FINER: MLLMs Hallucinate under Fine-grained Negative Queries

Rui Xiao[1,2], Sanghwan Kim[1,2,3], Yongqin Xian[4], Zeynep Akata[1,2,3], Stephan Alaniz[5]

[1]Technical University of Munich [2]Munich Center for Machine Learning
[3]Helmholtz Munich [4]Google [5]LTCI, Télécom Paris, Institut Polytechnique de Paris, France

Abstract

*Multimodal large language models (MLLMs) struggle with hallucinations, particularly with fine-grained queries, a challenge underrepresented by existing benchmarks that focus on coarse image-related questions. We introduce **FI**ne-grained **NE**gative que**R**ies (**FINER**), alongside two benchmarks: **FINER-CompreCap** and **FINER-DOCCI**. Using FINER, we analyze hallucinations across four settings: multi-object, multi-attribute, multi-relation, and "what" questions. Our benchmarks reveal that MLLMs hallucinate when fine-grained mismatches co-occur with genuinely present elements in the image. To address this, we propose **FINER-Tuning**, leveraging Direct Preference Optimization (DPO) on FINER-inspired data. Finetuning four frontier MLLMs with FINER-Tuning yields up to 24.2% gains (InternVL3.5-14B) on hallucinations from our benchmarks, while simultaneously improving performance on eight existing hallucination suites, and enhancing general multimodal capabilities across six benchmarks. Code, benchmark, and models are available at https://explainableml.github.io/finer-project/.*

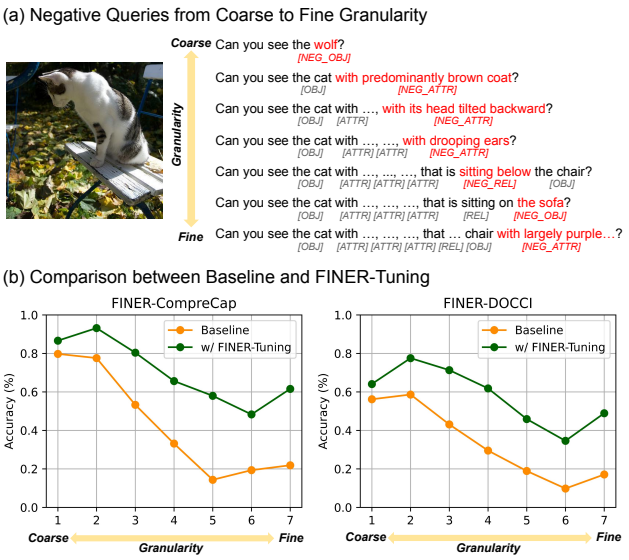

Figure 1. We compare the performance InternVL3.5-14B [38] (Baseline) with the one fine-tuned by FINER-Tuning under negative queries of seven different granularity levels.

1. Introduction

Multimodal large language models (MLLMs) have demonstrated significant progress in visual perception [2] and instruction following [21], enabling increasingly sophisticated image question answering. Real-world users, however, often ask fine-grained questions requiring precise understanding of image content. While current models [4, 22, 38] handle coarse questions reasonably well, it remains unclear whether they can detect nuanced errors in detailed user queries when describing image content. This is critical in domains like medical visual question answering, where trustworthiness requires spotting and correcting errors in complex queries. In the context of natural images, we focus on hallucination [5, 30], the generation of answers unsupported by the image, and define "negative queries" as those asking about non-existent image content. Prior studies show MLLMs often exhibit false-positive hallucination, failing to answer "No" to negative queries [3, 18, 36, 47]. Yet, these probes are largely coarse; POPE and DASH focus on *single* object presence [3, 18], and AMBER includes only *single* objects, attributes, and relations [36]. This raises an important question: *Can MLLMs reject fine-grained mistakes involving multiple objects, attributes, and relations, rather than only coarse mismatches?* To investigate, we first conduct a motivation study, increasing the granularity of negative queries to probe for false positives.

Question granularity affects hallucination. We examine how MLLMs behave as negative queries become progressively *more fine-grained*. Mimicking how human constructs a sentence: starting with a single object, adding attributes, and then relations, we construct queries of increasing granularity from coarse to fine, as shown in Fig. 1. This yields seven levels, each injecting a single, fine-grained contradiction (`NEG_OBJ`, `NEG_ATTR`, or `NEG_REL`) while keeping the

rest of the description visually consistent. For each sample, we feed the model with the image and each of the seven queries separately, limiting the answer to "Yes" or "No", while the correct answer is always "No". We sample from two sources: 320 from FINER-CompreCap and 1,687 from FINER-DOCCI. We report averaged accuracy per level for InternVL3.5-14B [38] and the model finetuned with FINER-Tuning.

As shown in Fig. 1, the accuracy of InternVL3.5-14B steadily decreases with increased query granularity, dropping from $\sim$ 80% at level 1 to $\sim$ 20% by levels 5-7 on FINER-CompreCap, and from $\sim$ 58% at level 1 to $\sim$ 15% by levels 6-7 on FINER-DOCCI. This demonstrates the model's brittleness to fine-grained negations: as granularity increases, it more often answers "Yes" to queries that should be "No", resulting in more false positives. The model finetuned with FINER-Tuning, however, consistently demonstrates performance gains, particularly at finer granularity. This highlights MLLMs' susceptibility to hallucination at finer granularity and the potential for improvement.

Hence, we ask: *Can we systematically study hallucinations under fine-grained negative queries?* Our initial analysis mixes objects, attributes, and relations, hindering isolation of causal factors. To disentangle these, we introduce FINER-CompreCap and FINER-DOCCI, which group queries into four settings: multiple objects (Multi-obj), multiple attributes (Multi-attr), multiple relations (Multi-rel), and "what"-questions (Wh). The first three target existence and binding, assessing whether the model can detect errors hidden in multiple objects, attributes, and relations. The Wh-setting probes factual answering with ill-posed queries, asking "what"-questions about a target object with one incorrect attribute. Together, these four settings reveal whether a model can say "No" to precise but wrong claims, beyond handling coarse mismatches.

2. FINER Benchmarks

Our FINER benchmarks aim to compose negative questions involving multiple semantic elements, i.e., objects, attributes, and relations, to evaluate an MLLM's ability to detect and reason about missing or incorrect components in a scene, even with subtle perturbations. We begin by explaining our benchmark construction as illustrated in Fig. 2.

2.1. Question Construction Pipeline

We base our FINER benchmarks on the scene graph (SG) of an image, encoding objects (OBJ), their attributes (ATTR), and spatial or semantic relations (REL). For each component, we generate negative counterparts (NEG_OBJ, NEG_ATTR, NEG_REL), semantically plausible but incorrect substitutions (e.g., replacing "door frame" with "pillar"). Unlike prior work [3, 18], which rely on a single negative, we generate four distinct negative variants per entity (as described in Sec. 2.3). The initial processing steps are visualized at the top of Fig. 2.

