Nye, Augustus Odena, Arushi Somani, and Sağnak Taşırlar. Introducing our multimodal models, 2023. 2

[6] Rohan Bavishi, Erich Elsen, Curtis Hawthorne, Maxwell Nye, Augustus Odena, Arushi Somani, and Sagnak Tasırlar. Introducing our multimodal models, 2023. 8

[7] Nitzan Bitton-Guetta, Aviv Slobodkin, Aviya Maimon, Eliya Habba, Royi Rassin, Yonatan Bitton, Idan Szpektor, Amir Globerson, and Yuval Elovici. Visual riddles: a commonsense and world knowledge challenge for large vision and language models. *arXiv preprint arXiv:2407.19474*, 2024. 8

[8] Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? *arXiv preprint arXiv:2403.20330*, 2024. 8

[9] Xuweiyi Chen, Ziqiao Ma, Xuejun Zhang, Sihan Xu, Shengyi Qian, Jianing Yang, David Fouhey, and Joyce Chai. Multi-object hallucination in vision language models. *Advances in Neural Information Processing Systems*, 37:44393–44418, 2024. 8

[10] Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025. 3, 6, 2

[11] Xiwen Chen, Wenhui Zhu, Peijie Qiu, Xuanzhao Dong, Hao Wang, Haiyu Wu, Huayu Li, Aristeidis Sotiras, Yalin Wang, and Abolfazl Razi. Dra-grpo: Exploring diversity-aware reward adjustment for r1-zero-like training of large language models. *arXiv preprint arXiv:2505.09655*, 2025. 8

[12] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198, 2024. 1

[13] Jianfeng Chi, Ujjwal Karn, Hongyuan Zhan, Eric Smith, Javier Rando, Yiming Zhang, Kate Plawiak, Zacharie Delpierre Coudert, Kartikeya Upasani, and Mahesh Pasupuleti. Llama guard 3 vision: Safeguarding human-ai image understanding conversations, 2024. 6, 2

[14] Muzhi Dai, Shixuan Liu, and Qingyi Si. Grpo-λ: An efficient and stabilized variant of grpo for long-chain reasoning in llms. *arXiv preprint arXiv:2505.18086*, 2025. 8

[15] Wenliang Dai, Zihan Liu, Ziwei Ji, Dan Su, and Pascale Fung. Plausible may not be faithful: Probing object hallucination in vision-language pre-training. *arXiv preprint arXiv:2210.07688*, 2022. 1

[16] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *arXiv preprint arXiv:2305.06500*, 2023. 1, 8

[17] Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *arXiv preprint arXiv:2409.17146*, 2024. 2

[18] Haiwen Diao, Yufeng Cui, Xiaotong Li, Yueze Wang, Huchuan Lu, and Xinlong Wang. Unveiling encoder-free vision-language models. *arXiv preprint arXiv:2406.11832*, 2024. 6, 8, 2

[19] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 8

[20] Jiafei Duan, Wilbert Pumacay, Nishanth Kumar, Yi Ru Wang, Shulin Tian, Wentao Yuan, Ranjay Krishna, Dieter Fox, Ajay Mandlekar, and Yijie Guo. Aha: A vision-language-model for detecting and reasoning over failures in robotic manipulation. *arXiv preprint arXiv:2410.00371*, 2024. 1

[21] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023. 8

[22] Jensen Gao, Bidipta Sarkar, Fei Xia, Ted Xiao, Jiajun Wu, Brian Ichter, Anirudha Majumdar, and Dorsa Sadigh. Physically grounded vision-language models for robotic manipulation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 12462–12469. IEEE, 2024. 1, 8

[23] Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi

Li, Lei Zhao, Lindong Wu, Lucen Zhong, Mingdao Liu, Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao, Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiao Xia, Xiaohan Zhang, Xiaotao Gu, Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yifan An, Yifan Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang, Zhen Yang, Zhengxiao Du, Zhenyu Hou, and Zihan Wang. Chatglm: A family of large language models from glm-130b to glm-4 all tools, 2024. 6, 8, 2

[24] Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, et al. Hallusionbench: an advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14375–14385, 2024. 2, 8

[25] Dong Guo et al. Deepseek-R1: Incentivizing reasoning capability in language models via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. 8

[26] Hongyu Hu, Jiyuan Zhang, Minyi Zhao, and Zhenbang Sun. Ciem: Contrastive instruction evaluation method for better instruction tuning. *arXiv preprint arXiv:2309.02301*, 2023. 8

[27] Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. Voxposer: Composable 3d value maps for robotic manipulation with language models. *arXiv preprint arXiv:2307.05973*, 2023. 8

[28] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 3, 8

[29] Kiyoon Jeong, Woojun Lee, Woongchan Nam, Minjeong Ma, and Pilsung Kang. Technical report of nice challenge at cvpr 2024: caption re-ranking evaluation using ensembled clip and consensus scores. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7366–7372, 2024. 5

[30] Olivier R Joubert, Denis Fize, Guillaume A Rousselet, and Michele Fabre-Thorpe. Early interference of context congruence on object processing in rapid visual categorization of natural scenes. *Journal of Vision*, 8(13):11–11, 2008. 1

[31] Adam Tauman Kalai, Ofir Nachum, Santosh S Vempala, and Edwin Zhang. Why language models hallucinate. *arXiv preprint arXiv:2509.04664*, 2025. 1, 2

[32] Martha Lewis, Nihal V Nayak, Peilin Yu, Qinan Yu, Jack Merullo, Stephen H Bach, and Ellie Pavlick. Does clip bind concepts? probing compositionality in large image models. *arXiv preprint arXiv:2212.10537*, 2022. 5

[33] Bo Li, Peiyuan Zhang, Jingkang Yang, Yuanhan Zhang, Fanyi Pu, and Ziwei Liu. Otterhd: A high-resolution multimodality model. *arXiv preprint arXiv:2311.04219*, 2023. 8

[34] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022. 8

[35] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023. 2, 3, 8

[36] Zongxia Li, Xiyang Wu, Hongyang Du, Huy Nghiem, and Guangyao Shi. Benchmark evaluations, applications, and challenges of large vision language models: A survey. *arXiv preprint arXiv:2501.02189*, 2025. 1

