OCCLUDEBENCH: CAN VLMS FIND WHAT'S MISSING?

Anonymous authors
Paper under double-blind review

ABSTRACT

How well can VLMs reason about what is missing from an image? We study this question with OCCLUDEBENCH, a manually curated benchmark for recovering fully occluded objects from the visual evidence that remains. OCCLUDEBENCH distinguishes **direct** cases, where the hidden object must be recovered from a specific physical cue such as a shadow or reflection, from **inferential** cases, where it can be inferred from broader scene context. Across five VLMs, accuracy averages 40.5% on direct cases versus 60.3% on inferential cases, exhibiting a 19.9-point gap. Humans attain higher accuracies of 63.4% versus 68.5% for direct and inferential images, respectively, and exhibit a much smaller gap of only 5.1 points. To locate this failure, we introduce OCCLUDETRACE, a four-stage diagnostic that isolates where the visual reasoning process breaks down. The bottleneck emerges when models must recover the relevant object from the projection within the full image: they succeed on only 45% of failed examples, compared with 90% of examples they answer correctly. Extended reasoning, tool use, and few-shot prompting yield inconsistent improvements and do not reliably eliminate the gap. In contrast, an inpaint-then-confront intervention, which prompts models to first generate a candidate hidden object and then evaluate that hypothesis against the visible physical cue, reduces the gap to a statistically non-significant difference for both models with image-generation capabilities. Together, our results establish a tractable, controlled diagnostic for isolating how VLMs reason about visual content that is missing from an image.

1 INTRODUCTION

VLMs have recently demonstrated strong performance across reasoning and vision-language benchmarks (Lu et al., 2022; Yue et al., 2024; Sarch et al., 2026). Large-scale systems such as GPT and Gemini (OpenAI, 2023; Gemini Team, 2023) further highlight the ability of these models to integrate information across modalities and solve complex tasks requiring both perception and reasoning. These successes concern content that is visible in the image. A basic question remains open: how well can VLMs reason about what is *missing* from an image? Everyday scenes routinely leave objects out of view, and recovering them requires reasoning from the evidence that remains, whether a shadow, a reflection, or the surrounding context. Because most benchmarks use full, unconstrained images, it is difficult to determine which visual cues a model uses, and hence whether it reasons from the evidence or answers from prior expectations.

We study this question with OCCLUDEBENCH, a manually curated benchmark for recovering fully occluded objects from the visual evidence that remains. We distinguish **direct** cases, where the hidden object must be recovered from a shadow or reflection, from **inferential** cases, where it can be inferred from broader scene context. Because both use the same task and masking procedure, the performance gap between them isolates differences in reasoning from specific physical evidence versus contextual inference (Figure 1).

We find models perform significantly worse on direct tasks than inferential ones, while humans show no corresponding gap. To locate where the reasoning process breaks down, we introduce OCCLUDETRACE, a four-stage diagnostic that decomposes recovering a hidden object into detecting the projection, recognizing it in isolation, mapping it to its source, and recovering the object from the cue inside the full image. The bottleneck is not seeing the shadow or reflection, but isolating which object within it actually identifies what is hidden.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

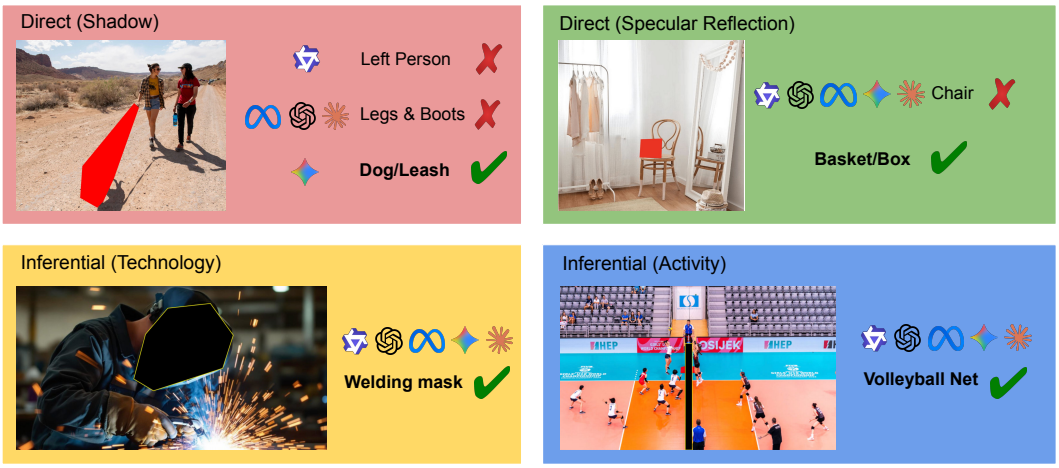


Figure 1. Examples of visual image problems involving direct or inferential evidence, along with their subcategories. We prompt the VLM to answer "**What is the primary object that is completely occluded in this image? Be as specific as you can.**" (see Appendix A.1). The responses of each model and the correct answer are shown on the right of each subfigure. Direct cases can be answered with a single visual cue, shadow or reflection, while inferential cases use broader context to arrive at the conclusion. In each question, we group the models by the response they each gave and indicated whether the answer is correct.

Attempts to mitigate this failure through in-context learning, tool use, extended reasoning, and targeted prompting yield inconsistent gains, and the direct-inferential gap still exists. The exception is an inpaint-then-confront intervention: the model first generates a candidate hidden object and then evaluates that hypothesis against the visible physical cue, which reduces the gap to a statistically non-significant difference for both models with image-generation capabilities.

Our dataset and diagnosis suggest that standard evaluations can mask an important failure mode: models may answer correctly without using the visual signal that determines the answer. By contrasting cases where the hidden object can only be identified from a specific visual cue with cases where it can be inferred from broader context, we identify evidence selection as a key missing capability in current VLMs. More broadly, occlusion provides a tractable, controlled way to test how VLMs reason about content that is missing from an image, and OCCLUDETRACE shows how such a test can be decomposed to isolate where that reasoning fails.