We then use a template-based approach to compose positive questions (q^+) mentioning multiple elements of the same category sampled from the positive SG. For example, a multiple-object question ($q^+_{\text{multi-obj}}$) might be "Can you see cat and door frame?". Corresponding negative questions (q^-) are constructed by replacing one randomly chosen element with a randomly sampled, negative counterpart (e.g., "Can you see cat and pillar?"). The correct answers are "Yes" and "No" respectively. To move beyond binary responses, we construct Multiple Choice Questions (MCQs) requiring the model to specify the correct entities in the image. For example, the correct answer to $q^-_{\text{multi-obj}}$ would be "No, but I can see cat and door frame". We use the other negative options of the same component as distractors for the other answer options (see "Multi-obj" in Fig 2.). Equivalently, we construct $q^{\pm}_{\text{multi-attr}}$ and $q^{\pm}_{\text{multi-rel}}$ from the SGs' attributes and relations. Finally, we create "what"-questions (Wh) asking about an object in relation to another, using either its positive or negative attribute. The complete question template is described in Sec. B in the supplementary.

Benchmarks. Based on this pipeline, we constructed FINER-CompreCap (based on CompreCap [24]) and FINER-DOCCI (based on DOCCI [27]). CompreCap provides human-annotated scene graphs, but is limited to COCO images. DOCCI consists of 5K images with long human-annotated captions which allow us to create a more large-scale question set. The detailed statistics of both benchmarks are in Sec. B in the supplementary. FINER-CompreCap consists of 6,300 Multi-obj, 3,338 Multi-attr, 4,280 Multi-rel, and 3,166 Wh MCQs with a maximum of 6,3,3 objects, attributes, or relations per question. FINER-DOCCI comprises 10,000 Multi-obj, 28,630 Multi-attr, 11,542 Multi-rel, and 20,944 Wh MCQs with a maximum of 6,5,3 objects, attributes, or relations per question. In the following, we detail how we extract the SG from DOCCI, and how we generate the negative components.

2.2. Scene Graph Extraction

For DOCCI, where ground-truth SGs are unavailable, we build a non-panoptic SG by extracting objects, attributes, and relations directly from the human-written long captions. We use a multi-stage pipeline powered by Gemini-2.0-Flash [34], with filtering by a strong MLLM (Qwen2.5VL-72B [4]) and human verification on sampled data, to convert captions into SG-like annotations. The validation steps reduce the risk of introducing incorrect features into the SG which is particularly important for REL. We provide more details regarding the pipeline in Sec. B.2 in supplementary.

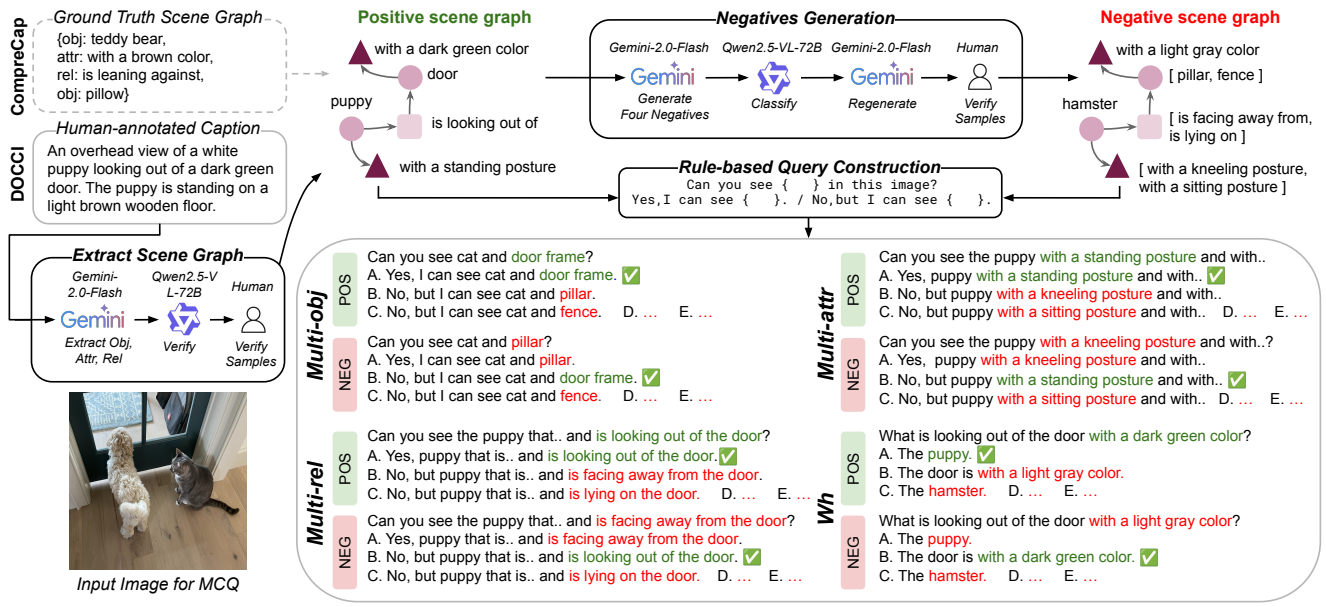

Figure 2. Data construction pipeline for FINER benchmarks. For FINER-DOCCI, we extract the positive scene graph (SG) from DOCCI [27] captions, while for FINER-CompreCap, the SG is provided by CompreCap [24]. From the positive SG, we generate the negative SG using Qwen3-14B [42] as negatives generator for FINER-CompreCap and Gemini-2.0-Flash [34] for FINER-DOCCI. Finally, a rule-based query construction pipeline builds multiple choice questions. In practice, choices are shuffled in both benchmarks.

2.3. Negatives Generation

Starting from the positive SGs, we generate four corresponding negatives for each object, attribute, and relation, using an LLM with carefully designed prompts. We use Qwen3-14B [42] for FINER-CompreCap and Gemini-2.0-Flash [34] for FINER-DOCCI to ensure consistency with the SG creation. To decrease the risk of generated negatives being present in the image, we use a strong MLLM (Qwen2.5-VL-72B) as a discriminator. If it fails to identify the positive item mixed into the negatives, we conclude that at least one negative is ambiguous or present in the image. Based on the MLLM's classification entropy, we identify which negatives require to be regenerated and repeat this process iteratively. Human verifies samples to specify regeneration thresholds. For more details on the negatives generation, please refer to Sec. B.3 in the supplementary.