[37] Paul Pu Liang, Akshay Goindani, Talha Chafekar, Leena Mathur, Haofei Yu, Ruslan Salakhutdinov, and Louis-Philippe Morency. Hemm: Holistic evaluation of multimodal foundation models. *arXiv preprint arXiv:2407.03418*, 2024. 8

[38] Zijing Liang, Yanjie Xu, Yifan Hong, Penghui Shang, Qi Wang, Qiang Fu, and Ke Liu. A survey of multimodel large language models. In *Proceedings of the 3rd International Conference on Computer, Artificial Intelligence and Control Engineering*, pages 405–409, 2024. 8

[39] Ji Lin, Hongxu Yin, Wei Ping, Yao Lu, Pavlo Molchanov, Andrew Tao, Huizi Mao, Jan Kautz, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models, 2023. 6, 2

[40] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 5

[41] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024. 8

[42] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 2024. 6, 2

[43] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pages 216–233. Springer, 2024. 5, 8

[44] Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi Wang. Visual-rft: Visual reinforcement fine-tuning. *arXiv preprint arXiv:2503.01785*, 2025. 2, 7

[45] Ziyu Liu, Zeyi Sun, Yuhang Zang, et al. Visual-RFT: Visual reinforcement fine-tuning. In *ICCV*, 2025. 8

[46] Zichen Liu et al. Unifying the grpo frameworks with learnable token preference. *arXiv preprint arXiv:2510.06870*, 2025. 8

[47] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521, 2022. 8

[48] Andrés Marafioti, Orr Zohar, Miquel Farré, Merve Noyan, Elie Bakouch, Pedro Cuenca, Cyril Zakka, Loubna Ben

Allal, Anton Lozhkov, Nouamane Tazi, Vaibhav Srivastav, Joshua Lochner, Hugo Larcher, Mathieu Morlon, Lewis Tunstall, Leandro von Werra, and Thomas Wolf. Smolvlm: Redefining small and efficient multimodal models. *arXiv preprint arXiv:2504.05299*, 2025. 2

[49] Dimity Miller, Niko Sünderhauf, Alex Kenna, and Keita Mason. Open-set recognition in the age of vision-language models. In *European Conference on Computer Vision*, pages 1–18. Springer, 2024. 1

[50] Matthias Minderer, Alexey Gritsenko, and Neil Houlsby. Scaling open-vocabulary object detection. *Advances in Neural Information Processing Systems*, 36, 2024. 5, 6, 2

[51] Soroush Nasiriany, Fei Xia, Wenhao Yu, Ted Xiao, Jacky Liang, Ishita Dasgupta, Annie Xie, Danny Driess, Ayzaan Wahid, Zhuo Xu, et al. Pivot: Iterative visual prompting elicits actionable knowledge for vlms. *arXiv preprint arXiv:2402.07872*, 2024. 8

[52] OpenAI. OpenAI o1 system card. *arXiv preprint arXiv:2412.16720*, 2024. 8

[53] OpenAI. Introducing gpt-5, 2025. 1, 6, 2

[54] Marius V Peelen, Eva Berlot, Floris P de Lange, and Michele Fabre-Thorpe. Predictive processing of scenes and objects. *Nature Reviews Psychology*, 3(1):13–26, 2024. 1

[55] Wujian Peng, Sicheng Xie, Zuyao You, Shiyi Lan, and Zuxuan Wu. Synthesize diagnose and optimize: Towards fine-grained vision-language understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13279–13288, 2024. 8

[56] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2, 3

[57] Tianhe Ren, Qing Jiang, Shilong Liu, Zhaoyang Zeng, Wenlong Liu, Han Gao, Hongjie Huang, Zhengyu Ma, Xiaoke Jiang, Yihao Chen, et al. Grounding dino 1.5: Advance the" edge" of open-set object detection. *arXiv preprint arXiv:2405.10300*, 2024. 5, 6, 2

[58] Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object hallucination in image captioning. *arXiv preprint arXiv:1809.02156*, 2018. 1, 8

[59] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Yiqun Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. 7

[60] Haoyu Song, Li Dong, Wei-Nan Zhang, Ting Liu, and Furu Wei. Clip models are few-shot learners: Empirical studies on vqa and visual entailment. *arXiv preprint arXiv:2203.07190*, 2022. 1, 8

[61] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023. 1

[62] Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024. 2

[63] Kimi Team, Angang Du, Bohong Yin, Bowei Xing, Bowen Qu, Bowen Wang, Cheng Chen, Chenlin Zhang, Chenzhuang Du, Chu Wei, et al. Kimi-vl technical report. *arXiv preprint arXiv:2504.07491*, 2025. 2

[64] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9568–9578, 2024. 8

[65] Guangzhi Wang, Yixiao Ge, Xiaohan Ding, Mohan Kankanhalli, and Ying Shan. What makes for good visual tokenizers for large language models? *arXiv preprint arXiv:2305.12223*, 2023. 8

[66] Junyang Wang, Yuhang Wang, Guohai Xu, Jing Zhang, Yukai Gu, Haitao Jia, Ming Yan, Ji Zhang, and Jitao Sang. An llm-free multi-dimensional benchmark for mllms hallucination evaluation. *arXiv preprint arXiv:2311.07397*, 2023. 2, 8

[67] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 8

[68] Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024. 6, 8, 2

[69] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022. 7

[70] Miles Wischnewski, Marius V Peelen, Michele Fabre-Thorpe, and Michele Fabre-Thorpe. Causal neural mechanisms of context-based object recognition. *Elife*, 10:e69736, 2021. 1

[71] Shujin Wu, Yi R Fung, Sha Li, Yixin Wan, Kai-Wei Chang, and Heng Ji. Macaroon: Training vision-language models to be your engaged partners. *arXiv preprint arXiv:2406.14137*, 2024. 8

[72] Le Xue, Manli Shu, Anas Awadalla, Jun Wang, An Yan, Senthil Purushwalkam, Honglu Zhou, Viraj Prabhu, Yutong Dai, Michael S Ryoo, et al. xgen-mm (blip-3): A family of open large multimodal models. *arXiv preprint arXiv:2408.08872*, 2024. 2