2 OCCLUDEBENCH

2.1 DATASET CREATION

Figure 2 provides an overview of the curation and evaluation pipeline used in this work. To create the dataset, we collected images from the Internet, Visual Genome (Krishna et al., 2017), and our own photographs, since synthetic images would weaken the ecological validity of the dataset. Initial testing also showed that AI-generated images for the direct category did not accurately portray the physical properties of shadows and reflections. LLM-generated examples of relational object appearances were also repetitive, not diverse, and did not scale sufficiently. For the direct images, we selected examples that explicitly contain a shadow or reflection and where the occluded object cannot be reasonably inferred from context. For the inferential examples, we followed the same procedure but instead looked for diverse images in which the occluded object can be reasonably inferred from contextual cues.

Dataset Distribution. The direct category is composed of different types of physical projection cues, including specular reflections (47.7%), diffuse reflections (28.4%), and shadows (23.9%). The inferential category spans a range of semantic domains, including technology (24.9%), household scenes (24.3%), culture (20.4%), activities (17.1%), and nature (13.3%). OCCLUDEBENCH contains 378 images, with 197 in the direct category and 181 in the inferential category.

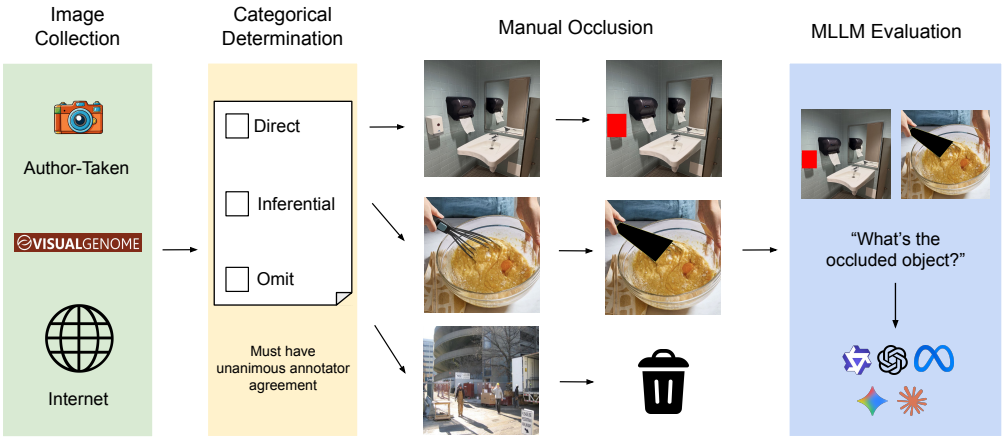


Figure 2. Overview of our dataset curation and evaluation pipeline. After sourcing images from the internet, existing datasets, and from self-collection, human annotators then unanimously agree on the correct category for the image as defined in Section 2.1. The manually occluded image is then fed into the VLM for evaluating how well it identifies the occluded object.

Occlusion Procedure. For each image, we artificially occlude a target object using a clearly visible bounding box or polygon, making it explicit that part of the scene is hidden through manual annotation where the bounding box/polygons coordinates are included in the dataset. We then include the image into our dataset if the human annotators unanimously agree that the image cleanly fits into the specified category. We use artificial occlusion rather than natural occlusion to avoid introducing confounds. For example, naturally occluded shadows or reflections may be distorted or missing.

2.2 Evaluation

Task Setup. We evaluate five state-of-the-art VLMs: Llama 4 Maverick (Meta, 2025), Qwen3-VL-8B-Instruct (Bai et al., 2025), Claude Opus 4.6 (Anthropic, 2026), Gemini 3.1 Pro (Google DeepMind, 2026), and GPT-5.4 (OpenAI, 2026). We prompt the models to produce open-ended responses (not multiple-choice) on what they believe the occluded object in the image to be. The task is designed to test whether models can recover the hidden object from the remaining visual evidence, either through direct cues or contextual understanding. We report accuracy based on whether the predicted object matches the ground truth under our judging framework.

Grading. We evaluate model predictions using an LLM-as-a-judge framework consisting of Claude Opus 4.6 and GPT-5.4 to determine whether the occluded object has been correctly identified (Zheng et al., 2023). We mark a prediction as correct if either judge deems it correct. We find that this LLM-as-a-judge framework achieves 98% agreement with ground-truth annotations by the authors, slightly higher than the agreement rate of single-judge evaluations, indicating high reliability. All prompts for evaluation and LLM-as-a-judge can be found in Appendix A.

2.3 Human Baseline

We collect a human baseline using Prolific (Prolific, 2025), recruiting 473 English-speaking participants from the United States, United Kingdom, Canada, Australia, and New Zealand. **Each participant is presented with 6 questions, consisting of 3 direct and 3 inferential items.** Questions are randomly sampled from our dataset under the constraint that all questions are approximately equally represented across participants. The order of the 6 questions is independently randomized for each participant, and questions are shown sequentially, one at a time. Responses are collected in a free-form setting without multiple-choice options.

Compensation and Protocol. Participants are compensated at a rate of $12.50/hour, with a median completion time of 2 minutes and 30 seconds. The task is conducted in a zero-shot setting, where participants do not receive example questions.

Model	Direct (%)	Inferential (%)	Gap
GPT-5.4	36.0	59.7	23.7
Claude Opus 4.6	48.5	72.9	24.4
Gemini 3.1 Pro	69.0	86.2	17.2
Qwen 3 VL 8B	19.1	43.6	24.5
Llama 4 Maverick	29.8	39.2	9.4

Table 1. VLM accuracy on occluded object identification, split by direct and inferential images. Gap is inferential minus direct accuracy in percentage points. Every model scores lower on direct images.

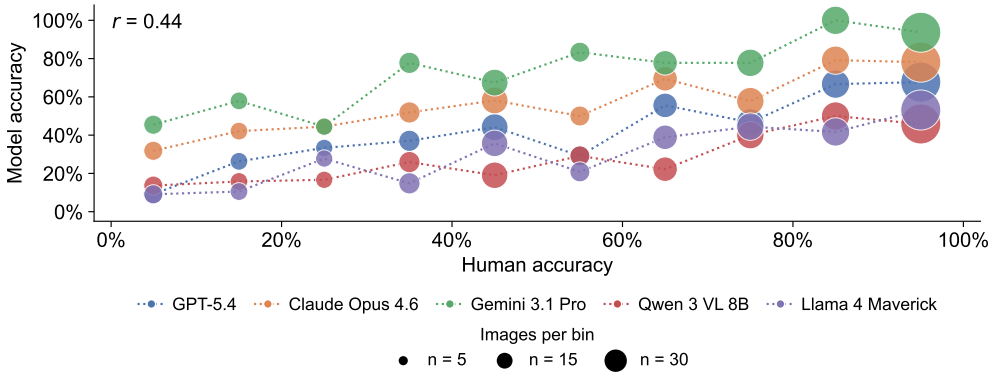


Figure 3. Model accuracy compared to human accuracy for occluded object identification. Each dot is a 10-point bin of per-image human accuracy, sized by the number of images in the bin. Across all five VLMs, questions that are more difficult for humans are also more difficult for models ($r = 0.44$ between per-image human accuracy and mean model accuracy).