2.4. Evaluation Setting

As binary "Yes/No" responses are vulnerable to model biases, we use MCQs to move models beyond simple negation and enforce visual understanding, with each MCQ including one correct answer and four distractors. To prevent bias toward positive or negative answers, we pair each negative MCQ (q^-) with its corresponding positive MCQ (q^+), requiring both to be answered correctly. This pairing ensures models cannot succeed by simply memorizing "No" patterns or exploiting label imbalances. As a result, let $M(\cdot)$ be the model, we define paired accuracy as the primary evaluation metric for N paired questions of q^+ and q^-:

$$\text{Acc}_{\text{paired}} = \frac{1}{N}\sum_{i=1}^{N} \Gamma(M(x_i, q_i^+))\,\Gamma(M(x_i, q_i^-)) \quad (1)$$

where $\Gamma(\cdot)$ evaluates to 1 for correct responses and 0 otherwise. This metric requires success on both positive and negative variants, ensuring robustness against false positives and false negatives.

3. Training with FINER (FINER-Tuning)

Observing MLLM vulnerabilities under FINER, we address them with a data-driven training approach via direct preference optimization (DPO) [29] using *fine-grained negative queries*, denoted as FINER-Tuning. Unlike approaches optimizing for simple queries [43, 46, 48], FINER-Tuning employs minimally edited, semantically precise contradictions over objects, attributes, and relations (e.g., "car with yellow bumper" vs. "car with chrome bumper"), including both fine-grained positive and negative queries. Fig. 3 illustrates our training data generation pipeline. It is inspired by the four settings in our benchmarks with both accept and reject answers for every query. This focuses learning on detecting fine-grained hallucinations in the queries, rather than solely avoiding them in the model's responses.

Setup. We select data *avoiding in-distribution leakage*, excluding COCO data [19], and the DOCCI training split [27]. To leverage the availability of dense image annotations, we

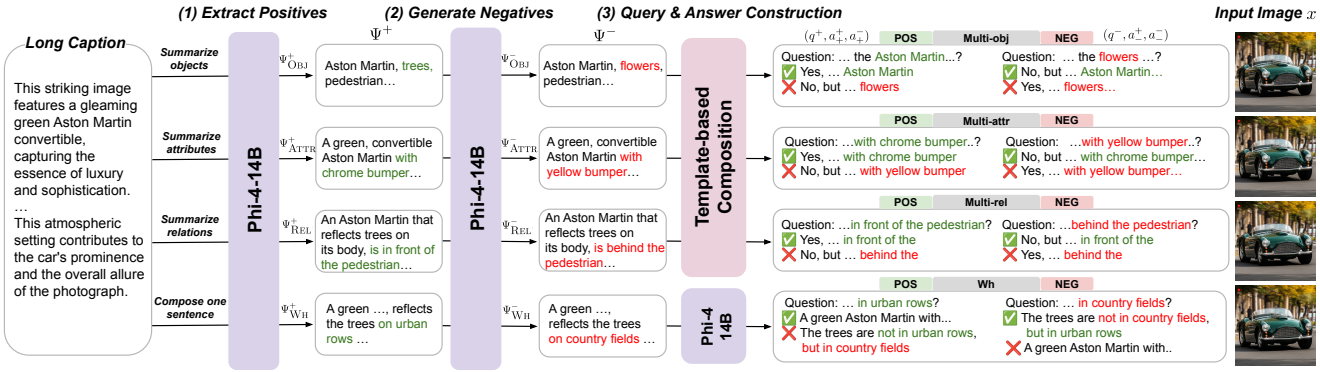

Figure 3. Training data generation pipeline for FINER-Tuning. (1) We adopt long captions from Pixmo [10] and extract diverse phrases with PHI-4-14B [1]. (2) We then prompt the same LLM to modify and generate negative phrases. (3) We construct both positive and negative query-answer tuples via template-based composition or LLM generation.

adopt Pixmo-caption [10] as our base corpus. We further avoid using the LLMs used for benchmark construction, employing `Phi-4-14B` [1] for our training data pipeline.

(1) Extract Positives. As illustrated in Fig. 3, given a long caption, we prompt Phi-4-14B to extract fine-grained positive phrases, mirroring our four evaluation scenarios: Multi-obj, Multi-attr, Multi-rel, and Wh. We define the following four positive phrase types:

$$\Psi^+ \in \left\{\Psi^+_{\text{OBJ}},\ \Psi^+_{\text{ATTR}},\ \Psi^+_{\text{REL}},\ \Psi^+_{\text{WH}}\right\} \tag{2}$$

The LLM produces: Ψ^+_{OBJ}: a phrase summarizing the objects; Ψ^+_{ATTR}: a phrase summarizing attributes for a random object; Ψ^+_{REL}: a phrase summarizing relations between a random object and others; Ψ^+_{WH}: a composed sentence describing two objects with a relation and summarized attributes, subsequently forming a positive question-answer pair. Templates are detailed in Sec. G in the supplementary.

(2) Generate Negatives. Transforming the positive phrases Ψ^+, we generate negative phrases Ψ^- with the same LLM:

$$\Psi^- \in \left\{\Psi^-_{\text{OBJ}},\ \Psi^-_{\text{ATTR}},\ \Psi^-_{\text{REL}},\ \Psi^-_{\text{WH}}\right\} \tag{3}$$

For each phrase type Ψ^+_{T} (where $\text{T} \in \{\text{OBJ}, \text{ATTR}, \text{REL}, \text{WH}\}$), we randomly select one instance of T, and prompt the LLM to replace that instance with a negative, forming Ψ^-_{T}. See Sec. E for full prompts.

(3) Query & Answer Construction. With Ψ^+ and Ψ^-, we construct query-answer pairs for DPO training, including both positive (q^+) and negative (q^-) questions paired with accepted (a^+) and rejected (a^-) responses. a^+ begins with the correct response ("Yes" for q^+, "No" for q^-) and mentions the correct image features, while a^- is the opposite.

For OBJ/ATTR/REL, we directly use question-answer templates on Ψ^+ and Ψ^- to construct (q^+, a^+_+, a^-_+) and (q^-, a^+_-, a^-_-) pairs. We use five templates to avoid overfitting to the benchmark's prompt pattern, as detailed in Sec. G. For WH, data pairs are already constructed by the LLM due to the free-form nature of these questions and answers. Fig. 3 provides example data for all data types and more examples are provided in Sec. C in the supplementary.

DPO Training. This creates a dataset of preference tuples

$$\mathcal{D} = \{(x, q^s, a^+_s, a^-_s)\}, s \in \{+, -\} \tag{4}$$

where x is the image. Let $\pi_\theta(\cdot \mid x, q)$ be the policy and π_{ref} be a frozen reference model. We train with DPO, maximizing the probability that the policy ranks a^+ above a^-:

$$\begin{aligned}
\Delta_\theta(x,q) &:= \log \pi_\theta(a^+ \mid x,q) - \log \pi_\theta(a^- \mid x,q), \\
\Delta_{\text{ref}}(x,q) &:= \log \pi_{\text{ref}}(a^+ \mid x,q) - \log \pi_{\text{ref}}(a^- \mid x,q), \\
\mathcal{L}_{\text{DPO}}(\theta) &= -\,\mathbb{E}_{(x,q,a^+,a^-)\sim\mathcal{D}}\Big[\log \sigma\big(\beta(\Delta_\theta - \Delta_{\text{ref}})\big)\Big].
\end{aligned} \tag{5}$$

where $\sigma(\cdot)$ is the logistic function and $\beta = 0.1$.