[73] Jingkang Yang, Yuhao Dong, Shuai Liu, Bo Li, Ziyue Wang, Haoran Tan, Chencheng Jiang, Jiamu Kang, Yuanhan Zhang, Kaiyang Zhou, et al. Octopus: Embodied vision-language programmer from environmental feedback. In *European Conference on Computer Vision*, pages 20–38. Springer, 2024. 1, 8

[74] Suorong Yang, Peng Ye, Wanli Ouyang, Dongzhan Zhou, and Furao Shen. A clip-powered framework for robust and generalizable data selection. *arXiv preprint arXiv:2410.11215*, 2024. 5

[75] Kaining Ying, Fanqing Meng, Jin Wang, Zhiqian Li, Han Lin, Yue Yang, Hao Zhang, Wenbo Zhang, Yuqi Lin, Shuo Liu, et al. Mmt-bench: A comprehensive multimodal benchmark for evaluating large vision-language models towards multitask agi. *arXiv preprint arXiv:2404.16006*, 2024. 8

[76] Qiying Yu et al. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025. 8

[77] Tianyu Yu et al. RLHF-V: Towards trustworthy MLLMs via behavior alignment from fine-grained correctional human feedback. In *CVPR*, 2024. 8

[78] Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*, 2023. 8

[79] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9556–9567, 2024. 8

[80] Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? *arXiv preprint arXiv:2210.01936*, 2022. 5

[81] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023. 1

[82] Yuexiang Zhai, Hao Bai, Zipeng Lin, Jiayi Pan, Shengbang Tong, Yifei Zhou, Alane Suhr, Saining Xie, Yann LeCun, Yi Ma, et al. Fine-tuning large vision-language models as decision-making agents via reinforcement learning. *arXiv preprint arXiv:2405.10292*, 2024. 8

[83] Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 1

[84] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 8

[85] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025. 3, 6, 2

ORIC: Benchmarking Object Recognition under Contextual Incongruity in Large Vision-Language Models

Supplementary Material

A. ORIC Method, Analysis, and ORIC-Bench Evaluation Metrics

A.1. LLM-Guided Sampling Method (Positive Question Construction)

Algorithm 1 Positive Question Construction

```
Require: Image $I$, objects $\mathcal{O} = \{(n_i, B_{ij})\}$, integer $k$
Ensure: Positive question $Q$
for $i = 1$ to $N$ do
    $A_i \leftarrow$ area($\bigcup_j B_{ij}$)
end for
Sort $\mathcal{O}$ by $A_i$ (descend.)
$\mathcal{O}_{\text{ROI}} \leftarrow$ bottom 50%, $\mathcal{O}_{\text{nonROI}} \leftarrow$ top 50%    ▷ Note: Objects exactly at the 50% boundary are classified as non ROI.
$\mathcal{C} \leftarrow \emptyset$
for $o \in \mathcal{O}_{\text{ROI}}$ do
    if LLM says "no" for $o$ given $\mathcal{O}_{\text{nonROI}}$ then
        $\mathcal{C} \leftarrow \mathcal{C} \cup \{o\}$
    end if
end for
Randomly pick $k$ objects from $\mathcal{C}$ as $Q$ return $Q$
```

Figure 6 presents the prompt used in LLM-guided rejection sampling for constructing positive questions in the ORIC. Specifically, {`background_objects`} serves as a placeholder for all non-ROI objects. For example, if there are three non-ROI objects, they could be represented as `["car", "person", "bottle"]`. Meanwhile, {`target_object`} represents a placeholder for a specific ROI object, such as `"vase"`.

LLM-Guided Rejection Sampling

Given the following background objects: {`background_objects`}, can you determine whether the following target object {`target_object`} is present in the image without relying on textual priors, common-sense knowledge, or general assumptions about object co-occurrences?
Please respond with yes or no.

Figure 6. **Prompt for LLM-guided rejection sampling.** {`background_objects`} is a placeholder for all non-ROI objects, and {`target_object`} denotes a specific ROI object.

A.2. CLIP-Guided Sampling Method (Negative Question Construction)

Algorithm 2 Negative Question Construction

```
Require: Query image $I_q$, candidate images $\{I_1, \ldots, I_n\}$,
    non-existent objects $\mathcal{O}_{\text{non}} = \{n_i\}_{i=1}^{M}$, integer $k$
Ensure: Negative question $Q$
Select the most similar image:
    $I' = \arg\min_{I_i \in \mathcal{I}} \left(1 - \frac{\mathbf{e}_q \cdot \mathbf{e}_i}{\|\mathbf{e}_q\|\|\mathbf{e}_i\|}\right)$
for $i = 1$ to $M$ do
    Construct text: $T_i \leftarrow$ "an image contains { $n_i$}"
    Compute CLIP score: $s_i \leftarrow$ CLIPScore($I', T_i$)
end for
Sort $\{n_i\}$ by $s_i$ (descending)
Select top $k$ objects: $\mathcal{S} \leftarrow \{n_{i_1}, \ldots, n_{i_k}\}$
Construct $Q$ using $\mathcal{S}$ return $Q$
```

A.3. Image Similarity Analysis via Minimum Distance

To further characterize the ORIC, we analyzed the visual relationships between positive and negative questions through image similarity measurements. Specifically, for each object class appearing in positive ("yes") questions, we computed its minimum visual distance to negative ("no") questions containing the same object class. Given an object o_i, let the set of positive images be $\mathcal{I}_i^+ = \{I_{i,1}^+, \ldots, I_{i,m}^+\}$ and the set of negative images be $\mathcal{I}_i^- = \{I_{i,1}^-, \ldots, I_{i,n}^-\}$. We extracted visual feature vectors using a ViT encoder and computed pairwise cosine distances as follows:

$$D(I_{i,k}^+, I_{i,l}^-) = 1 - \frac{e(I_{i,k}^+) \cdot e(I_{i,l}^-)}{\|e(I_{i,k}^+)\| \, \|e(I_{i,l}^-)\|} \tag{7}$$

where $e(\cdot) = \text{ViT}(\cdot)$ denotes the ViT feature extractor. The minimum distance between positive and negative sets is defined as $D_{\min} = \min_{k,l} D(I_{i,k}^+, I_{i,l}^-)$. To ensure thorough evaluation, we calculated these minimum distances using three widely used vision encoders commonly employed in encoder-based LVLMs: CLIP-ViT-BigG-P14, SigLIP-SO400M-P14-384 [81], and EVA02-CLIP-BigE-P14 [61]. These analyses highlight the distinctiveness of ORIC in capturing contextually challenging object recognition scenarios compared to existing benchmarks. In Tab. 6, questions generated from ORIC shows consistently smaller minimum cosine distances between ”yes” and ”no” samples than POPE across all three vision encoders. This suggests greater visual similarity between positive and negative examples, making object recognition more challenging and realistic.

