FINER: MLLMs Hallucinate under Fine-grained Negative Queries

Supplementary Material

A. Extended Related Works

A.1. Hallucination benchmarks

CHAIR [37] benchmarks object hallucination in image captioning by measuring how many generated words actually appear in the image, based on ground-truth captions and object segmentations. However, the CHAIR metric suffers from instability issues [22]. POPE [22] simplifies hallucination detection by asking models yes-or-no questions. RePOPE [33] identifies annotation errors in POPE and provides a revised version. Amber [44] evaluates hallucinations in both generative and discriminative settings. In the discriminative setting, it categorizes hallucinations into "object," "relation," and "attribute" types. A common limitation of these benchmarks is their reliance on the MSCOCO dataset [23]. To better detect object hallucinations at scale, DASH [3] adopts a retrieval-based approach to select images from LAION-5B [20]. CRPE [45] focuses on relation-based hallucinations but limits its evaluation to single-relation cases.

Beyond hallucination detection, MMMC [56] introduces the concept of "modality conflicts," referring to mismatches between the image and the text query, an approach we consider coarse-grained negative querying. FLAIR [50] constructs DOCCI-FG that also adopts DOCCI captions to test how well vision-language models understand images from a fine-grained perspective. COSMOS [19] evaluates and further improves fine-grained vision-language alignment via a self-distillation approach. The "Blind-faith-in-Text" phenomenon [12] shows that when a conflicting textual context is prefixed to a query, models tend to trust the text more than the image. Similarly, Hallu-PI [13] evaluates hallucinations by appending additional images or texts as a perturbation. In our work, we do not add extra textual context. Instead, we design user queries that contain subtle and nuanced conflicts with the image, allowing us to study hallucination behavior without altering the conversational setup. MMVU [28] also proposes a benchmark that investigates "negative questions." The key difference is that our work studies this problem at a finer level of granularity.

HaloQuest [47] includes a "false premise" subset with a similar motivation to our Wh setting. However, our setting differs because our false premises lie in the fine-grained attributes of existing objects, while HaloQuest mainly focuses on non-existent objects. Likewise, NOPE [30] mainly evaluates hallucinations involving non-existent objects but does not test hallucinations related to attributes or relations. ROPE [8] evaluates object hallucinations by prompting MLLM to pick the correct objects corresponding multiple input visual prompts. While this approach shares similarity with our Multi-obj subset, we aim for more flexibility by directly inserting the negative object at random position in the prompt and we do not rely on bounding boxes annotation from MSCOCO-Panoptic [23] or ADE20K [59]. ODE [43] introduces an open-set dynamic hallucination evaluation to prevent data contamination. This also aligns with our intuition to adopt DOCCI [34] as an additional data source and create the less-saturated FINER-DOCCI.

A.2. Hallucination-aware Fine-tuning

To reduce hallucinations, various fine-tuning techniques have been developed for MLLMs. Closely related to our motivation, LRV-Instruct [24] applies supervised fine-tuning (SFT) to MiniGPT-4 [61] and mPLUG-Owl [53], and introduces negative instructions by manipulating objects and factual knowledge using GPT-4 [2]. HALVA [38] leverages Gemini Vision Pro [41] to construct both correct and hallucinated responses, and applies a contrastive loss between them, explicitly pushing the model away from hallucinated generations.

PerturboLLaVA [6] appends misleading textual context as perturbations generated by GPT-4o [2] and trains the model via instruction tuning to remain robust under such distracting inputs. REVERSE [49] expands the model's vocabulary with special uncertainty tokens and builds a large-scale instruction-following dataset; the model learns to perform retrospective reasoning whenever these tokens are triggered, allowing it to revise potentially hallucinated content. RLHF-V [54] and LLaVA-RLHF [40] apply reinforcement learning from human feedback (RLHF) to vision-language models, using human preference signals to improve response quality and reduce hallucinations. RLAIF-V [55] instead leverages AI feedback (RLAIF): a stronger teacher model provides automatic preference judgments, and the student model is updated in a self-evolving manner over multiple training rounds.

Several studies employ Direct Preference Optimization (DPO) to reduce hallucinations. OPA-DPO [52] constructs on-policy data for hallucination mitigation and uses GPT-4V for fine-grained hallucination correction in the training set. CHiP [15] decomposes the DPO objective into response-level, segment-level, and token-level components to better localize hallucinations. HA-DPO [57] also uses GPT-4 [2] to identify and correct hallucinations in model outputs. POVID [60] adopts GPT-4V to inject hallucinated objects, attributes, and relations directly into the dispreferred responses, encouraging the model to reject these patterns during training.

In light of these works, our approach differs in three main aspects. First, most prior studies [38, 40, 52, 54, 55, 57, 60] focus on detecting and correcting hallucinations in model responses, whereas we explicitly construct fine-grained negative *input queries* at the object, attribute, and relation level. Second, previous efforts [38, 52] primarily target the LLaVA family, while we directly post-train several state-of-the-art MLLMs and evaluate them on the FINER benchmarks, improving model's robustness against nuanced errors in queries. Third, FINER-Tuning follows the standard DPO algorithm and does not require multi-iteration training as in RLAIF-V. Unlike prior works [6, 24, 38, 52, 57, 60] that rely heavily on costly closed-source models to build training data, we propose a scalable pipeline that uses an open-source LLM [1] to generate high-quality preference pairs from existing long-caption datasets.

B. FINER Benchmark Details

In this section, we describe the construction of FINER-CompreCap and FINER-DOCCI. FINER-CompreCap starts from human-annotated positive scene-graphs (SGs) with minor edits (Sec. B.1). FINER-DOCCI derives positive SGs from dense captions (Sec. B.2). We then apply the same negative-generation and filtering pipeline to obtain negative SGs (Sec. B.3). Finally, both positive and negative SGs are converted into benchmark questions via our rule-based MCQ pipeline (Sec. B.4).

The two benchmarks are motivated slightly differently. FINER-CompreCap builds on human-annotated SGs, supporting more precise evaluation. In contrast, FINER-DOCCI explores whether dense captions can be used to synthesize SGs beyond COCO object classes and images, enabling open-set evaluation [43] at substantially larger scale. As a result, FINER-DOCCI is primarily designed to validate our findings at scale, rather than to maximize per-sample annotation fidelity.

B.1. Positive SG for FINER-CompreCap

CompreCap [31] offers 560 human-annotated images, each with a scene-graph (SG) annotation. Each SG annotation already consists of objects, attributes, and relations. The attribute annotations in the original SG are lists of simple sentences, which we rewrite with Qwen3-14B [51] into "with {attr}" phrases without changing their original meaning. The original relation annotations are also sentences describing a relation between a subject and an object. Therefore, we use a rule-based method to parse the relation sentences into dictionary-like annotations. These steps are necessary because we need to combine objects, relations, and attributes in our MCQ construction. We manually inspect the positive annotations to ensure their integrity. Since our preprocessing only changes sentence structure and does not introduce new annotations, it is robust. We provide an example SG in Fig. 7. As shown in Fig. 7, the original attribute "The cat is black and orange" is rewritten as "with a black and orange color". Meanwhile, the original relation "The cat is lying on a desk" is parsed into a dictionary-like structure.

B.2. SG Extraction Pipeline for FINER-DOCCI

DOCCI [34] consists of 5,000 images, each paired with a detailed human-annotated caption. Such rich descriptions already contain the necessary information about objects, attributes, and relations. Fig. 8 shows an example caption together with the positive scene graph extracted by Gemini-2.0-Flash [10].

Directly prompting an LLM to "summarize" a full scene graph is known to be brittle and prone to errors. Instead, inspired by PerturboLLaVA [6], which prompts an LLM to extract objects, attributes, and relations from long captions, we design a conservative two-stage extraction pipeline that decomposes the task into simpler subproblems and incorporates explicit cross-checks and human validation.

Stage 1: object and attribute extraction. In the first stage, we only ask Gemini-2.0-Flash to extract objects and their attributes from the caption. The model is instructed to copy phrases *verbatim* from the caption and to avoid inventing new entities or attributes. This turns the problem into a pure information extraction task rather than open-ended generation. The prompt is visualized in Fig. 24. Human annotators inspect randomly sampled outputs to check the robustness of this stage, as the model only needs to detect and group textual mentions instead of inferring unseen content.

Stage 2: relation extraction and validation. In the second stage, we consider pairs of extracted objects and ask Gemini-2.0-Flash whether the caption explicitly states a relation between them. Given the full caption and a candidate object pair, Gemini is instructed to either (i) return the exact relation phrase from the caption, or (ii) not return anything if no relation is explicitly mentioned. The model is explicitly told not to infer or imagine relations that are not written in the caption. This again restricts Gemini to acting as an information extractor, which increases reliability. The prompt is displayed in Fig.25.

Even with these restrictions, some errors in the extracted relations remain. To further filter noisy relations, we perform a joint visual-textual validation step. For each candidate relation, we:

- run a binary classifier with Qwen2.5-VL-72B [4] to decide whether the relation holds in the image; and
- query Gemini again, this time asking whether the relation is *explicitly* supported by the caption.

If both models disagree with the proposed relation, we discard it. Among the misclassified relations, we further ask human annotators to verify a subset of 400 samples and, whenever they spot errors, remove incorrect extracted rela-

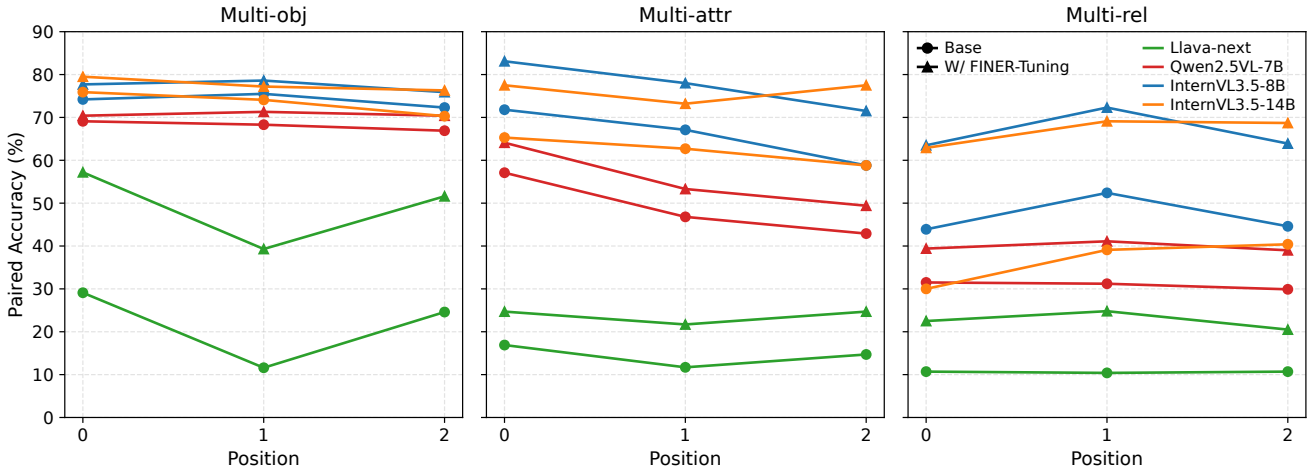

Figure 6. Positional bias analysis on FINER-CompreCap. We select all $q^{\pm}_{\text{multi-obj}}$, $q^{\pm}_{\text{multi-attr}}$, and $q^{\pm}_{\text{multi-rel}}$ that contain three entities. Since each q^{-} always has exactly one negated entity, we cyclically move that negated entity to each of the three positions (and move the corresponding positive entity accordingly), and compute the averaged paired accuracy $\text{Acc}_{\text{paired}}$ for each position.

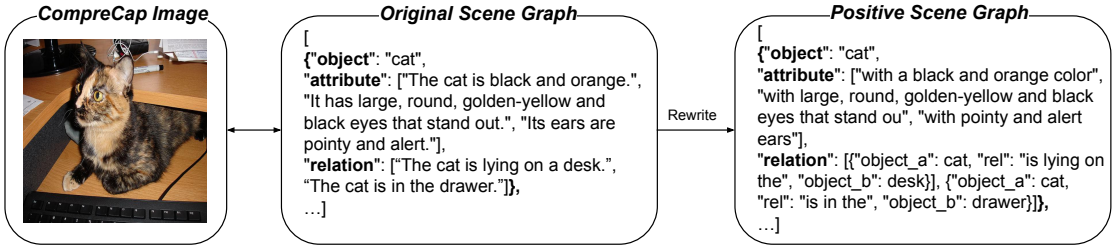

Figure 7. Example of positive scene graph (SG) in FINER-CompreCap. CompreCap [31] already pairs each image with SG-like annotation. We further adopts Qwen3-14B [51] to simply rewrite attribute sentences into phrases.

tion annotations. In total, this joint process of Qwen2.5-VL, Gemini, and humans filters out 1,771 relations.

Overall, this pipeline is deliberately conservative: we only keep relations that are supported by the caption (via extraction) and by the image (via a strong MLLM), with additional human checks on top. This design prioritizes precision over recall and makes our extracted SG for FINER-DOCCI more reliable despite the known challenges of using LLMs for scene-graph extraction.