4. Experiments

We present experiments of FINER-Tuning on three tasks, i.e., evaluation on FINER benchmarks (Sec. 4.2), other hallucination benchmarks (Sec. 4.3), and general MLLM capabilities (Sec. 4.4). In addition, we show qualitative examples on FINER benchmarks (Sec. 4.5), and ablate important training strategies and subset selections (Sec. 4.6).

4.1. Experimental Setup

Fine-tuning Setup. We are interested in applying FINER-Tuning to frontier-MLLMs: LLaVA-NeXT-7B (LLaVA-1.6-7B) [22], Qwen2.5-VL-7B-Instruct [4], and InternVL-3.5-8B [38]. To test scalability within our compute limits, we also include InternVL-3.5-14B [38]. We fine-tune each model on our constructed data with maximally 160k preference tuples. All models are trained for one epoch using LLaMA-Factory [49] with LoRA [14]. Full training details are in Sec. C in the supplementary.

Evaluation Setup. We evaluate all models on three tasks across 16 benchmarks. We primarily use VLMEvalKit [11]

Table 1. Paired accuracy (Acc_{paired}) results on FINER-CompreCap and FINER-DOCCI. *For Gemini-2.5-Flash, we evaluate on the whole FINER-CompreCap and on 3K MCQs per setting in FINER-DOCCI due to the scale of the benchmark.

Models	Size	FINER-CompreCap				FINER-DOCCI			
		Multi-obj	Multi-attr	Multi-rel	Wh	Multi-obj	Multi-attr	Multi-rel	Wh
Random Guess	-	4.0	4.0	4.0	4.0	4.0	4.0	4.0	4.0
LRV-V2 [20]	13B	6.1	6.8	5.6	4.0	6.3	5.4	6.1	5.2
LLaVA-RLHF [33]	13B	11.4	2.0	1.1	6.9	7.3	3.0	5.1	5.3
RLHF-V [45]	13B	13.4	6.1	1.6	10.8	13.2	7.2	8.1	7.0
OPA-DPO [43]	13B	10.9	3.0	2.2	6.9	8.1	5.5	8.3	8.0
RLAIF-V [46]	12B	62.2	39.6	19.2	20.5	46.5	31.7	32.4	19.4
LLaVA-1.6 [22]	7B	25.3	13.0	7.6	15.3	10.1	12.3	8.2	13.3
+FINER-Tuning	7B	48.4 23.1	38.4 25.4	24.2 16.6	22.1 6.8	26.4 16.3	29.4 17.1	24.7 16.5	18.5 5.2
Qwen2.5-VL [4]	7B	69.2	62.5	30.1	28.9	48.7	47.5	36.7	23.4
+FINER-Tuning	7B	71.4 2.2	67.0 4.5	38.3 8.2	34.8 5.9	49.8 1.1	52.2 4.7	43.4 6.7	28.0 4.6
InternVL-3.5 [38]	8B	75.0	72.5	49.8	23.5	58.1	54.3	41.8	16.8
+FINER-Tuning	8B	77.1 2.1	78.9 6.4	64.1 14.3	34.2 10.7	62.6 4.5	60.1 5.8	52.7 10.9	23.7 6.9
InternVL-3.5 [38]	14B	74.5	68.1	47.0	21.8	58.6	55.9	41.4	15.6
+FINER-Tuning	14B	80.0 5.5	78.9 10.8	71.2 24.2	30.1 8.3	65.9 7.3	65.0 9.1	57.0 15.6	23.0 7.4
InternVL-3.5 [38]	38B	77.8	78.1	66.8	50.9	62.3	64.8	54.2	36.6
Gemini-2.5-Flash [9]*	-	75.7	77.3	77.8	58.2	64.4	64.5	56.7	49.6

for standardized evaluations. For benchmarks not integrated in VLMEvalKit, we follow each benchmark's official evaluation protocol. Refer to Sec. D in supplementary for details.

4.2. Results on FINER benchmarks

Baselines. We primarily compare the performance of the four frontier MLLMs before and after FINER-Tuning, and also show the performance of stronger models such as InternVL-3.5-38B and Gemini-2.5-Flash [34]. Additionally, we benchmark hallucination-aware fine-tuning methods such as RLAIF-V [46], OPA-DPO [43], RLHF-V [45], Llava-RLHF [33], and LRV-Instruct-V2 [20]. Note that different methods are typically based on different MLLMs and fine-tuned on different data. Given their effectiveness on general hallucination reduction, we aim to find out how well they fare on our FINER benchmarks. Furthermore, we estimate human performance with a human study on a subset of 20 MCQs for each setting. The results and details of our human study can be found in Sec. F in the supplementary.

Main results. The results are presented in Tab. 1. Base model capability strongly influences overall performance. Hallucination-aware fine-tuning methods like RLHF-V [45] and LLaVA-RLHF [33] only achieve 1.6% and 1.1% paired accuracy on the Multi-rel subset of FINER-CompreCap. RLAIF-V-12B, while remaining the best among these methods, scores substantially below advanced MLLMs, including Qwen2.5-VL and InternVL-3.5. This shows that mitigating hallucination on previous datasets do not directly translate to our FINER benchmarks, highlighting the importance to start from and improve upon frontier MLLMs.

Meanwhile, FINER-Tuning consistently improves all baselines. Specifically, on FINER-CompreCap, LLaVA-1.6 shows remarkable 23.1% and 25.4%, and 16.6% on Multi-obj, Multi-Attr and Multi-Rel subsets, and InternVL-3.5-14B shows improvements of up to 24.2% (Multi-rel), outperforming its 38B version by 4.4%. On FINER-DOCCI, FINER-Tuning on InternVL-3.5-14B scores on-paar with Gemini-2.5-Flash in 3 out of 4 settings. Moreover, Wh-questions challenge all models. Even InternVL-3.5-38B and Gemini-2.5-Flash achieve only 36.6% and 49.6% Acc_{paired} on FINER-DOCCI, leaving room for future research on reducing hallucinations in FINER.

Different number of objects, attributes and relations. Both FINER benchmarks cover Multi-obj, Multi-attr, and Multi-rel settings. We study how Acc_{paired} changes as the number of entities increases (Fig. 4). Models show similar trends in all three settings: performance drops as the entity counts increases, with much smaller drops in Multi-obj. FINER-Tuning consistently improves performance, with larger gains in Multi-attr and Multi-rel, and the gains grow with higher counts. For example, FINER-Tuning improves InternVL3.5-14B by 8.3%, 19.1% and 28.1% in 6-obj, 3-attr and 3-rel setting on FINER-CompreCap.