3 Results

3.1 Main Results

VLMs struggle more with identifying occluded objects in direct images than in inferential ones. VLMs identify the occluded object correctly in 50.0% of cases on average (Table 1). Gemini 3.1 Pro has the highest total accuracy at 77.2% among all inferential and direct images, while Qwen 3 VL 8B attains the lowest at 30.9%. However, across all five VLMs evaluated, accuracy is highly variable between direct and inferential images; VLMs consistently attain lower accuracies when discerning the occluded object for direct images (19.9% lower on average) than with inferential images. While the smallest gap is found in Llama 4 Maverick at 9.4%, the rest of the VLMs have performance deficiencies of at least 17.2%, with the highest gap being observed with Qwen 3 VL 8B at 24.5%. Lastly, performance is also largely uniform across occlusion cue categories (Figure 9, Appendix B).

Most VLMs perform worse than humans at identifying the occluded object. Across all individual responses, humans achieve an accuracy of 65.9%, exceeding four of the five evaluated VLMs. Moreover, unlike VLMs, humans do not exhibit a statistically significant direct–inferential performance gap: human accuracy is 63.4% on direct images and 68.5% on inferential images, compared with model averages of 40.5% and 60.3%, respectively (Table 1). Models tend to perform worse on images that humans also find difficult ($r = 0.44$), and likewise with easier images in both the direct and inferential categories (Figure 3, Appendix D).

3.2 OccludeTrace: Why Do VLMs Fail?

VLMs misidentify occluded objects in direct images more frequently, but not where the failure occurs. Identifying an object from its shadow or reflection requires several skills in sequence: noticing that a projection exists, reading the object it depicts, linking the projection to the part of the scene that casts it, and recovering the object from the projection while it sits inside the full image. Any one of these



Figure 4. Examples of the OCCLUDETRACE stages. **(Left)** Zoomed-in object recognition: the model sees only the cropped projection and names the object. **(Middle)** Source mapping: the model chooses which of four candidate regions is the source of the marked projection. **(Right)** Highlighted-region recovery: the model describes everything in the marked projection region, with no mention of shadows, reflections, or occlusion.

OCCLUDETRACE stage	Accuracy on failed examples	Accuracy on successful examples
1. Cue detection	99%	100%
2. Zoomed-in object recognition	80%	83%
3. Source mapping	85%	82%
4. Highlighted-region recovery	45%	90%

Table 2. Accuracy at each OCCLUDETRACE stage on direct images, computed separately over the examples answered incorrectly and correctly in the base condition. Only the final stage separates the two groups.

could be the point of failure. We therefore develop OCCLUDETRACE, a four-stage diagnostic that isolates each skill with its own prompt on the same direct images (examples in Figure 4). A stage that causes the failures should show a large gap between the groups, whereas a stage that models handle equally well in both groups is not the bottleneck. We describe each stage and their results (Table 2).

Stage 1: Cue detection. We ask whether the image contains a shadow or reflection. Models answer correctly on 99% of failed examples and 100% of successful ones, so models notice the projection even when they go on to answer incorrectly.

Stage 2: Zoomed-in object recognition. We crop the shadow or reflection out of the image and ask the model what object it shows. Accuracy is 80% on failed examples and 83% on successful ones. When the projection is presented in isolation, models read its content about equally well regardless of whether they later fail on the full image.

Stage 3: Source mapping. We mark the projection with a transparent bounding box, occlude four candidate regions of the image, and ask which region casts the shadow or reflection. Models choose the correct region on 85% of failed and 82% of successful examples, so they can connect the projection to the occluded region that produces it.

Stage 4: Highlighted-region recovery. We mark the projection with a transparent bounding box in the intact image and ask the model to describe everything in the region in detail (prompt in Appendix A.4). We then check whether the description names the depicted object. This is the only stage where the two groups diverge: models recover the object on 90% of successful examples but on only 45% of failed ones.

Diagnosis. The first three stages rule out the obvious explanations. Failures are not caused by missing the cue, by being unable to read a shadow or reflection, or by being unable to link it to its source. Models fail at the stage where the projection must be read while embedded in the complete image, even when the region is pointed out to them. The bottleneck is visual extraction in context rather than reasoning about physical cues, which motivates the interventions in Section 3.3.

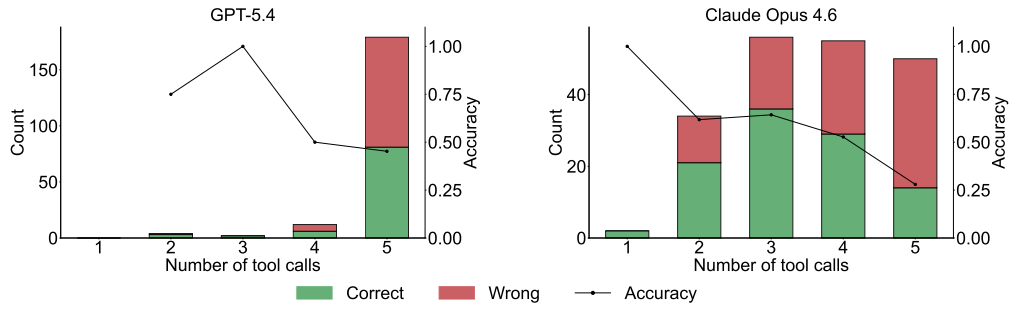


Figure 5. Tool call analysis on direct images. **Left:** GPT-5.4 accuracy by number of tool calls. GPT-5.4 exhausts its 5-call budget ($n = 179$) at 45% accuracy. There were no traces where GPT-5.4 used exactly one tool. **Right:** Claude Opus 4.6 accuracy by number of tool calls; accuracy declines monotonically from 62% (2 calls) to 28% (5 calls).