Quality Assessment. To assess the quality of the extracted objects, attributes, and relations in the positive SG of FINER-DOCCI, we run InternVL3.5-8B [46] as a binary classifier. For each extracted object, attribute, or relation, the model is asked to answer "Yes" or "No" regarding its presence in the image. As a baseline, we apply the same procedure to the positive SG of FINER-CompreCap, whose scene graphs are human-annotated. The results are reported in Tab. 6. InternVL3.5-8B achieves comparable performance (96.4% vs. 96.1%) when classifying ground-truth objects in both benchmarks. For attributes, its accuracy on FINER-DOCCI is 3.2% lower than on FINER-CompreCap. Given that the SG in FINER-DOCCI is much larger in scale than in FINER-CompreCap (see Tab. 7), this gap is acceptable. Notably, the accuracy on relations in the positive SG of FINER-DOCCI is slightly higher than that of FINER-CompreCap (85.1% vs. 82.8%). This likely reflects that the relation annotations in FINER-DOCCI are more detailed, providing the MLLM with more information to verify their correctness, rather than indicating that the human-annotated relations in FINER-CompreCap are of lower quality.

B.3. Negatives Generation Pipeline.

Having obtained the positive scene graphs (SGs) for both FINER-CompreCap and FINER-DOCCI, we construct a pipeline for generating negatives. For each object (OBJ), attribute (ATTR), and relation (REL), we generate four negative counterparts, denoted as NEG_OBJ, NEG_ATTR, and NEG_REL.

LLM-based negatives proposal. We first use an LLM as a "negatives generator". For FINER-DOCCI we use Gemini-2.0-Flash [41], and for FINER-CompreCap we

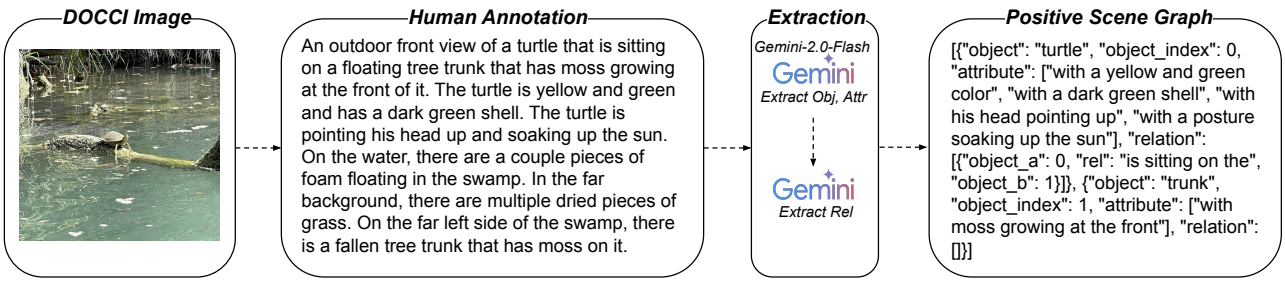

Figure 8. Example positive scene graph (SG) extracted by Gemini-2.0-Flash [41]. Given a long human-annotated caption from DOCCI [34], we apply a two-stage extraction pipeline to obtain the positive SG.

Table 6. Quality assessment of the extracted positive objects, attributes, and relations for FINER-DOCCI using InternVL3.5-8B [46] as a binary classifier. As a baseline, we also run InternVL3.5-8B as a binary classifier to classify the human annotations from FINER-CompreCap.

	FINER-CompreCap			FINER-DOCCI		
	Obj	Attr	Rel	Obj	Attr	Rel
Acc. (%)	96.4	91.5	82.8	96.1	88.3	85.1

use Qwen3-14B [51]. Given a positive phrase (OBJ, ATTR, or REL), the LLM is prompted to produce four negative phrases that have the opposite or a clearly different meaning from the positive. This step is efficient and does not directly inherit visual biases from any vision model, since it operates purely in text space.

A limitation of this step is that some generated negatives may in fact describe entities that are present in the image. Such "false negatives" are harmful for evaluation. Given the scale of the two positive SGs, pure human validation on the whole set is unfortunately not possible, so we need an automatic way to detect and filter these false negatives.

MLLM-based discrimination and entropy. To filter these cases, we use Qwen2.5-VL-72B [4] as a visual discriminator. For each positive phrase x (where x can be either OBJ, ATTR, or REL) and its four candidate negatives $\{x_j^-\}_{j=1}^4$, we form a five-choice multiple-choice question with the candidate set

$$\mathcal{C}(x) = \{x, x_1^-, x_2^-, x_3^-, x_4^-\}.$$

We query Qwen2.5-VL-72B with the image and the set $\mathcal{C}(x)$, and obtain a probability distribution

$$p = (p_1, \ldots, p_5), \qquad \sum_{i=1}^{5} p_i = 1,$$

over the five choices. We treat the original positive x as the correct label. If the model selects x, the classification is correct; otherwise it is misclassified.

We compute the entropy of the model output

$$H(p) = -\sum_{i=1}^{5} p_i \log p_i, \tag{6}$$

where the logarithm is natural. Low entropy means that the model is very confident in one of the options, while high entropy indicates uncertainty. If Qwen2.5-VL-72B makes a misclassification by choosing one negative while maintaining very low entropy, this indicates high confidence in its prediction. This likely reflects that the chosen entity somehow exists in the image (or, of course, the model can also be too confident about an actually wrong prediction).

We show several examples in Fig. 9. Empirically, we observe that many bad negatives that actually appear in the image lead to misclassifications with very low entropy. For example, in one sample, "ground" is proposed as a negative for the object "wall". Since the ground region is clearly visible in the image, Qwen2.5-VL-72B strongly prefers the option "ground", with an entropy of $H(p) = 0.0119$. This indicates that the model is highly confident that "ground" is present in the image, and therefore this negative should be rejected. In such cases, we prompt the LLM again and rewrite the negative, for example from "ground" to "ceiling", which does not appear in the image.

However, low entropy does not always mean that the negative actually appears in the image; the MLLM can also be confidently wrong. For instance, in the car example in Fig. 9, Qwen2.5-VL-72B misclassifies the relation phrase "is behind the" with low entropy $H(p) = 0.0119$, even though "is behind the" is a valid negative. In this case, we still replace it with a new negative proposal such as "is on top of the", which remains valid. Since our primary goal is to remove negatives that truly appear in the image, occasionally regenerating valid negatives is acceptable.

Entropy-based filtering with human verification. We denote the entropy filtering threshold as θ. For each benchmark and each level (object, attribute, relation), we choose a separate threshold θ.

To set these thresholds, we first run Qwen2.5-VL-72B on

Table 7. Statistics for the generating negative scene graph for FINER-CompreCap (denoted as C-SG) and FINER-DOCCI (denoted as D-SG).Counts: number of objects, attributes and relations inside the SG annotation.θ: entropy-based filtering threshold; #Re-gen.: number of re-generated negatives.

Benchmark		θ	Counts	# Re-gen.
C-SG	Obj	0.8	3505	320
	Attr	0.8	4509	414
	Rel	0.4	3494	173
D-SG	Obj	0.8	24,528	3,242
	Attr	0.4	52,911	2,827
	Rel	0.8	15,342	2,143

the entire dataset and record, for each example, the model prediction and the corresponding entropy $H(p)$. We then collect all misclassified examples and sort them in ascending order of entropy. Starting from the lowest-entropy region, a human annotator verifies 10 misclassified examples and labels whether the proposed negative actually appears in the image. We then incrementally increase the candidate entropy threshold and, at each step, again sample 10 misclassified examples around the current threshold for human verification. We repeat this process until no "bad negatives" (negatives that truly appear in the image) are found among the 10 inspected samples; we then take the current entropy value as the threshold θ such that misclassified examples with $H(p) < \theta$ are likely to be true false negatives (the negative phrase is in the image), while those with higher entropy are retained as hard but valid negatives.

During the full pipeline, each negative candidate that leads to a misclassification with $H(p) < \theta$ is sent back to the LLM and regenerated. The new proposal is checked again by Qwen2.5-VL-72B with the same procedure. After each round of regeneration and classification, we subsample a small set of misclassified examples and ask a human annotator to inspect the remaining negatives. This human-in-the-loop process is to reduce the risk of systematic errors introduced by the automatic filtering pipeline.

We summarize the thresholds θ, the total number of samples, and the number of regenerated negatives for each benchmark and each level (Obj, Attr, Rel) in Tab. 7.

Quality Assessment. Given the scale of our benchmarks, we adopt a model-based assessment approach. We assess the quality of the generated negatives by evaluating Qwen2.5-VL-72B on objects (Obj), attributes (Attr), and relations (Rel) in FINER-CompreCap and FINER-DOCCI. Tab. 8 reports the corresponding classification accuracies. For example, Qwen2.5-VL-72B achieves 94.1% accuracy when selecting the positive relation from its four negative counterparts in FINER-CompreCap, which supports the quality of the constructed negatives in this benchmark. On FINER-DOCCI, the model attains close to 90% accuracy in objects and attributes. Note that FINER-DOCCI is designed to test whether rich, human-described semantics can enable large-scale hallucination evaluation, rather than building a small, noise-free benchmark fully curated by humans. Given its substantially larger scale and higher difficulty, we consider the achieved negatives classification accuracies to show a sufficient negatives quality that helps validating our findings *at scale*.

Table 8. Quality assessment of generated negatives. We show the classification accuracy of Qwen2.5-VL-72B [4] after classifying the objects (obj), attributes (attr) and relations (rel) in FINER-CompreCap and FINER-DOCCI

	FINER-CompreCap			FINER-DOCCI		
	obj	attr	rel	obj	attr	rel
Acc. (%)	89.8	91.1	94.1	89.5	88.3	82.8

B.4. MCQ Design

Having obtained the positive SG and negative SG for FINER-CompreCap and FINER-DOCCI, we now construct MCQs. Sec. 2.1 already provides an explanation of our MCQ construction pipeline: we use a fixed template to compose both positive and negative MCQs ($q^{\pm}_{\text{multi-obj}}$, $q^{\pm}_{\text{multi-attr}}$, $q^{\pm}_{\text{multi-rel}}$). For $q^{\pm}_{\text{Wh}}$, we prompt Gemini-2.0-Flash to construct the question templates. We describe the two templates in detail.

Fixed question template. We use a simple yes/no-style template for all $q^{\pm}_{\text{multi-obj}}$, $q^{\pm}_{\text{multi-attr}}$, and $q^{\pm}_{\text{multi-rel}}$. To make the format explicit, we display it as a small template box:

```
Can you see {X} in this image?
A. Yes, I can see {Y} in this image.
B. No, but I can see {Z_1} in this image.
C. No, but I can see {Z_2} in this image.
D. No, but I can see {Z_3} in this image.
E. No, but I can see {Z_4} in this image.
```

Here, $\{X\}$, $\{Y\}$, and $\{Z_1\}, \ldots, \{Z_4\}$ are placeholders that will later be filled with phrases. In the benchmark, the choices are randomly shuffled.

Construction of $q^{\pm}_{\text{multi-obj}}$, $q^{\pm}_{\text{multi-attr}}$, and $q^{\pm}_{\text{multi-rel}}$. We only describe the construction process for $q^{\pm}_{\text{multi-obj}}$; the same procedure is applied to $q^{\pm}_{\text{multi-attr}}$ and $q^{\pm}_{\text{multi-rel}}$.

From the positive SG of an image, we first sample k distinct objects and concatenate them into a positive multi-object phrase P^{+}_{obj} (for example, "dog, ball, and tree"). This phrase P^{+}_{obj} contains only objects that truly appear in the image. We then randomly select one of these k objects,

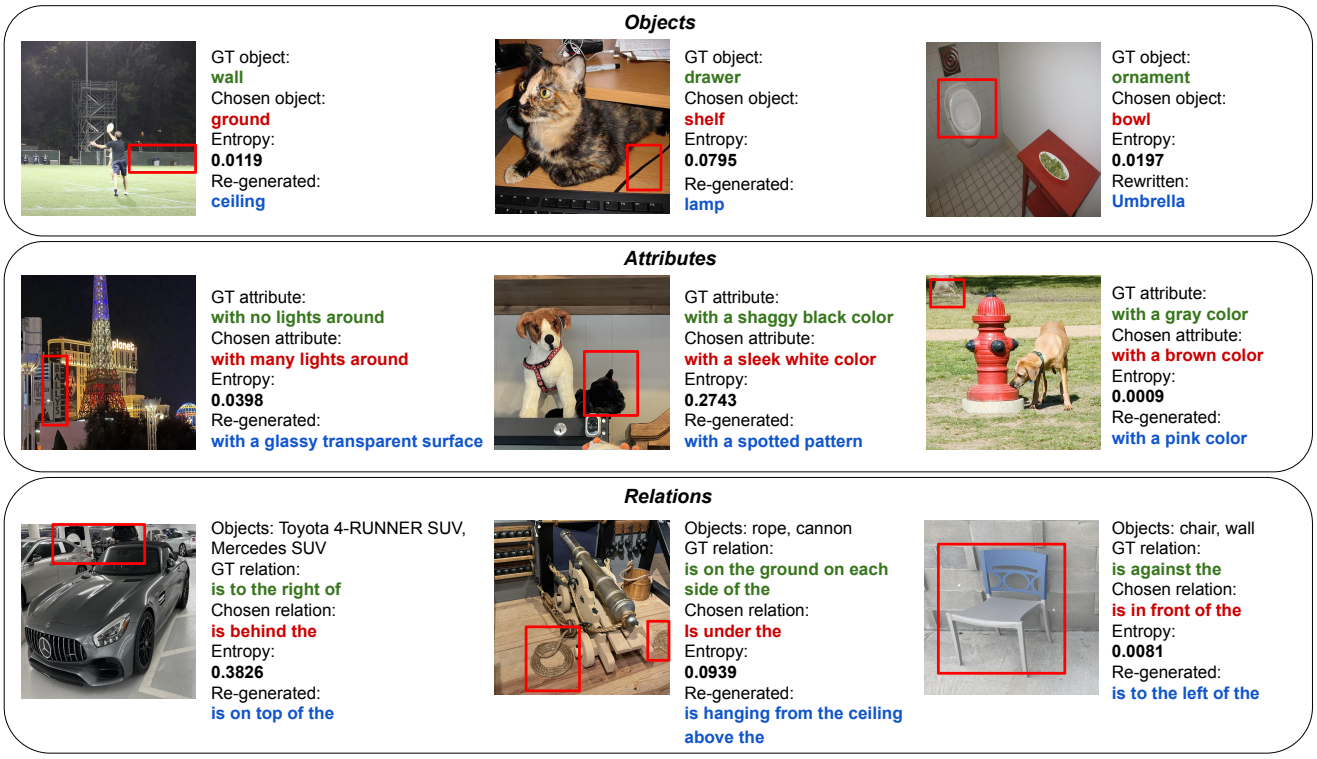

Figure 9. Examples of entropy-based filtering for objects, attributes, and relations. The corresponding objects are shown with red bounding boxes. The ground-truth object/attribute/relation is highlighted in green. We prompt Qwen2.5-VL-72B [4] to select the positive among four negatives. Green text indicates that the model makes an incorrect prediction and chooses a negative with low entropy scores. Blue text shows new negative candidates generated by the LLM. The examples are from both FINER-CompreCap and FINER-DOCCI.

denote the selected object by o, and retrieve its four negative counterparts $\{o_j^-\}_{j=1}^4$ from the negative SG. For each $j \in \{1, \ldots, 4\}$, we form a corrupted phrase $P_{\text{obj},j}^-$ by replacing o in P_{obj}^+ with o_j^- while keeping all other objects unchanged. Thus we obtain one positive phrase P_{obj}^+ and four negative phrases $P_{\text{obj},1}^-, \ldots, P_{\text{obj},4}^-$.