4.3. Results on other hallucination benchmarks

FINER-Tuning achieves consistent improvements on FINER benchmarks. Hence, we are interested how well models fine-tuned with FINER-Tuning generalize to other hallucination benchmarks. Additionally, we show the performance of RLAIF-V-12B against its baseline model OmniLMM-12B [28], to see whether other hallucination reduction methods achieve balanced improvements across various hallucination benchmarks. We evaluate models on both discriminative benchmarks like DASH [3], POPE [18],

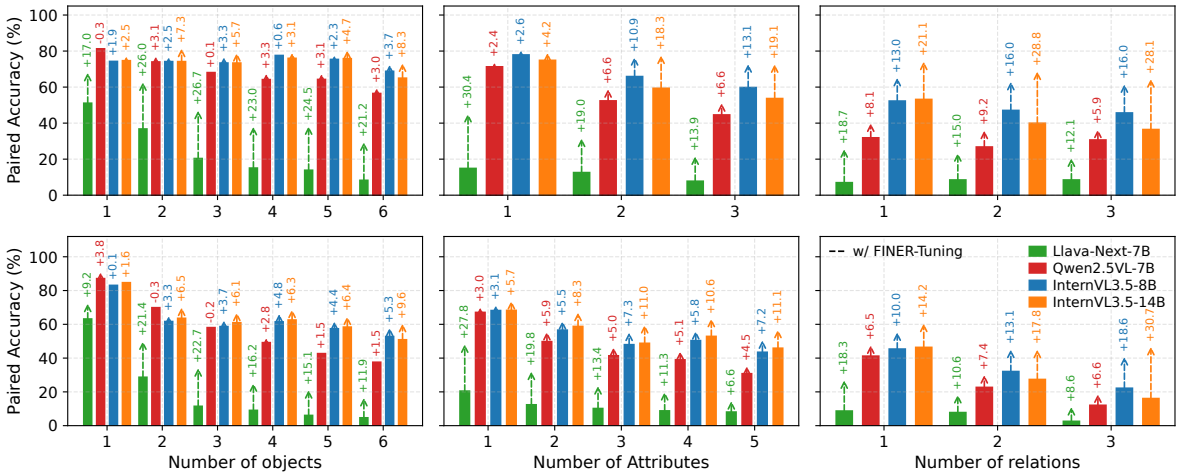

Figure 4. $\text{Acc}_{\text{paired}}$ versus the number of objects, attributes, and relations. Top: FINER-CompreCap; Bottom: FINER-DOCCI. Dashed arrows show the gain from FINER-Tuning.

Table 2. Results on hallucination benchmarks including discriminative (DASH [3], POPE [18], RePOPE [26], HallusionBench [13], AMBER [36], CRPE_R [37]) and generative ones (MMHal-Bench [33], HaloQuest [39]). Sc.:Score (max. 6); HR.: Hallucination Rate.

		DASH	POPE	RePOPE	HallBench	AMBER	CRPE_R	MMHal-Bench		HaloQuest
Models	Size	Acc. ↑	Acc. ↑	Acc. ↑	aAcc. ↑	Acc. ↑	Acc. ↑	Sc. ↑	HR. ↓	Sc. ↑
OmniLMM [28]	12B	79.0	88.0	93.8	54.9	86.9	51.7	3.5	34.0	39.9
+RLAIF-V [46]	12B	76.3 2.7	87.7 0.3	93.4 0.4	53.7 1.2	87.4 0.5	52.2 0.5	4.0 0.5	29.0 5.0	62.4 22.5
LLaVA-1.6 [22]	7B	58.0	88.2	92.3	33.0	78.1	56.5	3.3	43.0	44.2
+FINER-Tuning	7B	57.4 0.6	88.8 0.6	93.2 0.9	36.3 3.3	85.0 6.9	56.0 0.5	3.5 0.2	40.0 3.0	63.5 19.3
Qwen2.5-VL [4]	7B	74.6	86.4	92.4	65.4	85.2	69.9	4.6	18.0	74.8
+FINER-Tuning	7B	76.6 2.0	87.2 0.8	92.8 0.4	68.5 3.1	85.8 0.6	70.7 0.8	4.7 0.1	15.0 3.0	80.8 6.0
InternVL-3.5 [38]	8B	68.3	88.6	91.5	71.0	88.2	67.7	4.5	19.0	62.4
+FINER-Tuning	8B	74.5 6.2	89.4 0.8	93.1 1.6	73.0 2.0	88.6 0.4	68.0 0.3	4.6 0.1	14.0 5.0	73.5 11.1
InternVL-3.5 [38]	14B	55.8	89.5	91.8	69.5	88.0	67.2	4.7	11.0	65.0
+FINER-Tuning	14B	61.3 5.5	90.2 0.7	93.6 1.8	71.2 1.7	89.4 1.4	69.0 1.8	4.7	10.0 1.0	71.0 6.0

RePOPE [26], HallusionBench [13], AMBER [36], CRPE relation split (CRPE_R) [37], as well as generative benchmarks like MMHalBench [33] and HaloQuest [39]. The summarized results are shown in Tab. 2. In supplementary, We further include detailed breakdowns (Tabs. 13 and 14), results for AMBER *generative* (Tab. 15) and comparisons with more methods (Tab. 16). Intuitively, FINER-Tuning strengthens discrimination through FINER training; our results on discriminative benchmarks confirm this. FINER-Tuning consistently improves Qwen2.5-VL and InternVL-3.5 across all benchmarks. On DASH, it boosts the two InternVL-3.5 variants by 6.2% and 5.5%. LLaVA-1.6 also gains 6.9% on AMBER with FINER-Tuning. FINER-Tuning further reduces hallucination on generative benchmarks. On MMHal-Bench, it lowers hallucination rate for all base models, reaching 10% with InternVL-3.5-14B. On HaloQuest, it improves LLaVA-1.6 by 19.3%. Even for Qwen2.5-VL and InternVL-3.5, we observe at least 6% gains. In contrast, while RLAIF-V delivers strong gains on generative benchmarks, its improvements on discriminative tasks are less consistent, where FINER-Tuning benefits both. RLAIF-V degrades performance compared to the base OmniLMM on benchmarks like DASH, POPE, RePOPE, and HallusionBench. By comparing these "deltas" between fine-tuned models and baselines, we show that FINER-Tuning is a *balanced* approach that leads to a comprehensive reduction in hallucination. These results also validate the effectiveness of FINER benchmarks, showing that improvements on FINER benchmarks align with broader improvements in other benchmarks as well.