Vision Encoder	POPE	ORIC
CLIP-ViT-BigG-P14	0.37	**0.14**
SigLIP-SO400M-P14-384	0.28	**0.11**
EVA02-CLIP-BigE-P14	0.40	**0.13**

Table 6. **Comparison of Minimum Cosine Distances.** This table compares the minimum cosine distances between positive and negative questions across three vision encoders. A smaller distance indicates greater semantic similarity between images, meaning "yes" and "no" questions are linked to finer image details and higher representational clutter, making object recognition more challenging and realistic.

A.4. Evaluation Metric Formulas

For a binary classification problem with labels *yes* and *no*, we define the following terms:

- **TP** (True Positive): Number of samples correctly predicted as *yes* (Ground Truth: *yes*).
- **TN** (True Negative): Number of samples correctly predicted as *no* (Ground Truth: *no*).
- **FP** (False Positive): Number of samples incorrectly predicted as *yes* (Ground Truth: *no*).
- **FN** (False Negative): Number of samples incorrectly predicted as *no* (Ground Truth: *yes*).

The performance metrics include accuracy, the proportion of *yes* predictions, macro precision, recall, and F1 score. These are defined as follows:

Class-wise Metrics:

$$\text{Precision}_{\text{yes}} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{8}$$

$$\text{Recall}_{\text{yes}} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{9}$$

$$F1_{\text{yes}} = 2 \times \frac{\text{Precision}_{\text{yes}} \times \text{Recall}_{\text{yes}}}{\text{Precision}_{\text{yes}} + \text{Recall}_{\text{yes}}} \tag{10}$$

$$\text{Precision}_{\text{no}} = \frac{\text{TN}}{\text{TN} + \text{FN}} \tag{11}$$

$$\text{Recall}_{\text{no}} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{12}$$

$$F1_{\text{no}} = 2 \times \frac{\text{Precision}_{\text{no}} \times \text{Recall}_{\text{no}}}{\text{Precision}_{\text{no}} + \text{Recall}_{\text{no}}} \tag{13}$$

Macro-averaged Metrics:

$$\text{Precision}_{\text{macro}} = \frac{\text{Precision}_{\text{yes}} + \text{Precision}_{\text{no}}}{2} \tag{14}$$

$$\text{Recall}_{\text{macro}} = \frac{\text{Recall}_{\text{yes}} + \text{Recall}_{\text{no}}}{2} = \text{Accuracy} \tag{15}$$

Since our experimental datasets are all balanced, the number of positive and negative samples is equal. In this case, $\text{Accuracy} = \text{Recall}_{\text{macro}}$ because accuracy measures the overall proportion of correctly classified samples, and macro recall, being the unweighted average of recall for both classes, reflects the same value.

$$F1_{\text{macro}} = \frac{F1_{\text{yes}} + F1_{\text{no}}}{2} \tag{16}$$

Proportion of Yes Predictions: The proportion of "yes" predictions (i.e., the percentage of all predictions that are classified as "yes") is given by:

$$\text{Yes Proportion} = \frac{\text{TP} + \text{FP}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{17}$$

B. ORIC-Bench Experiment and Analysis

B.1. Evaluated Models

We evaluate **18** widely used LVLMs spanning both encoder-based and encoder-free architectures. The encoder-based models include Qwen3-VL-8B-Instruct [3, 4], SmolVLM2-2.2B-Instruct [48], InternVL3-9B [85], Kimi-VL-A3B-Instruct [63], Janus-Pro-7B [10], Llama-3.2-11B-Vision [13], LLaVa-v1.6-7B [42], Phi-3.5-Vision-Instruct [1], Molmo-7B-D-0924 [17], GLM-4V-9B [23], Chameleon-7B [62], VILA-1.5-13B [39], and BLIP3 [72]. Encoder-free models include Fuyu-8B [5], EVE-7B-HD-v1.0 [18], Emu3-Chat [68], and the closed-source GPT-5 [53]. What's more, we benchmark against **2** open-vocabulary detection models: Grounding DINO 1.5 Pro [57] and OWLv2 [50].

B.2. Prompt Templates of Experiments

Large Vision-Language Models (LVLMs) Fig. 7 illustrates the prompt used for LVLMs in both the POPE and LOPE-3 benchmarks. An example of a specific question is: *"Is there a person in the image?"*.

LVLMs

<image>
Question: {question}
Please answer the question based on the given image.

Figure 7. **The Prompt of LVLMs.** The prompt of a binary classification task for LVLMs is used in all experiments, where {question} serves as a placeholder for a specific query and <image> is the placeholder for a specific image.

We use four distinct prompts in our experiments, detailed below:

- Is there {object} in the image?

- Does the image contain {object}?
- Have you noticed {object} in the image?
- Can you see {object} in the image?

The {object} is the placeholder for a detail object.

Grounding DINO 1.5 Pro Prompt: Figure 8 shows the prompt for Grounding DINO 1.5 Pro. For example, if an image contains four unique objects—sports ball, person, car, and traffic light—the corresponding prompt would be: *"sports ball.person.car.traffic light"*.

Grounding DINO 1.5 Pro

$\{object1\}.\{object_2\}.\cdots.\{object_n\}$

Figure 8. **The Prompt of Grounding DINO 1.5 Pro.** The prompt used for the binary classification task in all experiments with Grounding DINO 1.5 Pro follows a dot-separated notation to specify multiple objects. Placeholders $\{object1\}, \{object_2\}, \cdots \{object_n\}$ represent unique objects in the image, where n denotes the total number of distinct objects.

OWLv2 Prompt: Figure 9 shows the prompt for OWLv2. An example of a specific object is: *"an image of truck"*.

OWLv2

an image of {object}

Figure 9. **The Prompt of OWLv2.** The prompt of a binary classification task for OWLv2 used in all experiments, where {object} serves as a placeholder for a specific object.