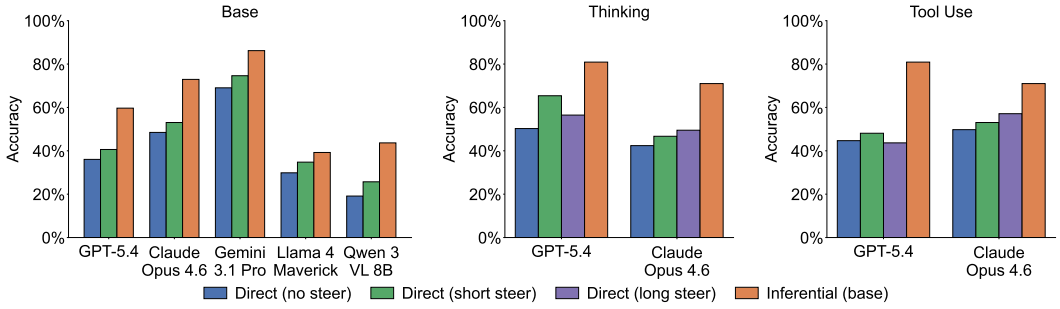


Figure 6. Accuracy on occluded object identification. All VLMs perform worse on direct images than on inferential images. Tool use and prompt steers provide consistent improvements but are insufficient to close the gap between direct and inferential images.

3.3 Can we improve performance on direct images?

Extended Reasoning. We investigate whether additional reasoning can improve performance overall. We evaluate the performance of GPT-5.4 and Claude Opus 4.6 with their respective highest reasoning modes activated on the same initial experimental setup. As shown in Figure 7, Claude Opus 4.6 decreases in accuracy for both direct and inferential images, whereas GPT-5.4 has accuracy increases for both categories. However, both GPT-5.4 and Claude Opus 4.6 still perform worse on direct than with inferential images.

Tool Use. We investigate whether augmenting VLMs with image-based tools can aid performance on direct images. We use tools commonly used in image manipulation, including cropping, reflecting, edge detection, rotating, and zooming (see Appendix A.5.1 for the details). We conduct experiments with GPT-5.4 and Claude Opus 4.6, high reasoning enabled. GPT-5.4 encounters a decrease in accuracy from 50.3% to 44.7%. Tool usage does not manage to bring the performance of Claude Opus 4.6 on direct images (42 to 51%) up to that for inferential images. Notably, more tool calls do not translate to higher accuracy: for both models, the correct-answer group uses *fewer* tools on average than the incorrect group. The two models also exhibit very different tool-call profiles (Figure 5), with GPT-5.4 nearly always exhausting its 5-call budget and relying heavily on `crop` and `zoom`, while Claude Opus 4.6 distributes calls more evenly and rarely crops. Expanding to full agentic harnesses with unrestricted code execution and high reasoning does not close the gap either. Codex with GPT-5.4 reaches 51.8% and Claude Code with Claude Opus 4.6 reaches 34.5%, both still well below inferential performance.

Steering Prompts. We additionally test two steering prompts that direct models to attend to projections: a short steering prompt ("Pay close attention to any reflective surfaces or shadows in the image as you will need it to answer the question.") and a longer steering prompt that further specifies what counts as a valid answer (full text in Appendix A.6). As shown in Figure 6, both steers yield

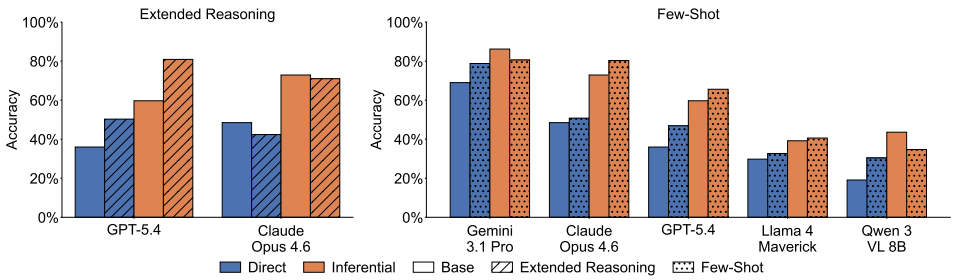


Figure 7. Accuracies when attempting to improve VLM performance on the occluded object task. Extended reasoning and few-shot prompting consistently improve performance with direct images, but don't reliably close the performance gap between direct and inferential images. Sometimes, for inferential images, performance decreases.

Model	Comparison	Rates	Difference	p
GPT-5.4	baseline direct vs inferential	37.6% vs 62.4%	+24.9pp	**<0.0001**
GPT-5.4	infill direct vs inferential	54.3% vs 62.4%	+8.1pp	0.12
Gemini 3.1 Pro	baseline direct vs inferential	71.6% vs 87.8%	+16.3pp	**0.0001**
Gemini 3.1 Pro	infill direct vs inferential	82.2% vs 87.8%	+5.6pp	0.15

Table 3. Accuracy of inpainting multi-step evaluation and p-values for GPT-5.4 and Gemini 3.1 Pro.

consistent improvements across base, thinking, and tool-use settings, with the largest gains on Claude Opus 4.6 + tool use (42.4% → 57.1%) and GPT-5.4 + thinking (50.3% → 65.4%). However, no steer configuration closes the gap to inferential performance, suggesting models can be cued to consider physical projections like shadows and reflections but still fail to reason about them reliably.

Few-Shot Prompting. We find that few-shot prompting has limited effects in reducing the performance gap. For the few-shot experiment, we include one direct and one inferential example (see Appendix A.5, **Few-Shot**). We hold out these two examples from the evaluation. As shown in Figure 7, while Gemini 3.1 Pro and Qwen 3 VL 8B have small performance gaps between direct and inferential images (79 vs. 81% and 30 vs. 35%, respectively), gaps still exist for GPT-5.4 (47 vs. 66%), Claude Opus 4.6 (51 vs. 80%), and Llama 4 Maverick (33 vs. 41%). Furthermore, we observe that the smaller gap observed with Qwen 3 VL 8B is associated with reduced accuracy for inferential images (43.6% to 34.7%).

Inpainting Multi-step Execution. We next test whether models can overcome the direct–inferential gap by converting open-ended object identification into a more discriminative, iterative task. We first ask the model to inpaint the occluded region with the hidden object made visible (prompts in Appendix A.7). We then present the resulting image and ask, "Does this infill make sense physically, given the properties of reflections and shadows? Now, what is the occluded object? Be as specific as you can." The generated infill provides a concrete hypothesis for the model to evaluate (Figure 8): mismatches between the proposed object and the observed shadow or reflection expose constraints on what the hidden object could instead be. Thus, rather than identifying the object directly from an indirect physical cue, the model can discriminate among and revise explicit visual hypotheses.