To build a *positive* MCQ $q_{\text{multi-obj}}^+$, we instantiate the template by setting

$$\begin{aligned} \{X\} &= P_{\text{obj}}^+, \\ \{Y\} &= P_{\text{obj}}^+, \\ \{Z_j\} &= P_{\text{obj},j}^- \text{ for } j = 1, \ldots, 4. \end{aligned}$$

In this case, the question and the "Yes" option both describe the true configuration P_{obj}^+, while each "No, but I can see $\{Z_j\}$" option contains exactly one incorrect object. The option that contains P_{obj}^+ is treated as the correct answer.

To build a *negative* MCQ $q_{\text{multi-obj}}^-$, we flip the roles of the positive and corrupted phrases in the template. We randomly choose one corrupted phrase, say $P_{\text{obj},1}^-$, and set

$$\begin{aligned} \{X\} &= P_{\text{obj},1}^-, \\ \{Y\} &= P_{\text{obj},1}^-, \\ \{Z_1\} &= P_{\text{obj}}^+, \\ \{Z_j\} &= P_{\text{obj},j}^- \text{ for } j = 2, 3, 4. \end{aligned}$$

Now the question asks about the corrupted phrase $P_{\text{obj},1}^-$, which does *not* match the image. Consequently, the "Yes" choice becomes a false-positive option, because it incorrectly confirms the existence of $P_{\text{obj},1}^-$. The option that says "No, but I can see P_{obj}^+ in this image" is now the correct answer, since it both denies the existence of $P_{\text{obj},1}^-$ and affirms the true configuration P_{obj}^+. Note that we randomly pick which corrupted phrase is used as the query, so each of $P_{\text{obj},1}^-, \ldots, P_{\text{obj},4}^-$ has an equal chance to replace $\{X\}$.

This fixed pattern keeps the surface form of the questions consistent across all MCQs while allowing the underlying content to vary. The same construction is applied to $q_{\text{multi-attr}}^{\pm}$ and $q_{\text{multi-rel}}^{\pm}$ by treating attribute phrases and relation phrases as the basic units instead of objects.

Wh question generation. Wh questions have more flexible

surface forms than yes/no questions. To construct Wh-style questions, we start from a relation triplet in the scene graph,

$$(\texttt{OBJ}_1, \texttt{REL}, \texttt{OBJ}_2),$$

where $\texttt{OBJ}_1$ and $\texttt{OBJ}_2$ are two objects and $\texttt{REL}$ is the relation between them. Each object can have one or more attributes, e.g. $\mathcal{A}(\texttt{OBJ}_1)$ for the first object.

Given a triplet $(\texttt{OBJ}_1, \texttt{REL}, \texttt{OBJ}_2)$, we randomly choose one of the two objects as the *answer target* and treat the other as *context*. Concretely, we either ask about $\texttt{OBJ}_1$ given $\texttt{OBJ}_2$ or about $\texttt{OBJ}_2$ given $\texttt{OBJ}_1$. We then mask the answer target in the textual description and prompt Gemini-2.0-Flash to produce a natural Wh question. For example, for the relation (dog, is standing under, table), Gemini-2.0-Flash can generate questions such as

"What is standing under the table?" (ask about the dog)
"What is the dog standing under?" (ask about the table).

Wh MCQ template. Once we fix the Wh question pattern for a given triplet, we turn it into an MCQ by providing five answer options. We represent the question body and the five options using placeholders:

```
Q: {Q}
A. {O1}
B. {O2}
C. {O3}
D. {O4}
E. {C}
```

Here, $\{Q\}$ is the Wh question text, $\{O_1\}, \ldots, \{O_4\}$ are object-level answer candidates, and $\{C\}$ is a full-sentence *correction* option that explicitly talks about the attribute of the target object. In the benchmark, the choices are randomly shuffled.

Construction of $q^{\pm}_{\text{Wh}}$. We illustrate the construction using the running example with the context object "dog" and the answer target "table". The dog has a positive attribute A^+ (e.g. "with brown fur") and a sampled negative attribute A^- (e.g. "with yellow fur"), while the relation and context (e.g. "standing under the table") are fixed by the triplet $(\texttt{OBJ}_1, \texttt{REL}, \texttt{OBJ}_2)$.

From the positive SG, we select "table" as the target object $o^\star$. We then randomly pick three negative objects o_1^-, o_2^-, o_3^- for this slot from the negative SG (e.g. "chair", "bench", "sofa"). Starting from the Wh question

"What is the dog standing under?",

We insert an attribute phrase for the dog and obtain an attribute-conditional question template

$$q(A) \equiv \text{"What is the dog } A \text{ standing under?"}.$$

Filling this template with A^+ or A^- gives us a positive or negative Wh question with the same surface pattern. Note that in the FINER benchmarks, a single object can have multiple attributes. In that case, we include all of its attributes in the descriptive context, then randomly choose one of them as the target attribute A^+ and sample the corresponding negative attribute as A^-.

Positive Wh MCQ. For the *positive* Wh question q^+_{Wh}, we fill the attribute slot with the true attribute A^+ and instantiate the MCQ template as

$$\begin{aligned}
\{Q\} &= q(A^+),\\
\{O_1\} &= o^\star,\\
\{O_j\} &= o^-_{j-1} \quad \text{for } j = 2, 3, 4,\\
\{C\} &= \text{"The dog is not } A^+\text{, but is } A^-\text{."}
\end{aligned}$$

The question $\{Q\}$ is now a valid Wh question about the image, and $\{O_1\}$ (the true object $o^\star$) is the correct answer. The three options $\{O_2\}, \{O_3\}, \{O_4\}$ are incorrect objects, and the correction sentence $\{C\}$ is also incorrect because it denies the true attribute A^+.

Negative Wh MCQ. For the *negative* Wh question q^-_{Wh}, we instead fill the question template with the negative attribute A^-, which makes the premise of the question partially inconsistent with the image. We keep the same four object candidates but flip the correction sentence:

$$\begin{aligned}
\{Q\} &= q(A^-),\\
\{O_1\} &= o^\star,\\
\{O_j\} &= o^-_{j-1} \quad \text{for } j = 2, 3, 4,\\
\{C\} &= \text{"The dog is not } A^-\text{, but is } A^+\text{."}
\end{aligned}$$

Now the question $\{Q\}$ is *incorrect* with respect to the image, because it attributes A^- to the dog. The object-only options $\{O_1\}, \ldots, \{O_4\}$ all implicitly accept the wrong attribute in the question and are therefore treated as incorrect. The correction option $\{C\}$ is the unique correct answer: it denies the wrong attribute A^- and restores the true attribute A^+.

In summary, q^+_{Wh} asks a Wh question whose premise matches the image and is answered by the true object $o^\star$, while q^-_{Wh} asks a Wh question whose premise uses a corrupted attribute and is correctly answered only by the explicit correction sentence. This construction mirrors the positive/negative symmetry used for the yes/no-style templates and keeps the Wh MCQs tightly grounded in the underlying scene graph.

Benchmark statistics. As described in Sec. 2.1, our MCQ design constructs both positive and negative questions for four settings: $q^{\pm}_{\text{multi-obj}}$, $q^{\pm}_{\text{multi-attr}}$, $q^{\pm}_{\text{multi-rel}}$, and $q^{\pm}_{\text{wh}}$. We present the detailed statistics of FINER-CompreCap and FINER-DOCCI in Tab. 9.

Table 9. Distribution of MCQ pairs over entity counts in FINER-CompreCap (FINER-C) and FINER-DOCCI (FINER-D). For each setting, we refer the entity counts for Obj/Attr/Rel as k and the corresponding number of pairs n_k in matching order. $(1, 6)$ represents that k ranges from 1 to 6.

Benchmark	Setting	k	# pairs n_k
FINER-C	$q^{\pm}_{\text{multi-obj}}$	$(1, 6)$	$560, 560, 560, 558, 535, 377$
	$q^{\pm}_{\text{multi-attr}}$	$(1, 3)$	$966, 472, 231$
	$q^{\pm}_{\text{multi-rel}}$	$(1, 3)$	$1217, 616, 307$
	$q^{\pm}_{\text{wh}}$	-	1583
FINER-D	$q^{\pm}_{\text{multi-obj}}$	$(1, 6)$	$65, 496, 909, 980, 874, 1676$
	$q^{\pm}_{\text{multi-attr}}$	$(1, 5)$	$2451, 5363, 3092, 1575, 1843$
	$q^{\pm}_{\text{multi-rel}}$	$(1, 3)$	$4404, 1168, 199$
	$q^{\pm}_{\text{wh}}$	-	10472

Post-hoc correction of MCQs. After constructing the MCQs for FINER-CompreCap and FINER-DOCCI, humans further corrected a subset of them: 100 MCQs per setting for FINER-CompreCap and 200 MCQs per setting for FINER-DOCCI. In the 3-relation subset of FINER-DOCCI, we additionally observed cases where multiple relations referred to the same objects. We therefore performed further human cleaning, resulting in 199 improved paired MCQs in this setting.

C. Training Details

Sec. 3 explains our training data generation pipeline, on which FINER-Tuning is trained. We also briefly describe the fine-tuning setup in Sec. 4.1. In this section, we first present concrete examples of the training data, and then provide the detailed fine-tuning configuration.

Training set examples. We apply the training data construction pipeline from Fig. 3 to the first 24 shards of Pixmo-caption [11]. As described in Sec. 3, each image x can yield up to eight preference tuples (x, q, a^+, a^-) across the four subsets $\{\text{Obj}, \text{Attr}, \text{Rel}, \text{Wh}\}$. Applying the pipeline to 24 shards produces more than 1.6M preference tuples, which is more than we need for training. In practice, we only use the first 6 shards (about 440K tuples) and uniformly subsample at most 160K tuples for DPO training. We visualize representative training examples (x, q, a^+, a^-) from all four subsets in Fig. 10.

Finetuning Setup. We summarize the training hyperparameters for FINER-Tuning in Tab. 10. All models are trained with LLaMA-Factory [58], using LoRA [17] as the parameter-efficient fine-tuning method. We apply LoRA adapters only to the projection layers q_{proj} and v_{proj}. We reserve 0.5% of the data as a validation set. Since the validation distribution closely matches the training distribution, we observe that training for too long drives the validation loss close to zero and brings little or no performance gain, sometimes even degrading downstream results. For DPO training, we therefore limit the number of training samples for each model: LLaVA-1.6 is trained on 40K examples, Qwen2.5-VL on 120K, and the InternVL3.5 series on 160K. For the SFT experiments in Tab. 4, we fine-tune InternVL3.5-8B on 160K examples with a learning rate of 1×10^{-4}. We use 4 NVIDIA H100 94GB GPUs to train InternVL3.5-14B, and 2 NVIDIA H100 GPUs for the other smaller models.

Config	Llava-1.6-7B	Qwen2.5VL-7B	InternVL3.5-8B	InternVL3.5-14B
Training Data	40K	120K	160K	160K
Global BS	64			
Optimizer	AdamW [29]			
Learning rate	5×10^{-6}			
Total epochs	1			
Warm up ratio	0.1			
LR scheduler	cosine decay			
LoRA rank	32			
LoRA target	$q_{\text{proj}}, v_{\text{proj}}$			
β	0.1			
Val. ratio	0.005			

Table 10. Fine-tuning hyper-parameters for FINER-Tuning on all baselines. Global BS: global batch size. LR scheduler: learning rate scheduler. β: inverse temperature parameter in the DPO loss, as shown in Eq. 5. Val. ratio: ratio of validation data size.

D. Evaluation Details

We detail the evaluation setups for three groups of tasks: the FINER benchmarks, other hallucination benchmarks, and general capabilities.

FINER benchmarks. Since the FINER benchmarks are multiple-choice (MCQ) benchmarks, we evaluate all models using greedy decoding with temperature 0, no sampling, and a maximum of 3 output tokens. Given an image and an MCQ, we append the instruction: ``Please answer with a single capital letter (A, B, C, D, or E).'' We compute the paired accuracy $\text{Acc}_{\text{paired}}$, which counts a pair as correct only if the model answers both q^+ and q^- correctly, ensuring that the model does not systematically favor either the positive or the negative version.