Model	Random	Pos Only	Neg Only
DINO 1.5 Pro	95.50 / 85.50	91.60 (-3.90)	53.05 **(-32.45)**
GPT-5-2025-08-07	81.53 / 96.12	71.92 (-9.61)	84.45 (-11.67)
Emu3	67.25 / 97.30	48.75 **(-18.50)**	81.17 (-16.13)
InternVL3-9B	80.88 / 97.83	68.83 (-12.05)	81.75 (-16.08)
Qwen3-VL-8B-Instruct	82.95 / 97.15	74.28 (-8.67)	83.90 (-13.25)

Table 7. **Ablation study of ORIC-Bench.** The table evaluates three sampling setups: **Random:** A baseline using randomly selected positive and negative objects. **Pos Only:** Employs LLM-guided sampling for positives and random negatives. **Neg Only:** Uses CLIP-guided sampling for negatives and random positives. All values are reported as (yes-recall / no-recall), with parentheses indicating the performance drop relative to the Random baseline.

B.3. Supplementary Experiments and Analysis

B.3.1. ORIC-Bench Ablation Study:

We follow the ORIC-Bench experiment settings, averaging LVLM metrics over four prompts and using a default prompt for detection models. Tab. 7 shows that both LLM-guided and CLIP-guided sampling increase question difficulty across four LVLMs and Grounding DINO Pro 1.5. LLM-guided sampling reduces yes-recall across all models, with Emu3 experiencing the largest drop (-18.50). Meanwhile, CLIP-guided sampling significantly lowers no-recall, with the most notable decline observed in DINO 1.5 Pro (-32.45). These results suggest that both positive and negative question constructions introduce challenges, though their effects differ. Notably, no-recall declines more sharply in most models. This discrepancy arises because positive questions reference real objects, aiding recognition even in incongruous backgrounds, whereas negative questions involve absent objects, leading models to over-rely on background context and hallucinate in congruous settings.

B.3.2. Full Results of Comparison between POPE and ORIC

Tab. 8 presents a comparative analysis of POPE and ORIC-Bench across 19 LVLMs and 2 open-vocabulary detection models. Notably, the macro F1 scores of Llama-3.2-11B-Vision, Chameleon-7B, BLIP-3, and VILA1.5-3B in POPE are comparable to or even exceed those in ORIC-Bench. A potential explanation is that these models exhibit a high proportion of "yes" responses in both benchmarks, suggesting a tendency to answer affirmatively regardless of context. This behavior indicates limited object recognition capabilities, as their responses remain consistent across different evaluation settings. Furthermore, the macro precision and recall of other models in ORIC-Bench are significantly lower than in POPE, leading to a sharp decline in macro F1 scores. This suggests that ORIC-Bench presents a greater challenge for all tested LVLMs, highlighting their struggles with object recognition, particularly when considering contextual incongruity.

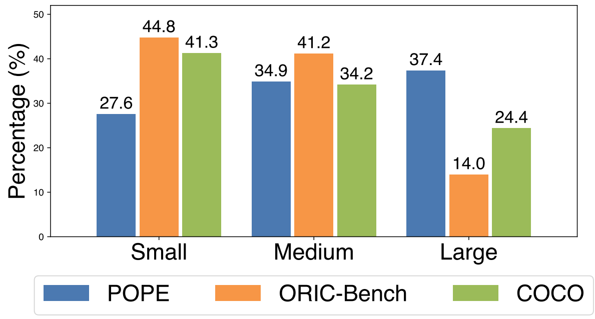


Figure 10. **Object Size Distribution across POPE, ORIC-Bench, and COCO.** Percentage distribution of small ($< 24 \times 24$ pt^2), medium (24×24–96×96 pt^2), and large ($\geq 96 \times 96$ pt^2) objects in the POPE, ORIC-Bench, and COCO datasets, highlighting ORIC-Bench's deliberate shift toward smaller and medium object scales.

Model	**POPE**				**ORIC-Bench**			
	Precision	**Recall**	**F1 Score**	**YP (%)**	**Precision**	**Recall**	**F1 Score**	**YP (%)**
Closed-source								
GPT-5-2025-08-07	89.06	88.60	**88.56**	44.62	79.50	78.75	**78.61**	42.12
Encoder-based								
Llama-3.2-11B-Vision	25.00	50.00	33.33	0.00	25.00	50.00	33.33	0.00
Chameleon-7B	47.08	50.01	33.95	99.29	59.75	50.10	34.08	99.28
BLIP-3	36.20	44.88	37.29	80.30	43.14	49.86	42.99	81.54
VILA1.5-13B	60.87	59.92	57.49	36.80	65.19	62.40	60.41	28.95
GLM-4v-9B	86.55	84.12	83.85	37.30	71.18	64.92	61.99	23.32
Phi-3.5-Vision-Instruct	86.76	86.28	86.23	44.35	68.69	68.06	67.79	40.86
InternLM-XComposer2.5-7B	84.72	83.16	82.98	39.84	73.32	70.35	69.33	33.77
SmolVLM2-2.2B-Instruct	87.57	86.89	86.83	43.56	72.87	71.44	70.95	38.01
Kimi-VL-A3B-Instruct	88.91	87.69	87.59	41.19	74.67	72.28	71.58	34.45
Molmo-7B-D-0924	83.76	81.45	81.03	61.42	78.92	73.74	71.95	69.34
LLaVA-v1.6-Vicuna-13B	88.24	88.14	88.13	51.39	75.29	74.56	74.37	56.94
Janus-Pro-7B	87.32	87.03	87.00	50.65	76.60	75.22	74.83	56.42
InternVL3-9B	88.8	88.69	**88.68**	47.96	77.33	76.95	76.87	44.60
Qwen3-VL-8B-Instruct	88.13	88.04	88.03	47.66	79.93	79.61	**79.55**	44.94
Encoder-free								
Fuyu-8B	68.39	53.47	40.48	95.70	44.83	50.16	34.16	99.29
EVE-7B-HD-v1.0	82.19	79.81	79.34	61.36	61.02	56.42	51.59	76.53
Emu3-Chat	87.43	86.72	**86.66**	43.25	67.74	65.79	**64.78**	33.41
Open-vocabulary Detection								
OWLv2	86.74	86.55	**86.53**	53.55	73.02	72.25	72.02	40.85
Grounding DINO 1.5 Pro	85.62	85.05	84.99	56.35	77.02	73.40	**72.48**	68.30

Table 8. **Full Model Performance Comparison: POPE vs. ORIC.** The table compares POPE and ORIC across various model categories: closed-source, encoder-based, encoder-free, and open-vocabulary detection models. Performance is evaluated using macro precision, recall, and F1 score. The yes proportion (YP (%)) indicates the percentage of ”yes” predictions. “Prec.” denotes precision, “Rec.” denotes recall, and “F1.” denotes the F1 score. All values are averaged across four prompts, except for detection models, which use a single prompt without averaging.