This is the most effective intervention we evaluate. Extended reasoning and tool use fail to eliminate the significant direct–inferential gap for either tested model, while few-shot prompting does so for only one of five models. In contrast, inpaint-then-confront eliminates the significant gap for both GPT-5.4 and Gemini 3.1 Pro, the two models with image-generation capabilities (Table 3). This suggests that direct physical evidence is not simply inaccessible to these models: once the inference is externalized as a visual hypothesis that can be inspected and rejected, they are substantially better able to detect and utilize it.

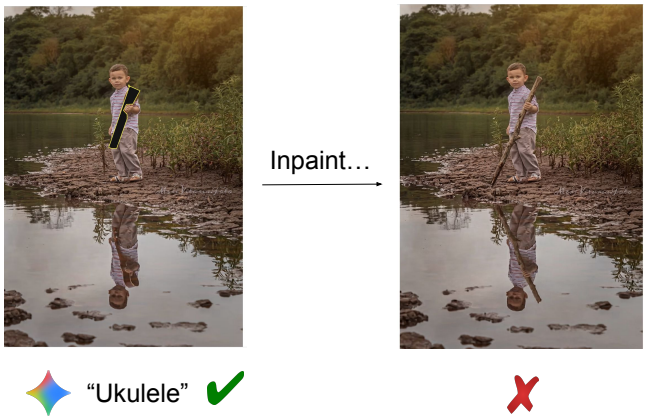


Figure 8. Inpaint-then-confront turns identification into hypothesis testing. Reading a hidden object directly off an indirect cue is hard, but checking whether a proposed object is consistent with that cue is easier. **(Left)** The occluded image: the reflection in the water constrains what the hidden object can be, and Gemini 3.1 Pro's text answer, a ukulele, satisfies that constraint. **(Right)** Asked to inpaint the region, the model externalizes a different hypothesis, a stick, and repaints the reflection to match it. Because the hypothesis is now a concrete image rather than a step buried in the model's reasoning, it can be held up against the original cue and rejected. The ukulele-shaped reflection rules the stick out.

4 Related Work

VLMs and Evaluations. Recent unified models support both visual understanding and generation (Xie et al., 2025; Wu et al., 2025a;b), yet these capabilities can conflict, as shared encoders force trade-offs between comprehension and generation (Wu et al., 2025a). VLMs have advanced rapidly through integration of vision encoders with LLM backbones (Alayrac et al., 2022; Liu et al., 2023b; Dai et al., 2023), achieving strong performance on perception and general multimodal evaluation benchmarks (Antol et al., 2015; Fu et al., 2025a; Liu et al., 2024; Li et al., 2024) and complex reasoning benchmarks (Yue et al., 2024; Xu et al., 2026; Lu et al., 2024; Yu et al., 2024). Nevertheless, reasoning about hidden objects remains underexplored.

Occlusion and Indirect Visual Evidence. Amodal completion, inferring occluded objects via semantics or symmetry, is a recognized core visual reasoning ability (Zhan et al., 2024; Ao et al., 2025; Pothiraj et al., 2025). While recent benchmarks probe spatial reasoning (Liu et al., 2023a; Fu et al., 2024) and partial occlusion (Liu et al., 2025), cases of total occlusion remain underexplored. Physical commonsense benchmarks (Bear et al., 2021; Chow et al., 2025) test related intuitions, but they rarely require models to reason from indirect visual cues like projections or reflections. Crucially, where prior work treats shadows and reflections as noise to be removed (Le & Samaras, 2019; Kee et al., 2025; Hu et al., 2026), we leverage these indirect cues as the primary evidence for reasoning about fully hidden objects.

Inference and Evaluation Gaps. Identifying occluded objects also requires semantic priors governing object co-occurrence (Marino et al., 2019; Zellers et al., 2019; Schwenk et al., 2022), cues humans naturally exploit (Kellman & Spelke, 1983; Võ et al., 2019) but whose use in VLMs remains unclear. Recent work has begun probing such logical scene reasoning (Zhao et al., 2025; Lin et al., 2026); we extend this to completely hidden objects. Meanwhile, broader scrutiny of VLM benchmarks reveals that many can be solved without visual input (Chen et al., 2024), models fail on low-level visual tasks where humans succeed (Fu et al., 2024; Tong et al., 2024), and task-relevant visual information captured by encoders often goes unused by downstream modules (Fu et al., 2025b; Liu et al., 2026).

5 Discussion

OccludeBench asks how well VLMs reason about what is missing from an image, using occlusion as a controlled diagnostic: we choose what is hidden, know the answer, and control the evidence that

remains. The answer is that VLMs often fail not because they cannot perceive visual information, but because they do not reliably extract and use the evidence required to recover the missing object. VLMs achieve only 50.0% accuracy compared to 65.9% for humans, with a 19.9% drop on direct evidence cases, a gap not observed in human performance. OCCLUDETRACE, our four-stage diagnostic, isolates where the reasoning process breaks down: models detect that a cue exists, recognize it in isolation, and map it to its source, but struggle to detect the relevant object within the projection once it is embedded in the full image. The most effective intervention we tested, inpaint-then-confront, has the model generate a candidate hidden object and then evaluate that hypothesis against the visible physical cue; it reduces the gap to a statistically non-significant difference for both models with image-generation capabilities, suggesting that making a hypothesis concrete enough to check against the evidence helps models catch their own mistakes.

OCCLUDEBENCH contains 378 manually curated images. Its simple design around reflections and shadows lets us isolate whether VLMs can reliably reason from exact visual evidence in a setting that is straightforward for humans. Our results suggest that current evaluations can overestimate visual reasoning by allowing models to succeed through partial cues or learned associations rather than the evidence that actually determines the answer. More broadly, these findings establish occlusion as a tractable, controlled diagnostic for how VLMs reason about missing visual content, highlight reliable use of task-relevant evidence as an unresolved challenge for current VLMs, and motivate future systems that more faithfully ground predictions in visual input.

REPRODUCIBILITY STATEMENT

All prompts used in our experiments are given verbatim in the appendix, along with the judging protocol, the tool descriptions, and the human study interface. We will release the dataset, occlusion masks, annotations, evaluation code, and raw model outputs upon publication.