Other hallucination benchmarks. We evaluate all models on both discriminative hallucination benchmarks (DASH [3], POPE [22], RePOPE [33], HallusionBench [16], AMBER [44], CRPE_R [45]) and generative hallucination benchmarks (MMHal-Bench [40], HaloQuest [47]).

We use VLMEvalKit [14] to evaluate HallusionBench, AMBER, and CRPE_R with their default configuration. We report all accuracy (aAcc.) for HallusionBench and averaged accuracy for CRPE_R. For DASH, POPE, and RePOPE, we follow their official evaluation protocols and prompt models to answer only with ``yes'' or ``no''.

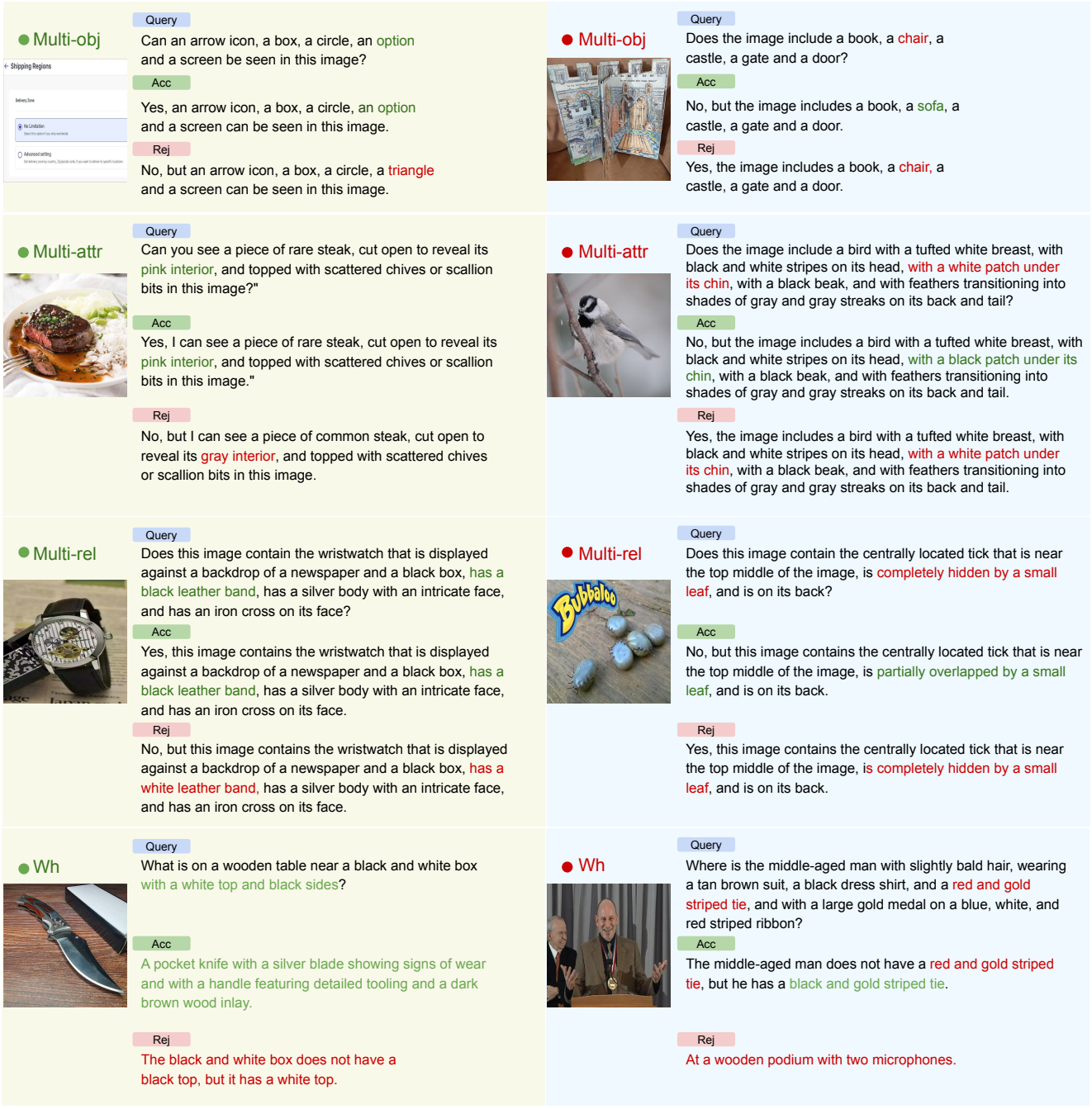

Figure 10. Examples from our constructed training set to train FINER-Tuning. Positive queries are in green color, while negative queries are in red color. We show both positive $((x, q_+, a_+^+, a_+^-))$ and negative $((x, q_-, a_-^+, a_-^-))$ preference tuples across four subsets: Multi-obj, Multi-attr, Multi-rel, Wh.

We again adopt greedy decoding for this binary setting to keep the setup consistent across models. We report the averaged accuracy in Tab. 2 and show the accuracy on each subset in Tab. 13.

For MMHal-Bench, we use the original evaluation code but replace the judge model with GPT-4.1-mini [2], since the original judge has been deprecated. For HaloQuest, we similarly follow the released evaluation pipeline but replace the judge with Gemini-2.0-Flash [41], as Gemini-1.5-Pro is no longer accessible. In both generative benchmarks, we

use temperature 0 to ensure reproducible results. We follow the metrics of both benchmarks, reporting score (max. 6) as well as hallucination rate in MMHal-Bench, as well as the averaged score in HaloQuest.

General capabilities. We evaluate general capabilities using six benchmarks: MMStar [7] (broad multi-skill evaluation), TextVQA [39] (text understanding from images), ChartQA [32] (chart and figure understanding), MMVP [42] (vision-centric reasoning), NaturalBench [21] (natural, compositional multi-step reasoning), and V* (visual search on high-resolution images). NaturalBench contains grouped, real-world questions that require models to jointly use perception, world knowledge, and compositional reasoning, making it a challenging test of robust, general-purpose vision-language ability.

We use VLMEvalKit [14] with default settings to evaluate all models on these six benchmarks. We report overall accuracy for MMStar, TextVQA, ChartQA, MMVP, and V*. For NaturalBench, we report group accuracy (G_ACC), as it is the most stringent and informative metric.

Table 11. Category statistics for Pixmo-caption [11].

Category	Count	Percentage
natural_image	176,881	78.13%
screenshot_ui	36,701	16.21%
chart_graph	8,061	3.56%
document_text	4,739	2.09%

Table 12. Filtering to only keep natural images ablation for FINER-Tuning with InternVL-3.5-8B [46]. Obj/Attr/Rel denote Multi-obj/Multi-attr/Multi-rel for both training and evaluation. The best results are bold.

Fitered?	FINER-CompreCap				Other	
	Obj	Attr	Rel	Wh	RePOPE	M.S.
-	74.2	71.9	49.8	25.5	91.5	68.0
Yes	**76.8**	**78.6**	62.8	**36.1**	**93.1**	68.1
No	76.5	78.3	**64.1**	**36.1**	**93.1**	**68.3**

E. Additional Experiments

Despite the main experimental results presented in Sec. 4, we report additional experiments in this section. Specifically, we conduct a positional bias study (Sec. E.1), analyze the impact of training data filtering (Sec. E.2), present more qualitative results from FINER-DOCCI (Sec. E.3), provide per-subset results of three benchmarks (Sec. E.4), provide an extended comparison with additional hallucination reduction methods (Sec. E.5), provide a brief discussion of an alternative random guess baseline (Sec. E.6), and show results on the MCQ version of our motivational study (Sec. E.7).

E.1. Positional bias study

Both FINER-CompreCap and FINER-DOCCI contain MCQs that involve multiple objects, attributes, and relations ($q^{\pm}_{\text{multi-obj}}$, $q^{\pm}_{\text{multi-attr}}$, and $q^{\pm}_{\text{multi-rel}}$). When constructing a negative MCQ q^{-}, we choose one entity (object, attribute, or relation) at a random position and replace it with its negative counterpart. A natural question is whether the model's behavior depends on which position is negated.

To test this, for all $q^{\pm}_{\text{multi-obj}}$, $q^{\pm}_{\text{multi-attr}}$, and $q^{\pm}_{\text{multi-rel}}$ with exactly three entities, we keep the same triplet but rotate which entity is negated, so that the negative appears once in each of the three positions. We then measure the paired accuracy $\text{Acc}_{\text{paired}}$ for each position. As shown in Fig. 6, base models exhibit clear positional bias. For example, in $q^{\pm}_{\text{multi-obj}}$, LLaVA-Next performs much worse when the negative is in the middle position, and Qwen2.5-VL-7B shows a drop of about 15% when the last position is negated compared to the first. In $q^{\pm}_{\text{multi-rel}}$, the preferred position even differs across models: InternVL3.5-8B achieves the highest accuracy when negating the middle entity, while InternVL3.5-14B peaks when the third entity is negated. Fine-tuning with FINER-Tuning consistently improves accuracy at all positions, but the curves are still not flat, indicating that positional bias remains. We suspect this is related to the inherent sequence structure of current MLLM architectures and leave a deeper investigation to future work. We also assume that the current MCQ format is not the best option for testing positional bias, and we are looking forward the community to dive deeper into language positional bias in open-ended generation questions.

E.2. Ablation: Training Data Filtering

In Pixmo-caption [11], we observed that certain amount of of images are charts/graphs or screenshots: content outside the evaluation scope of FINER-CompreCap and FINER-DOCCI (which target natural images). For example, one screenshot image can be found in the upper left corner of Fig. 10. Therefore, we first run Phi-4-14B over all the long captions to classify the images into four categories: "natural images", "screenshot_ui", "chart_graph" and "document_text". Since FINER benchmarks target only natural images. The statistics are in Tab. 12. Excluding these images resulted in almost no significant difference in performance. Therefore, to maintain simplicity and generality, we do not apply any filtering and retain the original dataset composition.

E.3. Qualitative Results

Following the qualitative results in Sec. 4.5 on FINER-CompreCap, we provide additional examples from

FINER-DOCCI in Fig. 11. These cases cover all four settings: Multi-obj, Multi-attr, Multi-rel, and Wh. We only visualize the negative MCQs here, as they are much more challenging than their positive counterparts. However, some positive MCQs can be found in our human study examples (Fig. 15 and Fig. 14).

As shown in Fig. 11, in the Multi-obj setting, only Gemini-2.5-Flash [10] and our FINER-Tuning-tuned InternVL3.5-14B reliably identify the fine-grained concept "macbook". In the Multi-attr setting, the questions target subtle details such as "the white note on the back driver's side window" or "the cat with perked-up ears". In the Multi-rel setting, some models, such as Qwen2.5-VL-7B [4], hallucinate the dog as being "behind the fence", even though it is clearly in front of the fence. Finally, in the Wh setting, only Gemini and FINER-Tuning correctly detect the anomalous attributes of the floor and the duck and answer the questions accordingly.

E.4. Per-subset results

POPE, RePOPE, AMBER. In Sec. 4.3, we report the averaged performance on POPE [22], RePOPE [33], and AMBER discriminative subset [44] (denoted as AMBER throughout this paper). In Tab. 13, we further break down the results and report the accuracy for each subset of these three benchmarks. Notably, with FINER-Tuning, LLaVA-1.6 achieves a 20.1% absolute improvement on AMBER, further demonstrating the effectiveness of FINER-Tuning.

HallBench, CRPE_R, HaloQuest. Apart from the per-subset results reported in Tab. 13, we further report detailed breakdowns for HallBench [16], CRPE_R [45] and HaloQuest [47] in Tab. 14. To further probe the captioning capabilities of different models, we include the results for AMBER generative subset (AMBER_G) and report four metrics: CHAIR (CH.), COVER (CO.), Hal. and Cog. in Tab. 14. On Hallbench, FINER-Tuning improves over all baselines by maximally 6.8% (fAcc. of LLaVA-1.6), showcasing that FINER-Tuning can still work effectively in reducing general halucinations. In HaloQuest, the performance gain is mainly in Insufficient Context (IC.) subset and false premise (FP.) subset. Some catchy improvements are: FINER-Tuning improves LLaVA-1.6 by 19.0% on IC and 31% on FP. FINER-Tuning also improves the latest InternVL-3.5-8B by 15.7% and 15.3% each. Note that HaloQuest is a free-form generative benchmark. This shows that FINER-Tuning can effectively correct the false premise hallucinations or withhold over-confident preidctions in free-form generations.

AMBER_G. To further probe the captioning capabilities of different models, we include the results for AMBER generative subset (AMBER_G) and report four metrics: CHAIR, COVER, Hal and Cog in Tab. 15. Lastly, FINER-Tuning consistently improves over three baselines (Qwen2.5-VL-7B, InternVL-3.5-8B, InternVL3.5-14B) on AMBER_G. We therefore think that when the base models are strong enough, FINER-Tuning can further improve the captioning capabilities of the model.

E.5. Comparing with more methods

It is challenging to totally fairly compare hallucination reduction methods because they are often trained on different datasets and base models. In this section, we fine-tune LLaVA-1.5-7B [26] with FINER-Tuning using 40K training examples from our dataset. We then evaluate on discriminative hallucination benchmarks (POPE [22], AMBER [44]) and generative benchmarks (MMHal-Bench (MMHal) [40] and HaloQuest [47]). We compare against the state-of-the-art REVERSE [49], as well as DoLA [9], HA-DPO [57], and HALVA [38]. We also compare FINER-Tuning with RLAIF-V-7B [55] on the same LLaVA-1.5-7B base model, resulting in a more direct comparison than Tab. 1 and Tab. 2. The results are in Tab. 16.