4.4. Results on general capabilities

Since FINER-Tuning adds fine-grained negative queries to DPO, a natural concern is *over-rejection*: the model becoming overly cautious, refusing answerable questions, or regressing on existing skills. To test this, we compare each

Table 3. Results on six general purpose MLLM benchmarks. M.S.: MMStar [7]; Text: TextVQA [32]; Chart: ChartQA [25]; M.P.: MMVP [35]; N.B.: NaturalBench [17]; V*: V* Bench [40]

Models	M.S.	Text	Chart	M.P.	N.B.	V*	Avg.
OmniLMM-12B	39.7	**64.5**	24.2	69.7	**26.9**	52.9	**46.3**
+RLAIF-V	**40.9**	**64.5**	**25.1**	**70.0**	19.4	**54.4**	45.7
LLaVA-1.6-7B	37.6	63.7	54.4	65.0	15.7	53.9	48.4
+FINER-Tuning	**39.2**	**63.9**	**54.9**	**68.7**	**19.8**	**55.0**	**50.3**
Qwen2.5-VL-7B	63.7	84.9	**87.0**	76.7	**34.1**	72.7	69.8
+FINER-Tuning	**64.7**	**85.1**	86.4	**77.3**	**34.1**	**72.8**	**70.1**
InternVL3.5-8B	68.0	77.8	**86.7**	76.7	30.4	69.1	68.1
+FINER-Tuning	**68.3**	**77.9**	**86.7**	**77.0**	**31.1**	**71.2**	**68.7**
InternVL3.5-14B	67.2	**77.2**	86.4	78.3	30.7	68.0	68.0
+FINER-Tuning	**67.7**	**77.2**	**86.8**	**78.7**	**35.5**	**70.2**	**69.4**

base model and its FINER-Tuning-tuned counterpart on six additional benchmarks: MMStar [7] (general abilities), TextVQA [32], ChartQA [25], MMVP [35] (vision-centric abilities), NaturalBench [17] (compositionality), and V* (visual search). The results are shown in Tab. 3. Unlike prior work reporting an "alignment tax", with gains on target benchmarks at the cost of general ability [47], FINER-Tuning avoids this trade-off and even improves strong baselines on general benchmarks (improving InternVL3.5-14B by 1.4%). This shows that FINER provides a useful training signal that complements the model's internal capabilities.

4.5. Qualitative Results

Fig. 5 shows four FINER-CompreCap examples; more qualitative results, including FINER-DOCCI, are in Sec. E in the supplementary. FINER-Tuning avoids the spurious "necklace" in the Multi-obj case and correctly identifies the fine color details of the strawberry-patterned food in the Multi-attr case. In the Multi-rel example, both Qwen2.5-VL and InternVL3.5 hallucinate the second relation as "hiding behind the football". In the Wh example, FINER-Tuning shifts InternVL-3.5-14B from answering "bear" to flagging the incorrect attribute of the rock. These examples indicate that FINER-Tuning helps the model detect fine-grained errors and locate correct the information in complex queries.

4.6. Ablation Studies

Training strategies. FINER-Tuning trains on both positive and negative queries $\{(x, q^+, a^+_+, a^-_+),\ (x, q^-, a^+_-, a^-_-)\}$. To ablate this setting, we investigate the training with and without positive questions, and compare the performance of DPO against supervised fine-tuning (SFT).

We train four InternVL-3.5-8B variants accordingly and compare with the baseline in Tab. 4. Results show mixed outcomes for SFT: with both queries, SFT reduces Multi-obj performance by 36.7% relative to the baseline. DPO with only negative queries exceeds the base model but still lags behind DPO with both query types (FINER-Tuning), underscoring the value of training with both.

Table 4. Ablation study on different training strategies. SFT methods only use a^+. The base model is InternVL-3.5-8B [38]. Q.Type: Query Type; M.S.: MMStar [7]

Method	Q.Type		FINER-CompreCap				Other	
	Neg	Both	Obj	Attr	Rel	Wh	RePOPE	M.S.
Base	-	-	74.2	71.9	49.8	25.5	91.5	68.0
+SFT	✓		47.4	59.7	53.8	**38.7**	69.1	61.7
+SFT		✓	37.5	49.5	55.2	18.9	92.2	63.3
+DPO	✓		75.8	75.2	52.4	29.8	**93.1**	**68.3**
+DPO		✓	**76.5**	**78.3**	**64.1**	36.1	**93.1**	**68.3**

Table 5. Training-on-subset ablation for FINER-Tuning with InternVL-3.5-8B [38]. Obj/Attr/Rel denote Multi-obj/Multi-attr/Multi-rel for both training and evaluation.

Train Subset	FINER-CompreCap				Other	
	Obj	Attr	Rel	Wh	RePOPE	M.S.
Base	74.2	71.9	49.8	25.5	91.5	68.0
Obj	**78.8**	76.4	54.2	28.7	**93.5**	67.9
Attr	71.3	76.7	56.8	26.5	91.5	68.2
Rel	69.2	73.0	**66.7**	24.1	91.4	67.7
Wh	75.9	75.3	55.0	**46.5**	92.9	**68.3**
All	76.5	**78.3**	64.1	36.1	93.1	**68.3**

Training on subsets. Our training data matches the benchmark query types: Multi-Obj, Multi-Attr, Multi-Rel, and Wh. We train InternVL-3.5-8B on each subset separately and compare to FINER-Tuning trained on all subsets, keeping the total number of training samples fixed at 160k. As shown in Tab. 5, models trained only on Multi-Obj, Multi-Rel, or Wh achieve the best scores on their corresponding tests. Notably, they also improve on other settings, suggesting the model is not merely echoing supervision from data: FINER fosters a more general rejection pattern that transfers beyond the seen subset. Overall, training on all subsets yields the most balanced results.

5. Related Works

Hallucination Benchmarks. POPE [18] probes object hallucination by asking yes-or-no questions. RePOPE [26] identifies and corrects annotation errors in POPE. Amber [36] categorizes hallucinations into "object," "relation," and "attribute" types in its discriminative subset. A common limitation of these benchmarks is their reliance on the MSCOCO dataset [19]. Therefore, DASH [3] applies retrieval to select challenging images from LAION-5B [16]. CRPE [37] focuses on relation hallucinations but is limited to single-relation cases. NOPE [23] targets non-existent objects, not attribute or relation hallucinations. ROPE [8] probes object classes with visual prompts (bounding boxes). Unlike ROPE, our Multi-obj setting randomly replaces a