ETHICS STATEMENT

Our human baseline was collected through Prolific from 473 participants who were shown the study scope, compensation, and consent information before starting and were paid $12.50 per hour. Only free-form text answers were collected and results are reported in aggregate. The images depict everyday scenes and the task never asks about anyone's identity. We see no direct negative societal impact from this work; its purpose is to diagnose whether models ground their answers in visual evidence.

GENERATIVE AI DISCLOSURE

In this work, we used generative AI tools to provide feedback on research methodology and experiments, to implement methods, and to support qualitative and thematic data analysis, namely LLM-as-a-judge grading of model responses and coding of tool-use traces. We have not used generative AI tools to propose or refine hypotheses, to clean or reformat the dataset, or to interpret results, and generating synthetic data sets, developing theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, and assisting with translation are not applicable to this work. Additionally, we used generative AI tools to create and edit software code, to create artifacts, for brainstorming, for sourcing and searching for information, to identify relevant literature, and to edit the paper to improve readability. We have reviewed all AI-assisted work. AI-assisted code was reviewed and tested for correctness by two authors; papers and information surfaced by AI tools were read and verified manually before use; and the LLM-as-a-judge grading was validated against author-provided gold labels, with which it agrees on 98% of items. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L. Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikołaj Bińkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: A visual language model for few-shot learning. In *NeurIPS*, 2022.

Anthropic. Introducing Claude Opus 4.6. https://www.anthropic.com/news/claude-opus-4-6, 2026.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual question answering. In *ICCV*, 2015.

Jiayang Ao, Yanbei Jiang, Qiuhong Ke, and Krista A. Ehinger. Open-world amodal appearance completion. In *CVPR*, 2025.

Shuai Bai et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Daniel Bear, Elias Wang, Damian Mrowca, Felix Binder, Hsiao-Yu Tung, Pramod RT, Cameron Holdaway, Sirui Tao, Kevin Smith, Fan-Yun Sun, Fei-Fei Li, Nancy Kanwisher, Josh Tenenbaum, Dan Yamins, and Judith Fan. Physion: Evaluating physical prediction from vision in humans and machines. In *NeurIPS Datasets and Benchmarks Track*, 2021.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. Concreteness ratings for 40 thousand generally known English word lemmas. *Behavior Research Methods*, 2014.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. Are we on the right way for evaluating large vision-language models? In *NeurIPS*, 2024.

Wei Chow, Jiageng Mao, Boyi Li, Daniel Seita, Vitor Campagnolo Guizilini, and Yue Wang. PhysBench: Benchmarking and enhancing vision-language models for physical world understanding. In *ICLR*, 2025.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In *NeurIPS*, 2023.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, Rongrong Ji, Caifeng Shan, and Ran He. MME: A comprehensive evaluation benchmark for multimodal large language models. In *NeurIPS Datasets and Benchmarks Track*, 2025a.

Stephanie Fu, Tyler Bonnen, Devin Guillory, and Trevor Darrell. Hidden in plain sight: VLMs overlook their visual representations. In *COLM*, 2025b.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. BLINK: Multimodal large language models can see but not perceive. In *ECCV*, 2024.

Gemini Team. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Google DeepMind. Gemini 3.1 Pro model card. https://deepmind.google/models/model-cards/gemini-3-1-pro/, 2026.

Xiaowei Hu, Zhenghao Xing, Tianyu Wang, Chi-Wing Fu, and Pheng-Ann Heng. Unveiling deep shadows: A survey and benchmark on image and video shadow detection, removal, and generation in the deep learning era. *IJCV*, 2026.

Eric Kee, Adam Pikielny, Kevin Blackburn-Matzen, and Marc Levoy. Removing reflections from RAW photos. In *CVPR*, 2025.

Philip J. Kellman and Elizabeth S. Spelke. Perception of partly occluded objects in infancy. *Cognitive Psychology*, 1983.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 2017.

Hieu Le and Dimitris Samaras. Shadow removal via shadow image decomposition. In *ICCV*, 2019.

Bohao Li, Yuying Ge, Yixiao Ge, Guangzhi Wang, Rui Wang, Ruimao Zhang, and Ying Shan. SEED-Bench: Benchmarking multimodal large language models. In *CVPR*, 2024.

Ling Lin, Yang Bai, Heng Su, Congcong Zhu, Yaoxing Wang, Yang Zhou, Huazhu Fu, and Jingrun Chen. OODBench: Out-of-distribution benchmark for large vision-language models. *arXiv preprint arXiv:2602.18094*, 2026.

Fangyu Liu, Guy Emerson, and Nigel Collier. Visual spatial reasoning. *TACL*, 2023a.

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

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. MMBench: Is your multi-modal model an all-around player? In *ECCV*, 2024.

Zhaochen Liu, Kaiwen Gao, Shuyi Liang, Bin Xiao, Limeng Qiao, Lin Ma, and Tingting Jiang. Beyond the visible: Benchmarking occlusion perception in multimodal large language models. *arXiv preprint arXiv:2508.04059*, 2025.

Zhining Liu, Ziyi Chen, Hui Liu, Chen Luo, Xianfeng Tang, Suhang Wang, Jingying Zeng, Zhenwei Dai, Zhan Shi, Tianxin Wei, Hanqing Lu, Benoit Dumoulin, and Hanghang Tong. Seeing but not believing: Probing the disconnect between visual attention and answer correctness in VLMs. In *ICLR*, 2026.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *NeurIPS*, 2022.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *ICLR*, 2024.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. OK-VQA: A visual question answering benchmark requiring external knowledge. In *CVPR*, 2019.

Meta. Llama 4 model card. https://github.com/meta-llama/llama-models/blob/main/models/llama4/MODEL_CARD.md, 2025.

OpenAI. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

OpenAI. GPT-5.4 thinking system card. https://openai.com/index/gpt-5-4-thinking-system-card/, 2026.

Atin Pothiraj, Elias Stengel-Eskin, Jaemin Cho, and Mohit Bansal. CAPTURe: Evaluating spatial reasoning in vision language models via occluded object counting. In *ICCV*, 2025.

Prolific. Prolific. https://www.prolific.com, 2025.

Gabriel Sarch, Linrong Cai, Qunzhong Wang, Haoyang Wu, Danqi Chen, and Zhuang Liu. Vero: An open RL recipe for general visual reasoning. *arXiv preprint arXiv:2604.04917*, 2026.

Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-OKVQA: A benchmark for visual question answering using world knowledge. In *ECCV*, 2022.

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.

Melissa Le-Hoa Võ, Sage E. P. Boettcher, and Dejan Draschkow. Reading scenes: How scene grammar guides attention and aids perception in real-world environments. *Current Opinion in Psychology*, 2019.

Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, and Ping Luo. Janus: Decoupling visual encoding for unified multimodal understanding and generation. In *CVPR*, 2025a.

Yecheng Wu, Zhuoyang Zhang, Junyu Chen, Haotian Tang, Dacheng Li, Yunhao Fang, Ligeng Zhu, Enze Xie, Hongxu Yin, Li Yi, Song Han, and Yao Lu. VILA-U: A unified foundation model integrating visual understanding and generation. In *ICLR*, 2025b.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. In *ICLR*, 2025.

Weiye Xu, Jiahao Wang, Weiyun Wang, Zhe Chen, Wengang Zhou, Aijun Yang, Lewei Lu, Houqiang Li, Xiaohua Wang, Xizhou Zhu, Wenhai Wang, Jifeng Dai, and Jinguo Zhu. VisuLogic: A benchmark for evaluating visual reasoning in multi-modal large language models. In *ICLR*, 2026.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. MM-Vet: Evaluating large multimodal models for integrated capabilities. In *ICML*, 2024.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *CVPR*, 2024.

Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *CVPR*, 2019.

Guanqi Zhan, Chuanxia Zheng, Weidi Xie, and Andrew Zisserman. Amodal ground truth and completion in the wild. In *CVPR*, 2024.

Yu Zhao, Ying Zhang, Xuhui Sui, Baohang Zhou, Xinying Qian, Li Shen, and Dacheng Tao. From hindsight to foresight: Self-encouraged hindsight distillation for knowledge-based visual question answering. *arXiv preprint arXiv:2511.11132*, 2025.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *NeurIPS Datasets and Benchmarks Track*, 2023.

A PROMPTS

This appendix collects the full text of every prompt used in our evaluation pipeline. Each subsection corresponds to the section of the main body where the prompts are used, and we briefly describe the role of each prompt before listing it verbatim. All prompts are issued at temperature 0 with open-ended (free-form) responses; we do not use multiple-choice options at any stage.

A.1 EVALUATION

Pipeline Overview. The base evaluation issues a single *Main Occluded Query* and records the model's open-ended response. **Main Occluded Query**

```
What is the primary object that is completely occluded in this image?
Be as specific as you can.
```

A.2 JUDGING

Ensemble Decision Rule. We use Claude Opus 4.6 and GPT-5.4 as independent judges. Each judge receives the ground-truth label and the model's free-form response, and returns a binary correctness verdict. Across the 378 items, this two-judge ensemble agrees with author-provided gold labels 98% of the time, exceeding either single-judge agreement rate.

Judge Correctness (Model Response)

```
You are evaluating whether a vision model correctly identified an
occluded object.

Ground truth:  "{ground_truth}"

Model response:  "{response}"

Does the model's response correctly identify the same object as
the ground truth?  Allow for reasonable variation in phrasing,
specificity, or synonyms (e.g., 'barber pole' matches 'barber's pole',
'wing mirror' matches 'side mirror', 'cat' matches 'pet' if the ground
truth is 'Reasonable Pet').  The response should identify the core
object -- partial credit guesses that get the category right but miss
specifics should be marked correct only if the category-level answer
is what the ground truth describes.  Answer only 'yes' or 'no'.
```

A.3 ROBUSTNESS EXPERIMENTS

Localization Control. The transparent-box query is used in the unmasked-bounding-box experiment to rule out localization failures (Table 4). The image is shown with the bounding box drawn as a yellow outline and the underlying contents visible, and the model is asked what the box contains. Asking about the box rather than about "what is occluded" keeps the prompt symmetric across direct and inferential items and avoids leaking the answer's modality.

Localization is not the bottleneck. Another concern is that VLMs may fail to identify the location of the occluded object in the first place. To address this, we evaluate how well VLMs identify the object within the occlusion bounding box, but unmasked. This serves as a proxy test for identifying the location of the occlusion bounding box itself. VLMs do so 86-92% of the time, with near-equal performance for both direct (85-93% accuracy) and inferential (86-91% accuracy) images (Table 4). Higher performance on transparent compared to occluded recognition suggests that failures cannot be primarily attributed to difficulties in localizing the occluded region of the image, indicating that identifying the bounding box is not a confound.

Transparent-Box Query

	Occluded (%)		Unmasked (%)	
Model	Direct	Infer.	Direct	Infer.
GPT-5.4	36.0	59.7	91.1	91.4
Claude Opus 4.6	48.5	72.9	85.3	90.6
Gemini 3.1 Pro	69.0	86.2	92.9	90.6
Qwen 3 VL 8B	19.1	43.6	87.0	88.7
Llama 4 Maverick	29.8	39.2	84.9	86.5

Table 4. VLM accuracy on occluded and unmasked images. *Occluded*: original images with the object hidden. *Unmasked*: bounding box shown but contents visible.

```
What is the object outlined by the yellow border in this image?  Be
as specific as you can.
```

A.4 Behavior Analysis of VLMs on Direct Images

Probing Scene Understanding Without Cue Framing. The Comprehensive Scene Description prompt is used in the projection-region probing experiment (Section 3.2, Stage 4). To separate perception failures from reasoning failures, the prompt avoids any mention of shadows, reflections, or occlusion and just asks for an exhaustive description of the highlighted region. We then check post hoc whether the description names the object depicted within the projection.

Comprehensive Scene Description of Highlighted Region

```
This image contains a thick-bordered transparent bounding box (the
border may be yellow, red, or green) highlighting a region of
the scene.  Provide a comprehensive, detailed description of the
entire scene:  describe every object, person, animal, text, surface,
background, lighting, colors, textures, spatial relationships, and
any other visual details you can observe.  Be exhaustive and precise.
```

A.5 Can we feasibly improve VLM performance on direct images?

Extended Reasoning Query (Same Task Prompt)

```
What is the primary object that is completely occluded in this image?
Be as specific as you can.
```

Tool-Augmented Wrapper Design. For the tool-use condition we wrap the same base query inside a per-round controller that gives the model a fixed budget of `{MAX_API_ROUNDS}` rounds, requires exactly one tool call per round, and terminates when the model submits a `final_answer`. The wrapper passes image dimensions in the prompt so the model can issue valid pixel coordinates directly. Restricting each round to a single tool call prevents the model from issuing many small operations per turn and matches the trace granularity used in the analysis in Section 3 and Appendix A.5.1.