Using 40K training samples curated by Phi-4-14B [1], FINER-Tuning already achieves comparable performance on discriminative benchmarks to HALVA and HA-DPO, whose training data are curated by Gemini Vision Pro [41] and GPT-4 [2], respectively, while substantially outperforming them on generative benchmarks. Compared with the SOTA method REVERSE, FINER-Tuning matches or surpasses its performance on discriminative tasks and further improves HaloQuest by 6.3%, but still lags behind on MMHal-Bench. Overall, these results indicate that FINER-Tuning is effective at reducing hallucinations, and its benefits appear more pronounced when applied to stronger, frontier MLLMs, as also evidenced in Tab. 2. Compared to RLAIF-V, FINER-Tuning performs better on discriminative benchmarks such as POPE and AMBER (a +5.5% gain on AMBER), but remains weaker on generative benchmarks like MMHal-Bench.

E.6. Smarter random guess baselines

In Tab. 1, we report a uniform random-guess baseline of 4%, which corresponds to independently sampling one out of five answer options for both the positive and negative questions: $(1/5)^2$.

However, due to the structured answer space in our Multi-obj/Multi-attr/Multi-rel MCQs (one `Yes, I can see...` option and four `No, but I can see...` options), a stronger no-knowledge baseline is a *polarity-aware* random guesser. Specifically, it first guesses the polarity (`Yes` vs. `No`) uniformly, and if it guesses `No`, it then uniformly selects one of the four `No` options.

Since each pair consists of one positive question whose ground-truth is always `Yes` and one negative question whose ground-truth is always one of the four `No` options, the probability of guessing correctly is 0.5 for a positive

Multi-obj NEG

Can you see wall, croissant, desktop, headphones, and floor in this image?
A. No, but I can see wall, croissant, typewriter, headphones, and floor in this image.
B. Yes, I can see wall, croissant, desktop, headphones, and floor in this image.
C. No, but I can see wall, croissant, macbook, headphones, and floor in this image.
D. No, but I can see wall, croissant, calculator, headphones, and floor in this image.
E. No, but I can see wall, croissant, tablet, headphones, and floor in this image.

B ✗ B ✗ C ✓ B ✗ C ✓

Multi-obj NEG

Can you see cat, door, pillar, floor, puppy, and bag in this image?
A. No, but I can see cat, door, archway, floor, puppy, and bag in this image.
B. No, but I can see cat, door, baseboard, floor, puppy, and bag in this image.
C. Yes, I can see cat, door, pillar, floor, puppy, and bag in this image.
D. No, but I can see cat, door, door frame, floor, puppy, and bag in this image.
E. No, but I can see cat, door, banister, floor, puppy, and bag in this image.

C ✗ C ✗ D ✓ C ✗ D ✓

Multi-attr NEG

Can you see the sedan with a red poster on the front passenger's side window and with a silver color in this image?
A. No, but I can see the sedan with a green sign on the front passenger's side window and with a silver color in this image.
B. Yes, I can see the sedan with a red poster on the front passenger's side window and with a silver color in this image.
C. No, but I can see the sedan with a yellow flyer on the front passenger's side window and with a silver color in this image
D. No, but I can see the sedan with a white note on the back driver's side window and with a silver color in this image.
E. No, but I can see the sedan with a black advertisement on the front passenger's side window and with a silver color in this image.

A ✗ D ✓ D ✓ B ✗ D ✓

Multi-attr NEG

Can you see the cat with its tail end in the shadows, with ears missing entirely, and with a seated posture in this image?
A. No, but I can see the cat with its tail end in the shadows, with ears drooping down, and with a seated posture in this image.
B. No, but I can see the cat with its tail end in the shadows, with ears perked up, and with a seated posture in this image.
C.Yes, I can see the cat with its tail end in the shadows, with ears missing entirely, and with a seated posture in this image.
D.No, but I can see the cat with its tail end in the shadows, with ears twitching rapidly, and with a seated posture in this image.
E.No, but I can see the cat with its tail end in the shadows, with ears folded back, and with a seated posture in this image.

A ✗ B ✓ B ✓ C ✗ B ✓

Multi-rel NEG

Can you see the puppy that is jumping to the left of the chair and is looking upwards at the cat in this image?
A. No, but I can see the puppy that is running behind the chair and is looking upwards at the cat in this image.
B. No, but I can see the puppy that is standing on the right of the chair and is looking upwards at the cat in this image.
C. No, but I can see the puppy that is hanging above the chair and is looking upwards at the cat in this image.
D. No, but I can see the puppy that is lying at the base of the chair and is looking upwards at the cat in this image.
E. Yes, I can see the puppy that is jumping to the left of the chair and is looking upwards at the cat in this image.

A ✗ B ✗ D ✓ E ✗ D ✓

Multi-rel NEG

Can you see the puppy that is to the left of the fence in this image?
A. No, but I can see the puppy that is under the fence in this image.
B. No, but I can see the puppy that is behind the fence in this image.
C. Yes, I can see the puppy that is to the left of the fence in this image.
D. No, but I can see the puppy that is to the right of the fence in this image.
E. No, but I can see the puppy that is in front of the fence in this image.

C ✗ B ✗ E ✓ C ✗ E ✓

Wh NEG

What is laying on the floor with a green metal surface?
A. The poodle with a body facing the right side of the image, with a majority of the rest of its body with a black color, and with a collar.
B. The chihuahua with a body facing the right side of the image, with a majority of the rest of its body with a black color, and with a collar.
C. The Yorkshire Terrier with a body facing the right side of the image, with a majority of the rest of its body with a black color, and with a collar.
D. The dachshund with a body facing the right side of the image, with a majority of the rest of its body with a black color, and with a collar.
E. The floor is not with a green metal surface, but is with a brown wooden surface.

A ✗ C ✗ E ✓ C ✗ E ✓

Wh NEG

What is the duck with a red and green color, with its beak in the water, and with water dripping off it standing on?
A. The rocks with a pile formation
B. The shells with a pile formation.
C. The pebbles with a pile formation
D. The duck is not with a red and green color, but is with a black and white color.
E. The driftwood with a pile formation

A ✓ D ✗ D ✓ C ✗ D ✓

LLaVA-1.6-7B Qwen2.5-VL-7B Gemini-2.5-Flash InternVL3.5-14B InternVL3.5-14B w/ FINER-Tuning

Figure 11. Qualitative Results from FINER-DOCCI.

Models	Size	POPE			RePOPE			AMBER		
		Ran. ↑	Pop. ↑	Adv. ↑	Ran. ↑	Pop. ↑	Adv. ↑	Exis. ↑	Attr. ↑	Rel. ↑
OmniLMM	12B	89.3	87.8	87.1	95.1	93.2	93.1	85.6	94.2	80.7
+RLAIF-V	12B	89.0 0.3	87.5 0.3	86.8 0.3	95.0 0.1	92.8 0.4	92.6 0.5	86.1 0.5	90.2 4.0	85.7 5.0
LLaVA-1.6 [27]	7B	89.7	88.4	86.6	93.9	92.1	91.0	82.0	93.6	58.7
+FINER-Tuning	7B	90.4 0.7	88.8 0.4	87.2 0.6	94.9 1.0	92.9 0.8	91.8 0.8	83.5 1.5	92.6 1.0	78.8 20.1
Qwen2.5-VL [4]	7B	87.0	86.5	85.8	93.6	91.9	91.7	84.1	95.7	75.6
+FINER-Tuning	7B	88.0 1.0	87.0 0.5	86.4 0.6	94.1 0.5	92.2 0.3	91.9 0.2	84.0 0.1	96.2 0.5	77.1 1.5
InternVL-3.5 [46]	8B	93.3	87.7	85.0	95.4	90.7	88.5	80.4	88.0	80.1
+FINER-Tuning	8B	92.7 0.6	88.7 1.0	86.6 1.6	95.9 0.5	92.6 1.9	90.9 2.4	80.6 0.2	88.2 0.2	80.6 0.5
InternVL-3.5 [46]	14B	93.4	89.6	85.7	94.7	92.1	88.8	82.6	89.4	81.9
+FINER-Tuning	14B	93.0 0.4	90.2 0.6	87.3 1.6	95.8 1.1	93.6 1.5	91.4 2.6	82.5 0.1	91.0 1.6	81.5 0.4

Table 13. Per-subset results on POPE [22], RePOPE [33], and AMBER [44]. Rand.: Random; Pop.: Popular; Adv.: Adversarial; Exis.: Existence; Attr.: Attribute; Rel.: Relation

Models	HallBench			CRPE_R				HaloQuest		
	aAcc. ↑	fAcc. ↑	qAcc. ↑	Sub. ↑	Pred. ↑	Obj. ↑	Tot. ↑	VC. ↑	IC. ↑	FP. ↑
LLaVA-1.6-7B	33.0	10.6	8.3	61.7	52.6	61.6	56.5	50.5	38.0	42.9
+FINER-Tuning	36.3 3.3	17.4 6.8	13.0 4.7	62.6 0.9	51.7 0.9	59.8 1.8	56.0 0.5	50.5	57.0 19.0	73.9 31.0
Qwen2.5-VL-7B	65.4	35.8	40.0	77.2	66.1	71.7	69.9	66.5	76.0	79.2
+FINER-Tuning	68.5 3.1	40.0 4.2	43.6 3.6	77.9 0.7	67.0 0.9	72.4 0.7	70.7 0.8	65.9 0.6	86.7 10.7	87.5 8.3
InternVL-3.5-8B	71.0	45.1	47.0	75.6	63.3	70.8	67.7	66.5	51.2	64.4
+FINER-Tuning	73.0 2.0	48.9 3.8	49.3 2.3	76.5 0.9	63.4 0.1	70.9 0.1	68.0 0.3	65.9 0.6	66.9 15.7	80.7 15.3
InternVL-3.5-14B	69.5	46.8	47.0	77.2	60.7	73.3	67.1	63.7	54.5	70.0
+FINER-Tuning	71.2 1.7	49.2 2.4	49.7 2.7	78.5 1.3	63.1 2.4	73.9 0.6	68.9 1.8	63.7	61.2 6.7	79.2 9.2

Table 14. Per-subset results on HallBench [16], CRPE relation subset (CRPE_R) [45], and HaloQuest [47]. Sub.: Subject; Pred.: Predicate; Obj.:Object; Tot.: Total; VC.::Visually Challenge subset; IC.: Insufficient Context subset; FP.: False Premise subset;

Models	AMBER_G			
	CHAIR ↓	COVER ↑	Hal ↓	Cog ↓
Qwen2.5-VL-7B	5.3	64.0	27.1	1.9
+FINER-Tuning	5.0 0.3	64.7 0.7	25.9 1.2	1.6 0.3
InternVL-3.5-8B	6.9	61.3	49.9	3.1
+FINER-Tuning	6.3 0.6	61.4 0.1	47.0 2.9	2.5 0.6
InternVL-3.5-14B	7.9	68.6	57.6	5.4
+FINER-Tuning	7.4 0.5	68.7 0.1	54.4 3.2	4.4 1.0

Table 15. Extended results on AMBER generative subset (AMBER_G).

Method	POPE	AMBER	MMHal	HaloQuest
	Acc. ↑	Acc. ↑	HR. ↓	Score ↑
LLaVA-1.5-7B	85.9	74.7	54.0	22.6
+HALVA [38]	84.8	**83.4**	54.0	23.9
+HA-DPO [57]	**86.9**	78.1	60.0	-
+DoLA [9]	85.7	74.5	56.0	22.9
+RLAIF-V [55]	85.2	76.8	32.3	-
+REVERSE [49]	85.9	74.2	**30.0**	32.3
+FINER-Tuning	86.7	82.3	49.0	**38.8**

Table 16. Extended comparison with other hallucination reduction methods on LLaVA-1.5-7B [26]. HR.: Hallucination rate. The best results are bold while the second best results are underlined.

MCQ. For the negative MCQ, it is 0.5×0.25. Therefore, the paired accuracy is $0.5 \times (0.5 \times 0.25) = 0.0625$.

E.7. MCQ Version of the Motivational Study

Yes/no probing is standard in prior benchmarks such as DASH, POPE, and AMBER for evaluating *false-positive hallucinations*. In the main paper, we adopt this simple setup for the motivational study because it is easy to understand. In contrast, our FINER benchmarks are evaluated using multiple-choice questions (MCQs). Using two different evaluation protocols may cause confusion for some readers. Therefore, we additionally reformulate the motiva-

FINER-CompreCap				FINER-DOCCI			
Multi-obj	Multi-attr	Multi-rel	Wh	Multi-obj	Multi-attr	Multi-rel	Wh
92.5	92.5	97.5	95.0	92.5	95.0	90.0	90.0

Table 17. Human performance in paired accuracy ($\text{Acc}_{\text{paired}}$) on FINER-CompreCap and FINER-DOCCI.

tional study in the same MCQ format as used in our benchmarks. Fig. 12 shows the same trend as the yes/no version in the main paper: accuracy decreases as query granularity increases. More specifically, the false-positive (FP) rate is much higher than the false-negative (FN) rate, confirming that false-positive hallucination is the main cause of the performance drop.

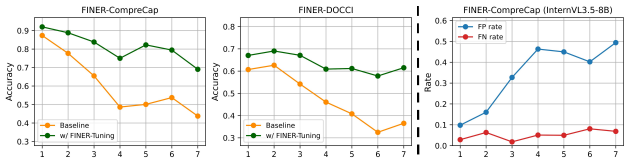

Figure 12. Left: MCQ version of the motivational study. Right: False-positive (FP) and false-negative (FN) rates at each granularity level.