B.3.3. Comparison of Object Size Distribution between POPE, ORIC-Bench, and COCO:

Fig. 10 compares the proportions of small ($< 24 \times 24$ pt^2), medium (24×24–96×96 pt^2), and large ($\geq 96\times96$ pt^2) objects in POPE, ORIC-Bench, and COCO. In ORIC-Bench, small objects are the single largest category at 44.8%—yet they do not constitute a majority: medium objects follow closely at 41.2%, while large objects still make up a substantial 14.0%. Relative to POPE (27.6% small, 34.9% medium, 37.4% large) and COCO (41.3% small, 34.2% medium, 24.4% large), ORIC-Bench deliberately boosts the share of small and medium instances at the expense of large ones. This design amplifies the need for fine-grained recognition and scale-robust feature extraction in the face of context incongruity, while still retaining a substantial number of medium and large objects to ensure the benchmark is not solely focused on small instances and can assess model performance across the full spectrum of object scales.

C. Visual-RFT Experimental Details

C.1. Visual-RFT Training Hyper-parameters

Tab. 9 lists the full set of hyper-parameters used in our Visual-RFT training. We include all optimization, sampling, and generation settings to ensure complete reproducibility.

C.2. R1-Style Prompt for Reinforcement Fine-Tuning

Fig. 11 shows the R1-style prompt used in our reinforcement fine-tuning (RFT) experiments. An example of a specific question is: *”Is there a cat in the image?”*.

C.3. Zero-Shot CoT Prompt of LVLMs:

Fig. 12 shows the zero-shot CoT prompt for LVLMs. An example of a specific question is: *”Is there a person in the image?”*.

Hyper-parameter	Configuration
VLM Init	Qwen3-VL-8B-Instruct
KL Penalty (β)	0
Optimizer	AdamW
Learning Rate	2×10^{-6}
Clipping Range ϵ	0.2
LR Scheduler	Cosine
Weight Decay	0
Precision	BF16
Gradient Clipping	1.0
Per-device Batch Size	1
Gradient Accumulation	4
Rollout Temperature	0.7
Rollout Top-p	0.8
Rollout Top-k	20
Group Size G	8
Max Prompt Length	1024
Max Completion Length	256
Epochs	15
GPUs	4× NVIDIA H100 80GB

Table 9. **Training Configuration.** Key hyper-parameters for GRPO-based Visual-RFT of Qwen3-VL-8B-Instruct.

R1-Style Prompt for Visual RFT

`<image>`
Prompt: Is there a/an {`object`} in the image? Please first provide your reasoning or working out on how you would go about solving the question between `<REASONING>` and `</REASONING>` and then your final answer between `<SOLUTION>` and (put yes or no here) `</SOLUTION>`.

Figure 11. **The R1-style prompt used for reinforcement fine-tuning.** The prompt elicits explicit reasoning (`<REASONING>...</REASONING>`) and a verifiable final answer (`<SOLUTION>...</SOLUTION>`) to enable reward evaluation.

Zero-Shot CoT of LVLMs

`<image>`
Question: {`question`}
Let's think step-by-step and then answer the question based on the given image.

Figure 12. **The zero-shot CoT Prompt of LVLMs.** The prompt of a binary classification task for LVLMs using zero-shot CoT prompting strategy.

D. CLIPScore as a Proxy for Contextual Alignment

While CLIPScore is not a perfect object detector and has known limitations in capturing compositional semantics [32, 80], we use it solely as an external probe to assess the contextual alignment of replaced objects. Specifically, CLIP-guided sampling is applied only to "no"-label cases to select ground-truth nonexistent yet contextually plausible objects with higher CLIPScores, thereby constructing more challenging negatives. Our ablation study B.3 confirms this strategy by showing a significant reduction in negative recall, indicating increased contextual incongruity.

Importantly, CLIPScore is never used for model evaluation but serves as a heuristic signal of object–context compatibility. To ensure robustness, we validate our findings across three independent CLIP variants in A.3, all consistently showing that ORIC "yes" or "no" pairs exhibit higher visual similarity than those in POPE, thus increasing task difficulty. While CLIP's co-occurrence bias may contribute to high scores for out-of-context objects, we argue this reflects its tendency to associate such objects with plausible scenes—precisely the kind of confounding signal our benchmark targets. Despite its limitations, CLIPScore remains a useful proxy for semantic alignment, as supported by recent work [29, 74].

E. Visualization of ORIC Examples

E.1. Error Questions from Human Evaluation

Fig. 13 presents six error cases from **300** sampled questions (150 "yes" and 150 "no" labels) in ORIC using the MSCOCO dataset. We assess two key aspects: accurate object labeling and the appropriateness of visual backgrounds, ensuring incongruous context in both "yes" and "no" questions. The identified errors fall into two categories:

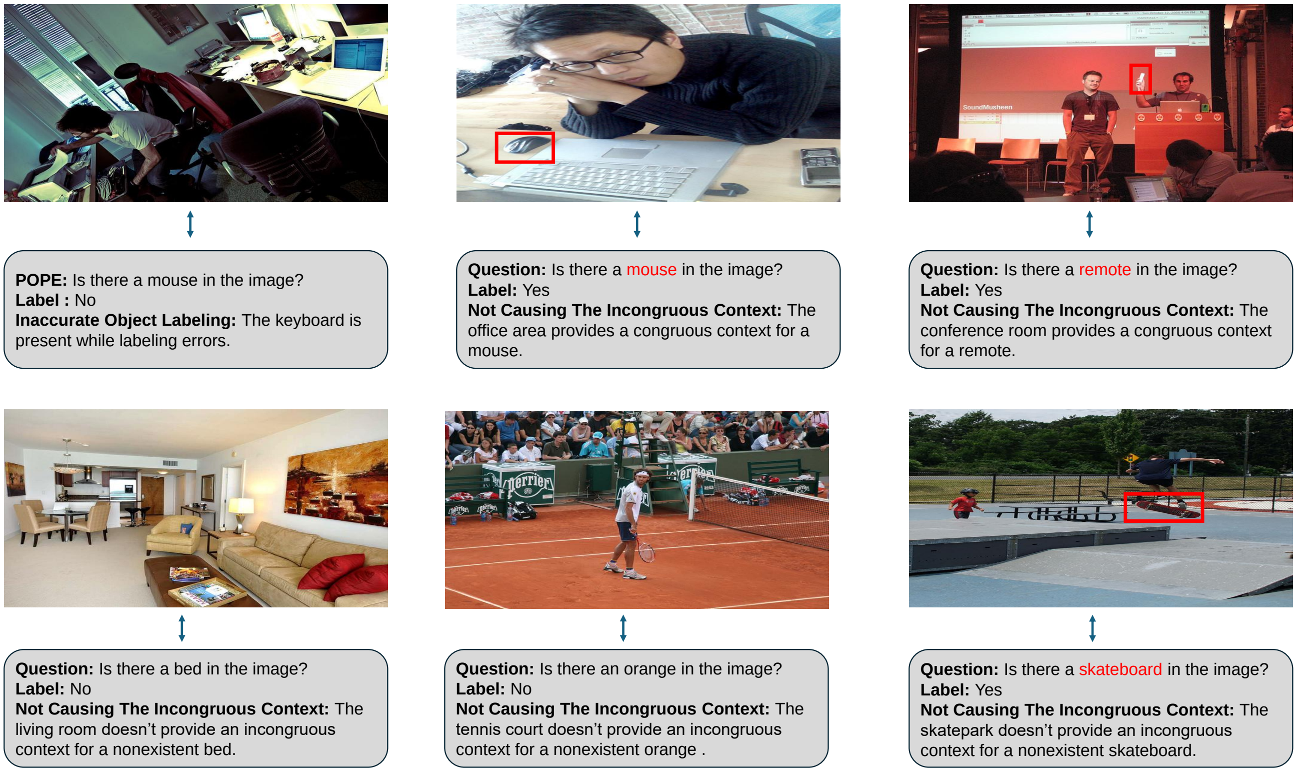
- **Inaccurate Object Labeling**: The presence of objects does not match the actual image content due to errors in human annotation within the MSCOCO dataset.
- **Not Causing the Incongruous Context**: In "yes"-label questions, the visual context aligns with the target object, making the questions less challenging. In "no"-label questions, the visual context does not create incongruity for the nonexistent object.

E.2. ORIC Question Examples

Fig. 14 presents various examples from ORIC. In "yes"-label and "no"-label questions, visual contexts are incongruous with the question-related objects. Our LLM-guided and CLIP-guided sampling method effectively generates challenging questions considering contextual incongruity.

Model	**Overall**				**Label Yes**			**Label No**		
	Pre.	**Rec.**	**F1**	**YP (%)**	**Pre.**	**Rec.**	**F1**	**Pre.**	**Rec.**	**F1**
Closed-source										
GPT-5-2025-08-07	79.50	78.75	**78.61**	42.12	84.14	70.88	**76.92**	71.84	88.62	**79.35**
Vision-encoder-based										
Llama-3.2-11B-Vision	25.00	50.00	33.33	0.00	0.00	0.00	0.00	50.00	100.00	66.67
Chameleon-7B	59.75	50.10	34.08	99.28	50.05	99.38	66.57	69.45	0.82	1.59
BLIP-3	43.14	49.86	42.99	81.54	45.36	51.22	47.02	40.92	48.50	38.96
VILA1.5-13B	65.19	62.40	60.41	28.95	71.44	41.35	51.86	58.92	83.45	68.96
GLM-4v-9B	71.18	64.92	61.99	23.32	82.41	38.25	51.61	59.94	91.60	72.35
Phi-3.5-Vision-Instruct	68.69	68.06	67.79	40.86	72.12	58.92	64.85	65.27	77.20	70.73
InternLM-XComposer2.5-7B	73.32	70.35	69.33	33.77	80.96	54.12	64.17	65.67	86.58	74.49
SmolVLM2-2.2B-Instruct	72.87	71.44	70.95	38.01	78.30	59.45	67.38	67.44	83.42	74.52
Kimi-VL-A3B-Instruct	74.67	72.28	71.58	34.45	82.32	56.73	67.13	67.02	87.83	76.02
Molmo-7B-D-0924	78.92	73.74	71.95	69.34	68.22	93.08	76.61	89.62	54.40	65.59
LLaVA-v1.6-Vicuna-13B	75.29	74.56	74.37	56.94	71.76	81.50	76.19	78.82	67.62	72.55
Janus-Pro-7B	76.60	75.22	74.83	56.42	73.30	81.65	76.71	79.90	68.80	72.95
InternVL3-9B	77.33	76.95	76.87	44.60	80.27	71.55	75.60	74.39	82.35	78.13
Qwen3-VL-8B-Instruct	79.93	79.61	**79.55**	44.94	82.96	74.55	**78.51**	76.91	84.68	**80.59**
Vision-encoder-free										
Fuyu-8B	44.83	50.16	34.16	99.29	50.08	99.45	**66.61**	39.59	0.88	1.71
EVE-7B-HD-v1.0	61.02	56.42	51.59	76.53	54.82	82.95	65.27	67.22	29.90	37.90
Emu3-Chat	67.74	65.79	**64.78**	33.41	73.58	49.20	58.90	61.91	82.38	**70.67**
Open-vocabulary Detection										
OWLv2	73.02	72.25	72.02	40.85	77.23	63.10	69.46	68.81	81.40	**74.58**
Grounding DINO 1.5 Pro	77.02	73.40	**72.48**	68.30	67.13	91.70	**77.51**	86.91	55.10	67.44

Table 10. **Full Experimental Results on ORIC-Bench.** Performance is broken down by model category and label type (Yes/No). We report macro precision (Prec.), recall (Rec.), F1 score, and the proportion of “yes” predictions (YP). Results for LVLMs are averaged over four prompts, while detection models use a single prompt.