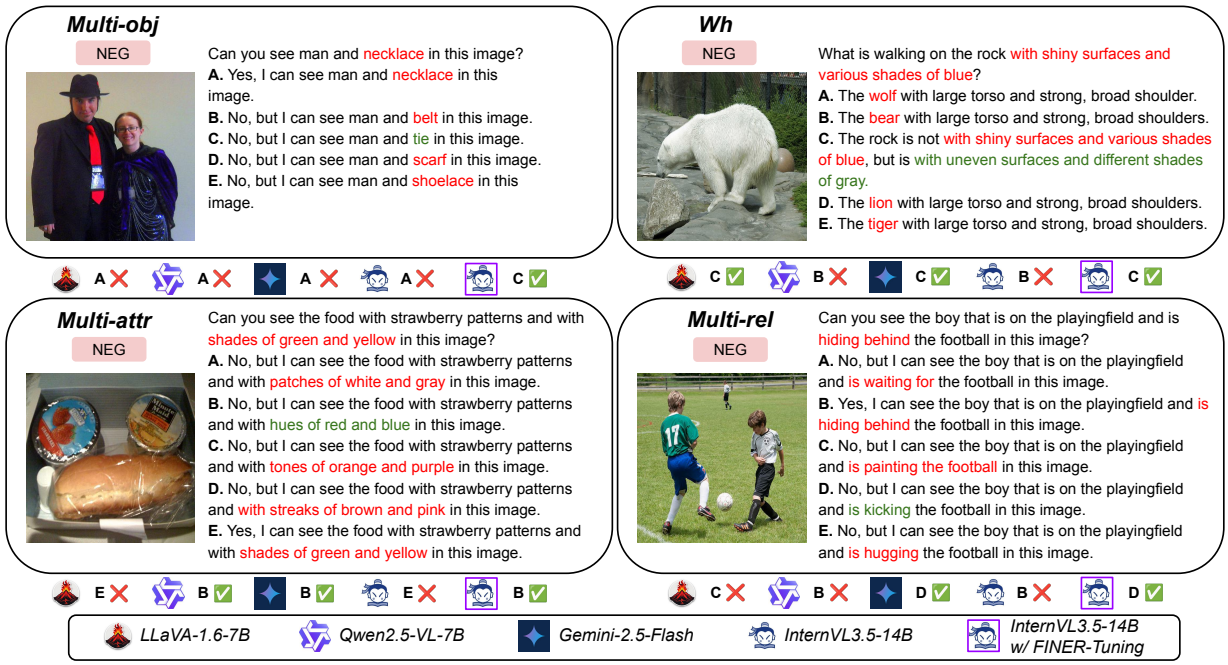

Figure 5. Qualiative examples of FINER-CompreCap MCQs for each category together with MLLM answers.

positive object with a negative one and does not rely on MSCOCO/ADE20K box annotations [19, 50]. MMHal-Bench [33] evaluates hallucination via eight types of questions with limited scale. HaloQuest [39] includes a "false premise" subset with a similar motivation to our Wh setting. However, our setting differs: we target false premises in fine-grained attributes of existing objects, whereas HaloQuest primarily targets non-existent objects.

Hallucination-aware Fine-tuning. Prior work reduces hallucinations via supervised or contrastive tuning and instruction-based data augmentation: LRV-Instruct [20] adds negative instructions to MiniGPT-4 [51] and mPLUG-Owl [44]; HALVA [31] builds paired correct vs. hallucinated responses for contrastive learning; PerturboLLaVA [6] trains under misleading contexts; REVERSE [41] adds uncertainty tokens and retrospective reasoning. Other studies use preference learning: OPA-DPO [43] constructs on-policy corrections with GPT-4V; CHiP [12] decomposes the DPO loss into three hierarchies; HA-DPO [48] detects and corrects hallucinations with GPT-4; LLaVA-RLHF [33] and RLHF-V [45] rely on human preferences; RLAIF-V [46] iterates with model feedback. FINER-Tuning differs in three ways: (1) we target fine-grained negative *input* queries, not only response-side errors [31, 33, 43, 45, 46, 48]; (2) we post-train frontier MLLMs beyond the LLaVA family [31, 43] and show strong performance against FINER; (3) we use standard DPO with a scalable data pipeline and a small LLM [1] for annotation, avoiding costly closed-source models and multi-iteration training [6, 20, 31, 43, 46, 48].

6. Conclusion and Limitation

Conclusion. We introduced FINER, a suite of fine-grained negative queries that reveals how current MLLMs fail under precise negations. Systematic evaluation across all four settings of FINER-CompreCap and FINER-DOCCI shows that even frontier MLLMs remain vulnerable to FINER-induced hallucinations. To address this, we proposed FINER-Tuning, a simple, model-agnostic recipe that aligns models to react correctly to fine-grained negative queries. Across diverse backbones and training regimes, FINER-Tuning consistently reduces hallucinations and improves paired accuracy on FINER benchmarks, as well as a wide range of hallucination and general purpose benchmarks. Despite these gains, high-granularity cases and Wh questions remain challenging. Future work will focus on stronger negation-aware reasoning, that comprehensively enhances MLLMs' capabilities. We envision FINER as a start for incentivizing better benchmarks and methods to mitigate hallucinations under fine-grained queries.

Limitations. Despite careful filtering, the large-scale benchmarks are not fully curated by human; constructing a noise-free, fully human-validated FINER benchmark is left for future research. Our rule-based MCQ construction enables flexible entity combinations but may reduce question naturalness. Future work could refine phrasing with LLMs or human rewrites while ensuring correctness. In addition, our Multi-rel subsets contain at most three relations, which, with a suitable data source, could be extended to improve model capabilities and further challenge FINER.

Acknowledgments. This work was supported by the German Research Foundation (DFG): SFB 1233, Robust Vision: Inference Principles and Neural Mechanisms, TP A2, project number: 276693517. This work was partially funded by the ERC (853489 - DEXIM), the German Federal Ministry of Education and Research (BMBF, grant number: 01IS18039A), and the Alfried Krupp von Bohlen und Halbach Foundation, which we thank for their generous support. This work is also supported by Hi! PARIS and ANR/France 2030 program (ANR-23-IACL-0005). This project was also supported by Google.org with a Google Cloud Platform (GCP) credit award. The authors gratefully acknowledge the scientific support and resources of the AI service infrastructure *LRZ AI Systems* provided by the Leibniz Supercomputing Centre (LRZ) of the Bavarian Academy of Sciences and Humanities (BAdW), funded by Bayerisches Staatsministerium für Wissenschaft und Kunst (StMWK). In addition, the authors gratefully acknowledge the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) for funding this project by providing computing time on the GCS Supercomputer JUWELS [15] at Jülich Supercomputing Centre (JSC).