F. Human Study

Since the FINER benchmarks are text-intensive, we asked human participants to answer a limited number of questions: 20 MCQs per subset. With eight subsets in total (four from FINER-CompreCap and four from FINER-DOCCI), this yields 160 MCQs. The results are shown in Tab. 17.

Unlike models, which answer the positive and negative versions of each MCQ independently, humans could in principle remember a MCQ and use the correspondence between q^+ and q^- to make the task easier. To avoid this, we create two versions (A and B) for each setting. For every MCQ pair, the positive and negative versions are randomly assigned to different versions. Each annotator only sees one version (either A or B), so they never see both sides of the same pair.

We recruit four human participants for each setting and compute paired accuracy based on their responses. The numerical results are reported in Tab. 1. Example survey pages from our human study are shown for Multi-rel and Wh questions from FINER-CompreCap in Fig. 14, and for Multi-obj and Multi-attr questions from FINER-DOCCI in Fig. 15. As illustrated in these figures, each MCQ has two versions (A and B), corresponding to its positive and negative forms, and no annotator ever answers both versions of the same MCQ.

Success and failure cases. As Tab. 17 shows, humans achieve over 90% paired accuracy across all settings in FINER-CompreCap and FINER-DOCCI. Although we can only evaluate human performance on a limited subset due to resource constraints, we do observe many cases where humans succeed on MCQs that a model like InternVL-3.5-14B [46] fails on. Notably, there are also MCQs where humans fail but models succeed. Representative success and failure cases are shown in Fig. 13.

From Fig. 13, human errors can be grouped into two main types: carelessness and ambiguity. In the upper-right example, the human selects "sleeping behind the window", likely due to a simple oversight or a "yes" bias, similar to how InternVL-3.5-14B fails in the lower-right example. The second type of error arises from subjective or ambiguous visual attributes. In the dog example, the human chooses "with bald ears that flap sideways" instead of "with floppy ears that hang down". This is partly understandable, since "flap sideways" describes some of the observed motion even though the ears are not truly "bald". Strictly speaking, "bald ears that flap sideways" should be considered a false attribute (only partially correct), especially when compared to "floppy ears that hang down" (correct).

This motivates our choice to design FINER as an MCQ benchmark rather than using simple yes/no questions. By comparing multiple options, both humans and models are encouraged to pick the better description, which reduces ambiguity to some extent. Nevertheless, even with our entropy-based filtering pipeline, additional human verification, and MCQ design, the scale of FINER means that a certain amount of subjectivity, ambiguity, and annotation errors in the descriptions remains unavoidable. A valid future direction is to construct FINER benchmarks fully with human annotations, better aligning the evaluation with human subjectivity in assessing hallucinations.

In our human studies, participants answer 20 MCQs per subset, which is small relative to the scale of both benchmarks. This is mainly because FINER is highly text-intensive, requiring substantial reading time. Scaling up the human study would likely further reduce human accuracy due to the reading burden and potential noise, since the benchmark is not fully created and validated by humans. We therefore treat the limited scale of the human studies as a limitation, and emphasize that these results only reflect human behavior on a small subset and given ample answering time, rather than serving as a valid measure of overall benchmark quality.

G. Templates

To construct training set for FINER-Tuning. Sec. 3 describes how we run Phi-4-14B [1] over captions to extract

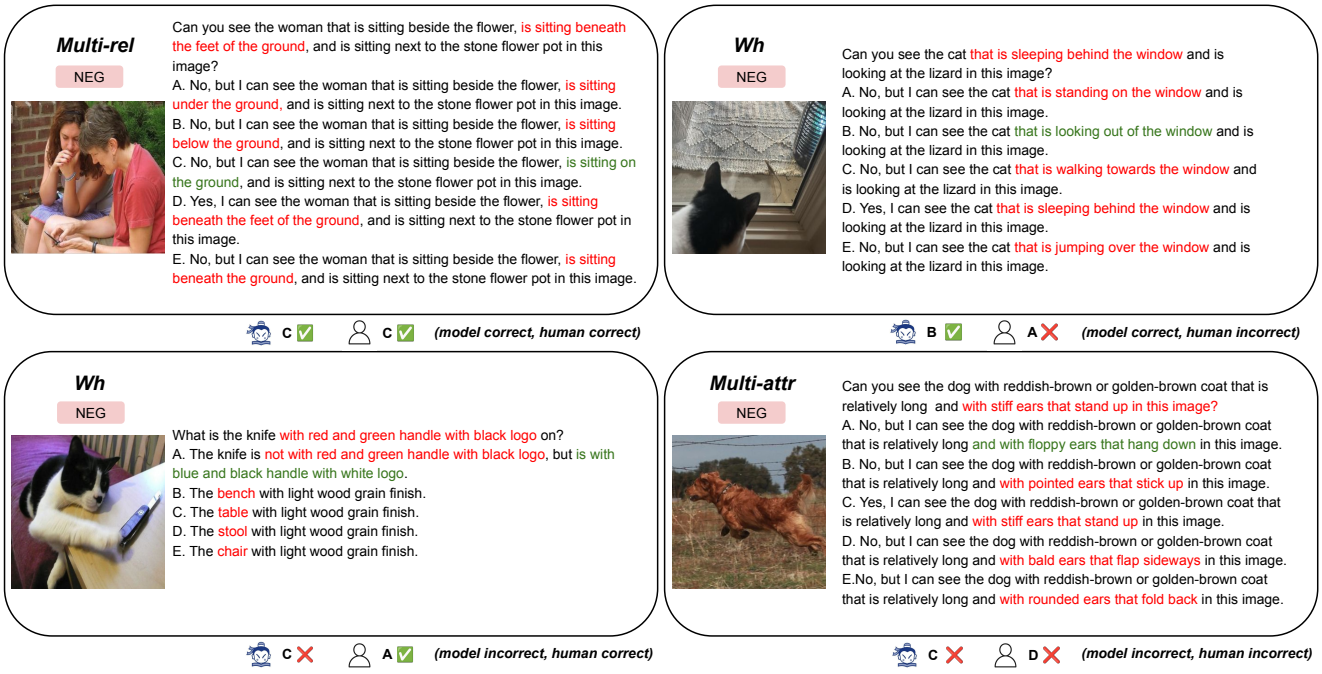

Figure 13. Success & failure analysis matrix for InternVL3.5-14B [46] (denoted as "model" in the figure) and Human. All MCQs are included in the human study.

positive phrases

$$\{\Psi^+_{\text{OBJ}},\ \Psi^+_{\text{ATTR}},\ \Psi^+_{\text{REL}},\ \Psi^+_{\text{WH}}\}$$

and negative phrases

$$\{\Psi^-_{\text{OBJ}},\ \Psi^-_{\text{ATTR}},\ \Psi^-_{\text{REL}},\ \Psi^-_{\text{WH}}\}.$$

OBJ / ATTR / REL. For OBJ, ATTR, and REL, we first extract positive phrases $\{\Psi^+_{\text{OBJ}},\ \Psi^+_{\text{ATTR}},\ \Psi^+_{\text{REL}}\}$ using the prompts shown in Fig. 16, Fig. 17, and Fig. 18. We then prompt the same LLM to generate the corresponding negative phrases $\{\Psi^-_{\text{OBJ}},\ \Psi^-_{\text{ATTR}},\ \Psi^-_{\text{REL}}\}$ with the prompts in Fig. 20, Fig. 21, and Fig. 22. Given these positive/negative phrase sets, we construct preference tuples

$$(q^+, a^+_+, a^-_+) \quad \text{and} \quad (q^-, a^+_-, a^-_-)$$

for each of OBJ, ATTR, and REL via template-based composition, by using a pool of five templates as below:

```
(1)
Does this image contain {X}?
Yes, this image contains {Y}.
No, but this image contains {Z}.
(2)
Does this image show {X}?
Yes, this image shows {Y}.
No, but this image shows {Z}.
(3)
Does this image include {X}?
Yes, this image includes {Y}.
No, but this image includes {Z}.
(4)
Can you see {X} in this image?
Yes, I can see {Y} in this image.
No, but I can see {Z} in this image.
(5)
Can {X} be seen in this image?
Yes, {Y} can be seen in this image.
No, but {Z} can be seen in this image.
```

To avoid overfitting to a single fixed pattern and to stay consistent with the FINER benchmarks, we randomly choose one of the above five templates for each example. Each template contains placeholders $\{X\}$, $\{Y\}$, and $\{Z_1\}, \ldots, \{Z_4\}$ that are filled with phrases.

In the positive configuration (q^+, a^+_+, a^-_+), the "Yes" an-

swer will be the accepted response a_+^+ while the "No" answer will be the rejected response a_+^-. The question and the "Yes" answer both use the positive phrase Ψ^+, while all "No" answers use the negative phrase Ψ^-:

$$\{X\} = \Psi^+,$$
$$\{Y\} = \Psi^+,$$
$$\{Z\} = \Psi^-$$

In the negative configuration (q^-, a_+^-, a_-^-), the "No" answer will be the accepted response a_-^+ while the "Yes" answer will be the rejected response a_-^-. The question and all "No" answers use the negative phrase Ψ^-, while the "Yes" answer uses the positive phrase Ψ^+:

$$\{X\} = \Psi^-,$$
$$\{Y\} = \Psi^+,$$
$$\{Z\} = \Psi^-$$

WH. For WH, the preference tuples

$$(q^+, a_+^+, a_+^-) \quad \text{and} \quad (q^-, a_-^+, a_-^-)$$

are directly constructed by the LLM, rather than via our fixed templates. We therefore do not apply the above template-based composition to WH, and instead use dedicated prompts to let the LLM generate the question and its positive/negative answers. The prompts used to construct a pair of (q^+, a_+^+) and (q^-, a_-^+) for WH are shown in Fig. 19 and Fig. 23, respectively. Concretely, the LLM first produces two Wh questions about the same underlying scene: a positive question q^+, whose premise is consistent with the image and whose accepted response a_+^+ directly answers what the question asks for, and a negative question q^-, whose premise partially conflicts with the image content so that its accepted response a_-^+ explicitly negates the question itself. We then symmetrize this pair by assigning each accepted response as the other question's rejected response, i.e., $a_+^- := a_-^+$ and $a_-^- := a_+^+$. In this way we obtain the final preference tuples (q^+, a_+^+, a_+^-) and (q^-, a_-^+, a_-^-).

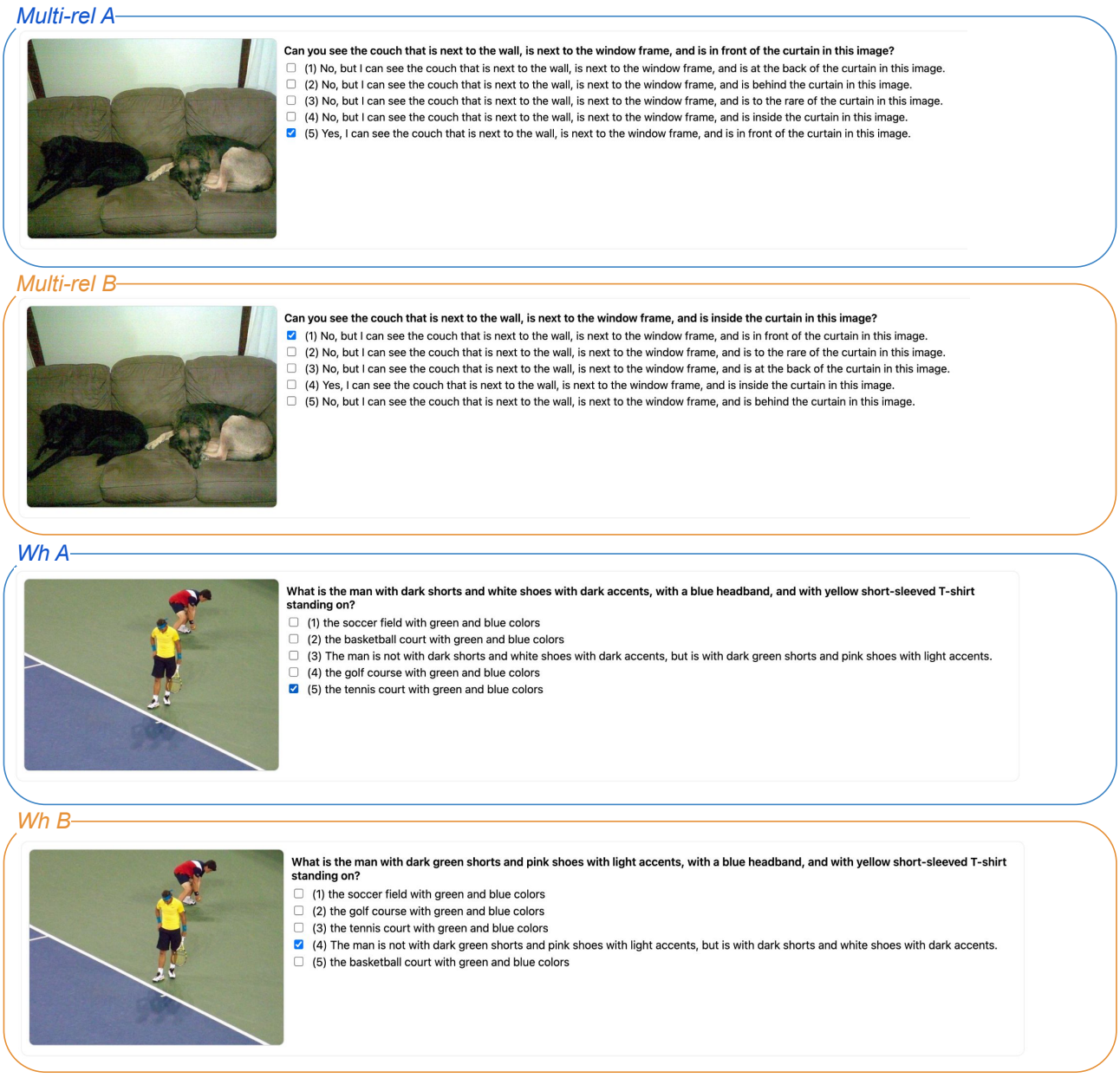

Figure 14. Examples of our human study survey for FINER-CompreCap. Example questions from Multi-rel and Wh are shown in the figure. Ticked boxes represent ground-truth choices. We use blue color to represent the questions for version A, while orange representing the questions for version B.