POPE: Is there a mouse in the image?
Label : No
Inaccurate Object Labeling: The keyboard is present while labeling errors.

Question: Is there a mouse in the image?
Label: Yes
Not Causing The Incongruous Context: The office area provides a congruous context for a mouse.

Question: Is there a remote in the image?
Label: Yes
Not Causing The Incongruous Context: The conference room provides a congruous context for a remote.

Question: Is there a bed in the image?
Label: No
Not Causing The Incongruous Context: The living room doesn't provide an incongruous context for a nonexistent bed.

Question: Is there an orange in the image?
Label: No
Not Causing The Incongruous Context: The tennis court doesn't provide an incongruous context for a nonexistent orange .

Question: Is there a skateboard in the image?
Label: Yes
Not Causing The Incongruous Context: The skatepark doesn't provide an incongruous context for a nonexistent skateboard.

Figure 13. **Error Examples of ORIC from Human Evaluation.** There are six error cases among the **300** sampled questions in ORIC using the MSCOCO dataset, resulting in an error rate of 2%. These errors can be classified into two categories. **Inaccurate Object Labeling** occurs when the labeled object's presence does not match the actual content of the image. **Not Causing the Incongruous Background** includes cases where the visual context aligns with an existent object in a "yes"-label question or does not introduce incongruity for a nonexistent object in a "no"-label question.

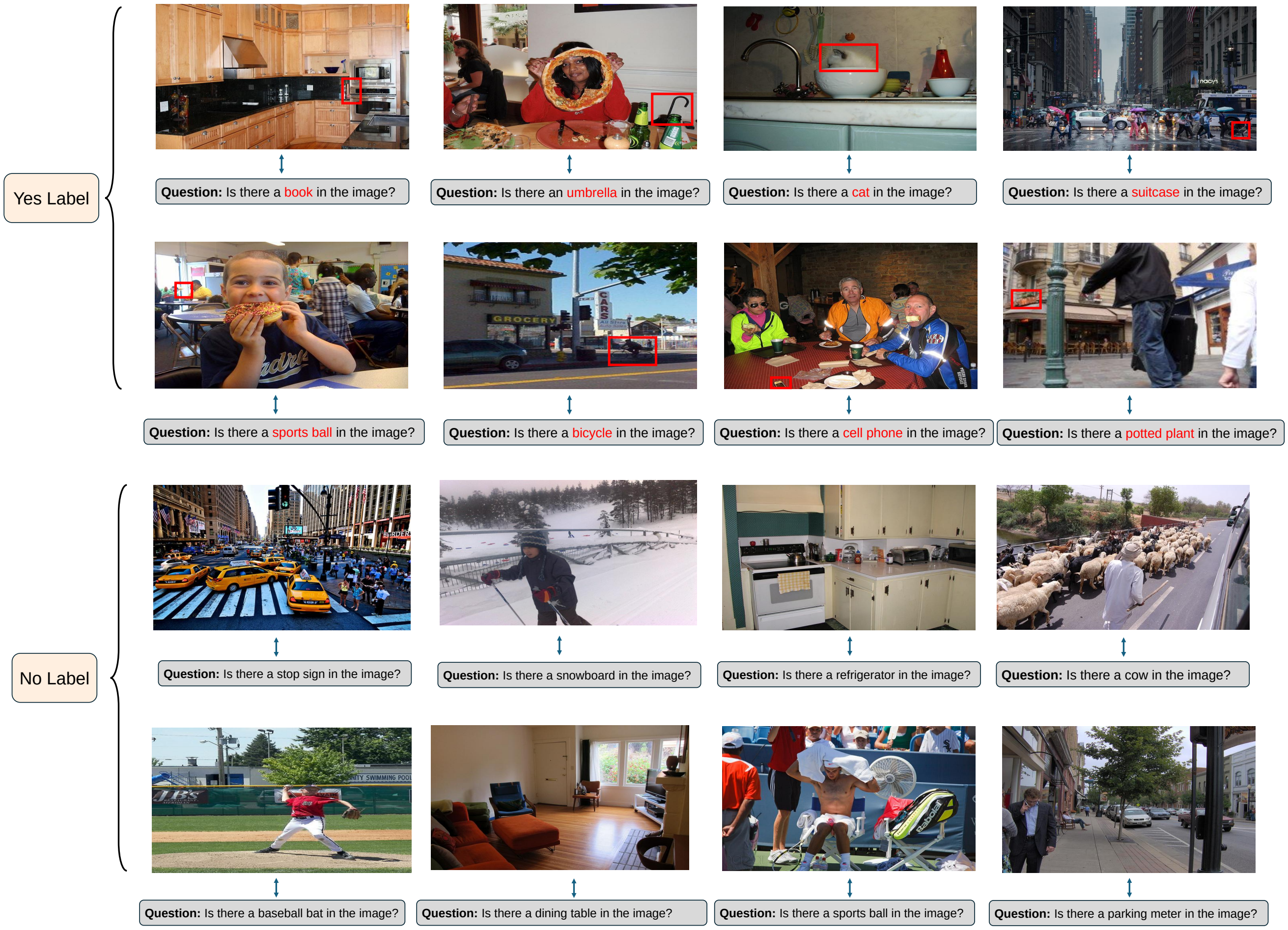

Yes Label
Question: Is there a book in the image?
Question: Is there an umbrella in the image?
Question: Is there a cat in the image?
Question: Is there a suitcase in the image?
Question: Is there a sports ball in the image?
Question: Is there a bicycle in the image?
Question: Is there a cell phone in the image?
Question: Is there a potted plant in the image?
No Label
Question: Is there a stop sign in the image?
Question: Is there a snowboard in the image?
Question: Is there a refrigerator in the image?
Question: Is there a cow in the image?
Question: Is there a baseball bat in the image?
Question: Is there a dining table in the image?
Question: Is there a sports ball in the image?
Question: Is there a parking meter in the image?

Figure 14. **Question Examples of ORIC.** The figure shows sampled question examples from ORIC using the MSCOCO dataset. The first and second rows contain questions labeled “yes,” while the third and fourth rows contain questions labeled ”no.” The red box highlights the bounding boxes of existing objects in “yes”-label questions.