References

[1] Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, et al. Phi-4 technical report. *arXiv*, 2024. 4, 8

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv*, 2023. 1

[3] Maximilian Augustin, Yannic Neuhaus, and Matthias Hein. Dash: Detection and assessment of systematic hallucinations of vlms. In *ICCV*, 2025. 1, 2, 5, 6, 7

[4] 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*, 2025. 1, 2, 4, 5, 6

[5] Zechen Bai, Pichao Wang, Tianjun Xiao, Tong He, Zongbo Han, Zheng Zhang, and Mike Zheng Shou. Hallucination of multimodal large language models: A survey. *arXiv*, 2024. 1

[6] Cong Chen, Mingyu Liu, Chenchen Jing, Yizhou Zhou, Fengyun Rao, Hao Chen, Bo Zhang, and Chunhua Shen. Perturbollava: Reducing multimodal hallucinations with perturbative visual training. *ICLR*, 2025. 8

[7] 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? *NeurIPS*, 2024. 7

[8] Xuweiyi Chen, Ziqiao Ma, Xuejun Zhang, Sihan Xu, Shengyi Qian, Jianing Yang, David Fouhey, and Joyce Chai. Multi-object hallucination in vision language models. In *NeurIPS*, 2024. 7

[9] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv*, 2025. 5

[10] Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In *CVPR*, 2025. 4

[11] Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, et al. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In *ACM MM*, 2024. 4

[12] Jinlan Fu, Shenzhen Huangfu, Hao Fei, Xiaoyu Shen, Bryan Hooi, Xipeng Qiu, and See-Kiong Ng. Chip: Cross-modal hierarchical direct preference optimization for multimodal llms. In *ICLR*, 2025. 8

[13] 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 *CVPR*, 2024. 6

[14] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 2022. 4

[15] Jülich Supercomputing Centre. JUWELS Cluster and Booster: Exascale Pathfinder with Modular Supercomputing Architecture at Juelich Supercomputing Centre. *Journal of large-scale research facilities*, 2021. 9

[16] LAION. Releasing re-laion-5b: transparent iteration on laion-5b with additional safety fixes, 2024. Accessed: 30 aug, 2024. 7

[17] Baiqi Li, Zhiqiu Lin, Wenxuan Peng, Jean de Dieu Nyandwi, Daniel Jiang, Zixian Ma, Simran Khanuja, Ranjay Krishna, Graham Neubig, and Deva Ramanan. Naturalbench: Evaluating vision-language models on natural adversarial samples. In *NeurIPS*, 2024. 7

[18] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *EMNLP*, 2023. 1, 2, 5, 6, 7

[19] 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 *ECCV*, 2014. 3, 7, 8

[20] Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. Aligning large multi-modal model with robust instruction tuning. In *ICLR*, 2024. 5, 8

[21] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 1

[22] 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. 1, 4, 5, 6

[23] Holy Lovenia, Wenliang Dai, Samuel Cahyawijaya, Ziwei Ji, and Pascale Fung. Negative object presence evaluation (nope) to measure object hallucination in vision-language models. In *Proceedings of the 3rd Workshop on ALVR*, 2024. 7

[24] Fan Lu, Wei Wu, Kecheng Zheng, Shuailei Ma, Biao Gong, Jiawei Liu, Wei Zhai, Yang Cao, Yujun Shen, and Zheng-Jun Zha. Benchmarking large vision-language models via directed scene graph for comprehensive image captioning. In *CVPR*, 2025. 2, 3

[25] Ahmed Masry, Xuan Long Do, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of ACL*, 2022. 7

[26] RePOPE: Impact of Annotation Errors on the POPE Benchmark. Neuhaus, yannic and hein, matthias. *arXiv*, 2025. 6, 7

[27] Yasumasa Onoe, Sunayana Rane, Zachary Berger, Yonatan Bitton, Jaemin Cho, Roopal Garg, Alexander Ku, Zarana Parekh, Jordi Pont-Tuset, Garrett Tanzer, et al. Docci: Descriptions of connected and contrasting images. In *ECCV*, 2024. 2, 3

[28] OpenBMB. Large multi-modal models for strong performance and efficient deployment, 2024. 5, 6

[29] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *NeurIPS*, 2023. 3

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

[31] Pritam Sarkar, Sayna Ebrahimi, Ali Etemad, Ahmad Beirami, Sercan Ö Arık, and Tomas Pfister. Data-augmented phrase-level alignment for mitigating object hallucination. In *ICLR*, 2025. 8

[32] Amanpreet Singh, Vivek Natarjan, Meet Shah, Yu Jiang, Xinlei Chen, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *CVPR*, 2019. 7

[33] Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liang-Yan Gui, Yu-Xiong Wang, Yiming Yang, et al. Aligning large multimodal models with factually augmented rlhf. *arXiv*, 2023. 5, 6, 8

[34] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv*, 2023. 2, 3, 5

[35] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *CVPR*, 2024. 7

[36] Junyang Wang, Yuhang Wang, Guohai Xu, Jing Zhang, Yukai Gu, Haitao Jia, Jiaqi Wang, Haiyang Xu, Ming Yan, Ji Zhang, et al. Amber: An llm-free multi-dimensional benchmark for mllms hallucination evaluation. *arXiv*, 2023. 1, 6, 7

[37] Weiyun Wang, Yiming Ren, Haowen Luo, Tiantong Li, Chenxiang Yan, Zhe Chen, Wenhai Wang, Qingyun Li, Lewei Lu, Xizhou Zhu, et al. The all-seeing project v2: Towards general relation comprehension of the open world. In *ECCV*, 2024. 6, 7

[38] Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv*, 2025. 1, 2, 4, 5, 6, 7

[39] Zhecan Wang, Garrett Bingham, Adams Wei Yu, Quoc V Le, Thang Luong, and Golnaz Ghiasi. Haloquest: A visual hallucination dataset for advancing multimodal reasoning. In *ECCV*, 2024. 6, 8

[40] Penghao Wu and Saining Xie. V?: Guided visual search as a core mechanism in multimodal llms. In *CVPR*, 2024. 7

[41] Tsung-Han Wu, Heekyung Lee, Jiaxin Ge, Joseph E Gonzalez, Trevor Darrell, and David M Chan. Generate, but verify: Reducing hallucination in vision-language models with retrospective resampling. In *NeurIPS*, 2025. 8

[42] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv*, 2025. 3

[43] Zhihe Yang, Xufang Luo, Dongqi Han, Yunjian Xu, and Dongsheng Li. Mitigating hallucinations in large vision-language models via dpo: On-policy data hold the key. In *CVPR*, 2025. 3, 5, 8

[44] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. mplug-owl: Modularization empowers large language models with multimodality. *arXiv*, 2023. 8

[45] Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng, Maosong Sun, et al. Rlhf-v: Towards trustworthy mllms via behavior alignment from fine-grained correctional human feedback. In *CVPR*, 2024. 5, 8

[46] Tianyu Yu, Haoye Zhang, Yuan Yao, Yunkai Dang, Da Chen, Xiaoman Lu, Ganqu Cui, Taiwen He, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. Rlaif-v: Aligning mllms through open-source ai feedback for super gpt-4v trustworthiness. In *CVPR*, 2025. 3, 5, 6, 8

[47] Zongmeng Zhang, Wengang Zhou, Jie Zhao, and Houqiang Li. Robust multimodal large language models against modality conflict. In *ICML*, 2025. 1, 7

[48] Zhiyuan Zhao, Bin Wang, Linke Ouyang, Xiaoyi Dong, Jiaqi Wang, and Conghui He. Beyond hallucinations: Enhancing lvlms through hallucination-aware direct preference optimization, 2023. 3, 8

[49] Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *ACL*, 2024. 4

[50] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017. 8

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