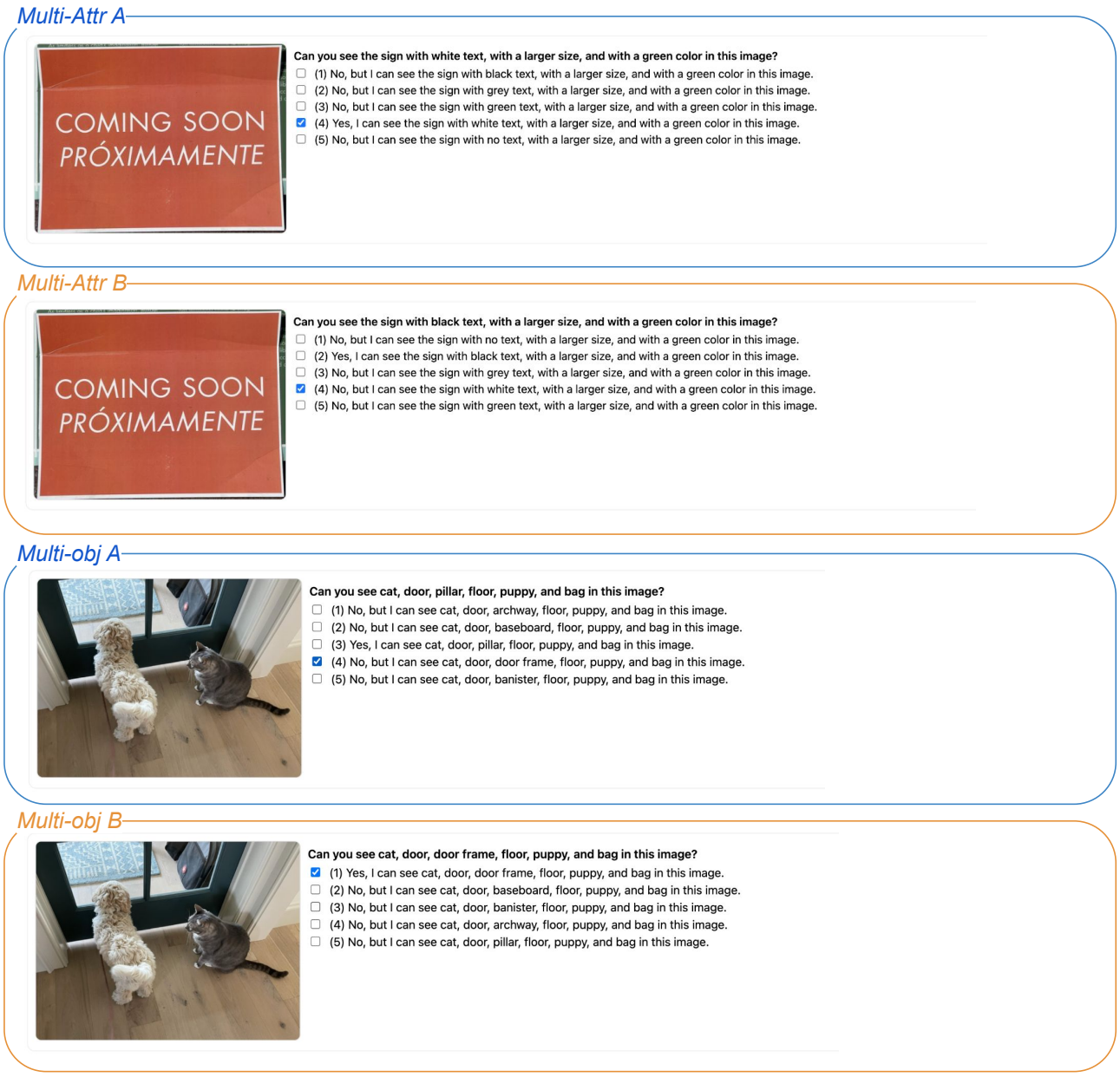

Figure 15. Examples of our human study survey for FINER-DOCCI. Example questions from Multi-attr and Multi-obj. Ticked boxes represent the ground-truth choice. We use blue color to represent the questions for version A, while orange representing the questions for version B.

Prompt template for extracting Ψ^{+}_{OBJ}

```
"You are an information extraction assistant.\n"
   "From the caption, select up to FIVE main objects. A main object must have at least one descriptive attribute in the caption "
   "(e.g., color, size, material, possession/with-phrase, appositive, relative clause, or an explicit number).\n"
   "Rules:\n"
   "• Output only object names that appear in the caption and are part of the described scene. Never invent or infer.\n"
   "• Prefer plain object names in the output (omit adjectives), but KEEP explicit numbers/quantifiers if the caption states them "
   "(e.g., 'two cats', 'five chickens', 'a pair of skis').\n"
   "• If multiple mentions share the same name and no explicit number is given, output the plural form (e.g., 'dogs'). \n"
   "• List 1–5 main objects; if only one is present, output just that one.\n"
   "Do not add quantifiers like 'some' unless present in the caption.\n"
   "Format:\n"
   "Return EXACTLY one line:\n"
   "PHRASE=<comma-separated list with 'and' before the last item, e.g., 'a dog, a cat and two birds'>\n"
   "No trailing period. No extra text."
```

Figure 16. Prompt Template for extracting Ψ^{+}_{OBJ}

Prompt template for extracting Ψ^{+}_{ATTR}

```
"You are an information extraction assistant.\n"
   "Select ONE main object from the caption that has at least one described attribute.\n"
   "Extract attribute phrases ONLY if they are explicitly stated and are used to describe the chosen main object—never infer or guess.\n"
   "Then compose a SINGLE noun phrase describing that object with the extracted attribute phrases.\n"
   "Use ONLY evidence from the caption. Never invent attributes.\n"
   "Allowed attribute types:\n"
   "• Appearance, color, pattern, size, shape, material, markings/printed text/numbers, "
   "condition/state, orientation/pose, and other visible features that describe the main object.\n"
   "• Accessories physically attached to the main object (e.g., a collar on a dog) count as attributes; unrelated co-occurring objects do not.\n"
   "Constraints:\n"
   "• Do NOT include spatial relations to other main objects (e.g., 'to the left of the bus').\n"
   "• Do NOT include actions involving other main objects (e.g., 'holding a cup').\n"
   "• The extracted attributes must clearly describe the chosen main object. NEVER invent attributes. NEVER extract attributes for the other
objects.\n"
   "• Extract 1–5 attributes for the chosen main object. If fewer than five are stated, extract fewer. If only one is present, use that one. NEVER
invent attributes.\n"
   "• Optionally, rewrite the original attribute phrase to either a plain adjective phrase (e.g., 'red', 'shiny metal', 'long-tailed'), "
   "or a 'with ...' phrase (e.g., 'with yellow eyes', 'with its nose pointing to the left', 'with the text \"SALE\"'). The rewriting should not
change the original meaning.\n"
   "• Connect multiple 'with ...' phrases smoothly using commas and 'and' (e.g., 'with yellow eyes, with a striped tail, and with a scar').\n"
   "Return EXACTLY one line:\n"
   "PHRASE=<your noun phrase with attributes>\n"
   "No trailing period. No extra text."
```

Figure 17. Prompt Template for extracting Ψ^{+}_{ATTR}

Prompt template for extracting Ψ^{+}_{REL}

```
"You are an information extraction assistant.\n"
   "Select ONE main object from the caption that clearly participates in at least one relation with another object.\n"
   "Extract relations ONLY if they are explicitly stated in the caption—never infer or guess.\n"
   "Allowed relation types:\n"
   "• Spatial: e.g., 'behind X', 'in front of Y', 'on Z', 'under W', 'next to Q', 'between A and B', 'near C', 'inside D', 'at E'.\n"
   "• Action with a target: verb phrases that take an object, e.g., 'holding a cup', 'biting a bone', 'looking at the door'.\n"
   "Constraints:\n"
   "• Every relation must involve the chosen object.\n"
   "• Refer to other objects with plain nouns; add attributes only to disambiguate same-named objects.\n"
   "• Use ONLY what the caption states; do NOT invent relations.\n"
   "• List 1–5 relations; if only one is present, output just that one.\n"
   "Compose ONE fluent phrase that starts with the object and then lists the relations.\n"
   "Prefer: 'The <object> is <relation1>, is <relation2>, ... and is <relationN>'.\n"
   "Return EXACTLY one line:\n"
   "PHRASE=<your single phrase with relations>\n"
   "No trailing period. No extra text."
```

Figure 18. Prompt Template for extracting Ψ^{+}_{REL}

Prompt template for generating q^{+}, a^{+}_{+}

```
"You create one WH-style QA pair from ONE sentence describing two main objects and their explicit relation, "
   "optionally with attributes. The sentence has the logical structure:\n"
   "[obj_a] [attr_a…] [rel] [obj_b] [attr_b…].\n"
   "\n"
   "Your task (A-mode):\n"
   "• Choose [obj_a][attr_a…] as the exact answer span.\n"
   "• Write ONE natural WH question whose answer is exactly that span.\n"
   "• In the QUESTION, preserve as much of [rel][obj_b][attr_b…] as natural, quoted verbatim when it fits, "
   "  and DO NOT repeat or paraphrase [obj_a][attr_a…] inside the question.\n"
   "• Be fluent and grammatical; do not invent details.\n"
   "Output EXACTLY one line:\n"
   "Q=<your question> || A=<the exact substring answer>\n"
   "No extra text."
```

```
"You create one WH-style QA pair from ONE sentence describing two main objects and their explicit relation, "
   "optionally with attributes. The sentence has the logical structure:\n"
   "[obj_a] [attr_a…] [rel] [obj_b] [attr_b…].\n"
   "\n"
   "Your task (B-mode):\n"
   "• Choose [obj_b][attr_b…] as the exact answer span.\n"
   "• Write ONE natural WH question whose answer is exactly that span.\n"
   "• In the QUESTION, preserve as much of [obj_a][attr_a…] and [rel] as natural, quoted verbatim when it fits, "
   "  and DO NOT repeat or paraphrase [obj_b][attr_b…] inside the question.\n"
   "• Be fluent and grammatical; do not invent details.\n"
   "Output EXACTLY one line:\n"
   "Q=<your question> || A=<the exact substring answer>\n"
   "No extra text."
```

Figure 19. Prompt Template for generating (q^{+}, a^{+}_{+}) for WH setting

Prompt template for generating Ψ^{-}_{OBJ}

```
"You are a negative object creator.\n"
   "You will receive a caption, an object list as PHRASE=..., and REPLACE_INDEX=k (1-based).\n"
   "Replace EXACTLY the k-th object in PHRASE with a distinctly different NEGATIVE object.\n"
   "\n"
   "Keep ALL other objects unchanged and preserve their order and punctuation. Keep the same quantifier/determiner "
   "for the replaced slot (e.g., 'two cats' -> 'two bicycles').\n"
   "\n"
   "Constraints for the NEGATIVE object:\n"
   "• It must be distinctly different from the replaced object (not a synonym; not just singular/plural).\n"
   "• It must NOT be a synonym or near-equivalent of ANY object that appears in the caption.\n"
   "• It must NOT appear anywhere in the caption (as a whole word, singular or plural).\n"
   "• Do not modify any other items; do not reorder items; do not add or remove items.\n"
   "\n"
   "Edge cases (must follow):\n"
   "• If REPLACE_INDEX is greater than the number of objects in PHRASE, replace the LAST object.\n"
   "• If REPLACE_INDEX is less than 1, replace the FIRST object.\n"
   "\n"
   "Self-check (must hold):\n"
   "• Same number of items as input; exactly one item (the k-th per the rule above) differs.\n"
   "\n"
   "Output EXACTLY one line:\n"
   "PHRASE=<the new list, same format>\n"
   "No extra text. No quotes. No trailing period."
```

Figure 20. Prompt Template for generating Ψ^{-}_{OBJ}

Prompt template for generating Ψ^{-}_{ATTR}

```
"You are a negative attribute editor.\n"
   "You will receive an ATTRIBUTE PHRASE: a single noun phrase describing one main object with 1–5 attributes.\n"
   "Each attribute is one replaceable unit: either (a) a pre-nominal adjective group (e.g., 'long-sleeved red') "
   "or (b) one entire 'with ...' clause or other forms of clause separated by commas or 'and'.\n"
   "\n"
   "Task:\n"
   "Pick exactly ONE attribute unit at random and replace it with a distinctly different NEGATIVE attribute.\n"
   "\n"
   "Randomness:\n"
   "• Replace the attribute unit at random position. Both pre-nominal adjective group or 'with ...' clause should have a chance to be replaced. \n"
   "\n"
   "Definitions & scope of attributes that can be changed:\n"
   "• Appearance, color, pattern, size, shape, material, texture, markings/printed text/numbers, "
   "condition/state, orientation/pose, and accessories physically attached to the main object.\n"
   "\n"
   "Constraints for the replacement:\n"
   "• Keep the object head and all other attributes unchanged; preserve order, punctuation, articles, quotes, units, and capitalization.\n"
   "• Keep the grammatical shape of the replaced unit (adjective group stays an adjective group; a 'with ...' clause stays a 'with ...' clause).\n"
   "• The replacement must be distinctly different from the original and NOT a synonym, near-synonym, or morphological variant of any attribute in
the phrase\n"
   "• Do not duplicate any existing attribute already present in the phrase.\n"
   "• Avoid always changing the same type of attribute; consider changing any types of attributes stated in the definitions above.\n"
   "\n"
   "Self-check before answering (must be satisfied):\n"
   "• Exactly one attribute unit differs; all other attribute units are identical.\n"
   "\n"
   "Output EXACTLY one line:\n"
   "PHRASE=<the rewritten noun phrase>\n"
   "No extra text. No quotes. No trailing period."
```

Figure 21. Prompt Template for generating Ψ^{-}_{ATTR}

Prompt template for generating Ψ^{-}_{REL}

```
"You are a negative relation editor.\n"
   "Input format:\n"
   "  CLAUSE_INDEX=<1-based index to edit>\n"
   "  PHRASE=The <HEAD> <clause1>, <clause2>, ... and <clauseN>\n"
   "Each clause is a relation expressed as a verb + complement, e.g., "
   "'is on a table', 'are between two cars', 'has a transparent faceplate', "
   "'holds a bottle', 'wears a red jersey', 'faces left', 'shows a temperature above 50 degrees'.\n"
   "\n"
   "Task:\n"
   "Select and edit EXACTLY the clause with the given CLAUSE_INDEX (1-based) to make it a clearly different (ideally opposite) NEGATIVE relation.\n"
   "\n"
   "Style guidance (choose ONE option to edit the selected clause):\n"
   "  (A) If the selected clause encodes a spatial relation via a preposition or comparator "
   "(e.g., in/on/inside/outside/under/over/above/below/behind/in front of/"
   "to the left of/to the right of/between/near/at/surrounding/is surrounded by/on top of/at the bottom of, etc.), replace that spatial term with"
   "opposite or distinctly different spatial relation (e.g., on→inside, in→out of, left→right, above→below, beside→inside). "
   "  (B) If the clause describes an action of the HEAD, replace this action with one distinctly different or opposite. Change the clause's main
lexical verb "
   "(e.g., holds→drops, wears→removes, shows→hides, opens→closes, runs→stands). You may also adjust adverbs or prepositions if any "
   "('is standing on'→'is running away from', 'is driving slowly to'→'is flying high from'). Preserve tense/number/aspect and auxiliaries "
   "(e.g., 'is holding'→'is dropping', 'has opened'→'has closed').\n"
   "  (C) If the selected clause describes possession or properties of the HEAD "
   "(e.g., has/have…, is/are made of…, shows/displays/reads/contains/wears…), "
   "replace the complement with something clearly different or opposite (e.g., 'contains two plastic bags'→'contains three paper bags').\n"
   "\n"
   "Hard constraints (must follow):\n"
   "• If the CLAUSE_INDEX is larger than the number of clauses you see, edit the LAST clause.\n"
   "• Keep the HEAD EXACTLY as in the input.\n"
   "• Keep ALL other clauses unchanged; preserve separators (commas and the final 'and').\n"
   "• Do NOT reorder clauses.\n"
   "• Edit ONLY the selected clause; do NOT add/remove clauses; the edited clause MUST be distinctly different from the original clause.\n"
   "• Avoid merely inserting 'not'; prefer concrete lexical or complement changes.\n"
   "• The new clause must not duplicate another clause and should remain grammatical (tense/number agreement intact).\n"
   "\n"
   "Output EXACTLY one line:\n"
   "PHRASE=<rewritten phrase>\n"
   "No extra text. No trailing period."
```

Figure 22. Prompt Template for generating Ψ^{-}_{REL}

Prompt template for generating q^-, a^+_-

```
"You will convert a POSITIVE wh-question into a counterfactual, NEGATIVE wh-question + answer by replacing EXACTLY ONE "
   "ATTRIBUTE CLAUSE that describes the main object mentioned in the question.\n"
   "\n"
   "DEFINITIONS (apply to the input question):\n"
   "• Main object: the plain head noun phrase that the attributes modify (e.g., 'a mug', 'the DSLR camera'). If multiple objects present in the
question, pick the one with more attributes as the main object.\n"
   "• Attribute clause: a modifier that directly describes the main object. It can be\n"
   "  – pre-nominal adjectives (color, material, pattern, size, shape, quantity), e.g., 'red', 'ceramic', 'wide'.\n"
   "  – post-nominal phrases (e.g., 'with …', 'featuring …', 'bearing …', 'labeled \"…\"', participial phrases like 'wearing …').\n"
   "  – other short descriptors attached to the object (texture, condition/state, orientation/pose, printed text/numbers).\n"
   "• Relation clause: the words expressing spatial or action relations that position the main object relative to something else, "
   "  e.g., 'on', 'under', 'next to', 'in front of', 'behind', 'to the left of', 'below', 'above', or light-verb forms like "
   "  'is on', 'is next to', 'is holding', 'is below'.\n"
   "\n"
   "To help you better identify the attribute clauses, the input questions are usually in the following forms:\n"
   "  – WH + [main object + attribute clauses] + [relation clause]?\n"
   "  – WH + [relation clause] + [main object + attribute clauses]?\n"
   "Note that the attribute clauses can either be pre-nominal (before the main object) or post-nominal (after the main object).\n"
   "\n"
   "EDIT RULES:\n"
   "1) Identify all attribute clauses attached to the main object you pick.\n"
   "2) Randomly choose ONE attribute clause (denoted as [original attribute]) and replace its content with a CLEARLY DIFFERENT or even OPPOSITE
attribute clause (denoted as [new attribute]).\n"
   "   • You may change multiple adjectives INSIDE [original attribute] to increase contrast.\n"
   "   • Do NOT add, remove, or reorder other attribute clauses—only replace the contents of the chosen [original attribute clause].\n"
   "3) Keep everything else unchanged:\n"
   "   • Do NOT change the main object.\n"
   "   • Do NOT change the relation clause.\n"
   "4) If the question truly has no attribute clauses for the main object, output exactly: SKIP\n"
   "\n"
   "RANDOMNESS:\n"
   "You MUST choose one attribute clause at random position. Both the attribute clauses before or after the main object should have a chance to be
chosen.\n"
   "\n"
   "ANSWER FORMAT (pick what fits; ensure correct number agreement and echo the original attribute verbatim):\n"
   "• The [main object] is not [new attribute], but it is [original attribute].\n"
   "• The [main object] does not have [new attribute], but it has [original attribute].\n"
   "• The [main object] contains no [new attribute], but it has/contains [original attribute].\n"
   "If none fits perfectly, write a brief, natural denial that clearly states the object lacks the [new attribute] and has the [original attribute]."
   "\n"
   "OUTPUT:\n"
   "Return EXACTLY ONE line:\n"
   "Q=<negative question> || A=<negative answer>\n"
   "No extra text."
```

Figure 23. Prompt Template for generating (q^-, a^+_-) for WH setting

Prompt template for Gemini-2.0-Flash extracting objects & attributes

```
"You are an expert at information extraction. Your task is to analyze an image description "
   "and extract all MAIN objects (objects that have at least ONE attribute) and their attributes into a JSON list of dictionaries.\n"
   "Follow these rules precisely:\n"
   "1. Your output MUST be a valid JSON list `[]` where each element is a dictionary.\n"
   "2. Each dictionary must contain two keys: `object` and `attribute`.\n"
   "3. The value for the `object` key must be a string containing the plain name of the MAIN object in the scene (e.g., 'airplane', 'man',
'bus').\n"
   "4. The value for the `attribute` key must be a list of strings. Each string must describe a characteristic of the object, such as its
appearance, material, color, text, number, size, state, or posture.\n"
   "5. Every attribute string MUST begin with the word 'with'. If the original description doesn't use 'with', rewrite the attribute to include it
while preserving the meaning.\n"
   "6. Do NOT include attributes that describe spatial relationships ('to the left of', 'behind') with OTHER MAIN objects. However, spatial
orientations NOT involving other MAIN OBJECTS should be counted as attributes. ('with its nose facing left').\n"
   "7. Do NOT include attributes that describe actions with OTHER MAIN objects('holding a black camera with flashy surface'). However, actions NOT
involving other MAIN OBJECTS should be counted as attributes ('with hands raising in the air')."
   "8. Return ONLY the JSON list and nothing else. Do not add explanations or markdown formatting."
```

Figure 24. Prompt Template for extracting objects and attributes using Gemini-2.0-Flash [41] when constructing FINER-DOCCI.

Prompt template for Gemini-2.0-Flash extracting relations

```
You are an expert at STRICT information extraction. Given
(1) a natural-language image description, and
(2) a numbered catalog of MAIN objects (each with ALL its attributes),

your task is to inspect pairs of objects and extract all **directed** relations
that are **explicitly stated in the description text**.

You must follow these rules:

1) Use ONLY the integer indices from the catalog for `object_a_idx` and `object_b_idx`.

2) For a pair (object_a, object_b), only output a relation if the caption
   directly describes how object_a is related to object_b in words.
   - Do NOT guess, infer, or rely on world knowledge.
   - If the relation is not explicitly written or is only implied, you MUST NOT output it.

3) The relation phrase `rel` must:
   - be a spatial or verb phrase,
   - be lower-case,
   - be copied verbatim from the description whenever possible, or be a **minimal paraphrase**
     that preserves exactly the same meaning (e.g., shortening function words).

4) The relation phrase must describe either:
   - a clear spatial relation (e.g., "is behind the", "is at the intersection of the"), or
   - a clear action or interaction (e.g., "holds", "moves along the").

5) The relation phrase must form a grammatical predicate between object_a and object_b:
   - it either begins with "is"/"are" (e.g., "is behind the", "is on top of the"), OR
   - it is a finite verb phrase that can directly follow object_a
     (e.g., "holds", "touches", "moves along the").

6) Output a JSON list of dictionaries, each of the form:
   {"object_a_idx": int, "rel": str, "object_b_idx": int}

7) No self-relations and no duplicate entries. If **no** explicit relations are found,
   return an empty list: [].
```

Figure 25. Prompt Template for extracting relations using Gemini-2.0-Flash [41] when constructing FINER-DOCCI.

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, 11, 14

[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, 9, 11

[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, 8

[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, 11, 13

[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, 1, 2

[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. 6, 7, 10

[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, 1

[9] Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James R Glass, and Pengcheng He. Dola: Decoding by contrasting layers improves factuality in large language models. In *ICLR*, 2023. 11, 13

[10] 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, 2, 11

[11] 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, 8, 10

[12] Ailin Deng, Tri Cao, Zhirui Chen, and Bryan Hooi. Words or vision: Do vision-language models have blind faith in text? In *CVPR*, 2025. 1

[13] Peng Ding, Jingyu Wu, Jun Kuang, Dan Ma, Xuezhi Cao, Xunliang Cai, Shi Chen, Jiajun Chen, and Shujian Huang. Hallu-pi: Evaluating hallucination in multi-modal large language models within perturbed inputs. In *ACM MM*, 2024. 1

[14] 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, 8, 10

[15] 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, 1

[16] 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. 5, 6, 8, 11, 13

[17] 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, 8

[18] 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

[19] Sanghwan Kim, Rui Xiao, Mariana-Iuliana Georgescu, Stephan Alaniz, and Zeynep Akata. Cosmos: Cross-modality self-distillation for vision language pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14690–14700, 2025. 1

[20] LAION. Releasing re-laion-5b: transparent iteration on laion-5b with additional safety fixes, 2024. Accessed: 30 aug, 2024. 7, 1

[21] 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, 10

[22] 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, 8, 11, 13

[23] 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, 1

[24] 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, 1, 2

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

[26] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *CVPR*, 2024. 11, 13

[27] 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, 13

[28] Yexin Liu, Zhengyang Liang, Yueze Wang, Xianfeng Wu, Feilong Tang, Muyang He, Jian Li, Zheng Liu, Harry Yang, Sernam Lim, et al. Unveiling the ignorance of mllms: Seeing

clearly, answering incorrectly. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025. 1

[29] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 8

[30] 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, 1

[31] 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

[32] 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, 10

[33] RePOPE: Impact of Annotation Errors on the POPE Benchmark. Neuhaus, yannic and hein, matthias. *arXiv*, 2025. 5, 6, 7, 1, 8, 11, 13

[34] 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, 1, 4

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

[36] 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

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

[38] 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, 1, 2, 11, 13

[39] 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, 10

[40] 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, 1, 2, 11

[41] 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, 1, 4, 9, 11, 24

[42] 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, 10

[43] Yahan Tu, Rui Hu, and Jitao Sang. Ode: Open-set evaluation of hallucinations in multimodal large language models. In *CVPR*, 2025. 1, 2

[44] 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, 5, 6, 7, 8, 11, 13

[45] 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. 5, 6, 7, 1, 8, 11, 13

[46] 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, 3, 10, 13, 14, 15

[47] 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, 1, 11, 13

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

[49] 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, 1, 11, 13

[50] Rui Xiao, Sanghwan Kim, Mariana-Iuliana Georgescu, Zeynep Akata, and Stephan Alaniz. Flair: Vlm with fine-grained language-informed image representations. In *CVPR*, 2025. 1

[51] 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, 2, 4

[52] 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, 1, 2

[53] 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, 1

[54] 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, 1, 2

[55] 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, 1, 2, 11, 13

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

[57] 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, 1, 2, 11, 13

[58] 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, 8

[59] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017. 8, 1

[60] Yiyang Zhou, Chenhang Cui, Rafael Rafailov, Chelsea Finn, and Huaxiu Yao. Aligning modalities in vision large language models via preference fine-tuning. *arXiv*, 2024. 1, 2

[61] 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, 1