Tool-Augmented Wrapper Instruction

```
Image dimensions:  {w}x{h} pixels (width x height, origin top-left).
You have at most {MAX_API_ROUNDS} rounds to examine the image.  Call
exactly one tool per round, then submit a final_answer when ready.

{prompt}
```

Tool-Augmented Base Query (Occluded)

```
What is the primary object that is completely occluded in this image?
Be as specific as you can.
```

Tool Loop Fallback

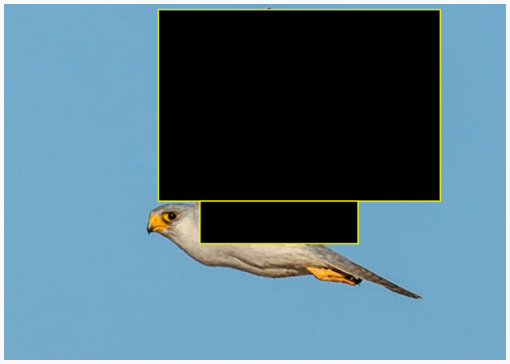
```
Please give your final answer now.
```

Held-Out Few-Shot Examples. The few-shot condition prepends two assistant traces to the base query: one direct example and one inferential example. Both items are excluded from the evaluation split for any few-shot run, so the reported few-shot numbers are computed on $378 - 2 = 376$ items. Each trace rules out plausible alternatives from context and then justifies its answer using the visible cue.

Few-shot Query

```
What is the primary object that is completely occluded in this image?
Be as specific as you can.
```

Few-shot Assistant Example 1



```
The answer is not the sky, as the sky is not completely occluded, nor
is it the bird, as the bird is not completely occluded.  The only
object that is completely occluded is the bird's wings.  It is the
bird's wings because, since a bird has wings and it is in the sky,
the only possible object that could be flapping is its wings.  Hence
the correct answer is wings.
```

Few-shot Assistant Example 2



```
The answer is not drink, as there are many possible drinks.  It's
also not just mug, as there are many different colors of mugs.  It
is specifically a dark mug.  We can arrive at this conclusion by
observing that the teapot is casting a reflection.  In the reflection
we observe the white mug and the plate, which are present in the
image, but we see that the dark mug next to the white mug is not
present, hence occluded.  Thus the answer is dark mug.
```

A.5.1 Description of Tools

Tool Palette Rationale. The tool set covers the basic manipulations a human would try when inspecting a shadow or reflection (zoom, crop, edge detection, flip, rotate), plus a final-answer signal. We exclude higher-level operations such as segmentation, object detection, and OCR, so that gains from tool use reflect geometric inspection rather than offloading recognition to an external module.

crop

```
Crops a rectangular region of the current image for closer inspection
using pixel coordinates (x1, y1, x2, y2) with top-left origin.  This
lets the model isolate a potentially informative local area without
altering semantic content.
```

edge_detect

```
Applies edge detection to the current image to emphasize boundaries,
contours, and high-frequency structure.  This is intended to make
weak shape cues more visible when texture or shading is ambiguous.
```

zoom

```
Extracts a rectangular region and rescales it back to the original
image dimensions.  Unlike simple crop, this preserves the full-frame
input size while magnifying a selected area to support fine-grained
inspection.
```

flip_horizontal

```
Mirrors the current image left-to-right.  This provides an
alternative orientation that can sometimes make alignment patterns
or object asymmetries easier to notice.
```

flip_vertical

```
Mirrors the current image top-to-bottom.  This offers another
geometric transform for inspecting whether weak visual evidence
becomes more interpretable under inversion.
```

rotate

```
Rotates the image counter-clockwise by a specified angle (for example
90, 180, or 270 degrees).  This helps when cues are easier to parse
in a canonical orientation.
```

final_answer

```
Signals termination of tool interaction and submits the model's final
predicted object label.  This tool does not transform the image; it
commits the final textual answer for scoring.
```

A.6 Steering Prompts

We test two steering prompts designed to direct VLMs toward indirect visual cues (shadows, reflections, and other physical traces) when answering questions about occluded objects. Both steering prompts are appended to the base prompt used in our main experiments.

Design Intent. The two prompts vary in strength. The short prompt only points the model at reflective surfaces and shadows; it says nothing about what a valid answer looks like. The long prompt additionally specifies the answer format ("do not answer 'shadow' or 'reflection"') and gives the model an inspection order. Comparing the two separates gains from attention direction alone from gains that also constrain the response format.

Short steering prompt

```
Pay close attention to any reflective surfaces or shadows in the
image as you will need it to answer the question.
```

Long steering prompt

```
You are answering visual questions about images where a primary
object may be completely occluded by a visible mask, box, or polygon.

Your task is to identify the hidden object, not the occluder and not
merely a shadow, reflection, surface, or visible nearby object.

Before answering, carefully inspect the whole image.  Pay special
attention to:  - shadows, reflections, mirrors, glass, water, shiny
surfaces, and other indirect visual cues; - whether those cues
physically correspond to the hidden object given position, lighting,
orientation, scale, and scene geometry; - surrounding context, but
only as secondary evidence when direct visual cues are insufficient.
```

```
Do not rely mainly on common-sense priors if a shadow or reflection
provides more specific evidence.  Do not answer "shadow" or
"reflection" unless the question specifically asks for the visual
phenomenon itself; answer the object casting the shadow or being
reflected.

Give your best single answer.  Be specific enough to distinguish the
object when the image supports it, but do not over-specify uncertain
details.
```

Setup. We evaluate both steering prompts on all 5 models in the base condition, and additionally GPT-5.4 and Claude Opus 4.6 across thinking, and tool-use settings. The long steering prompt is only applied in the thinking and tool-use settings.

A.7 Inpainting Prompts

Two Stages. The inpainting intervention (Section 3, "Inpainting Multi-step Execution") has two stages. In the first, the model is asked to inpaint the occluded region while leaving the rest of the image unchanged; GPT-5.4 uses its native image-generation tool and Gemini uses the image-generation variant of the same model family. In the second, a fresh call to the model shows the occluded original together with the generated infill and asks the model to judge the infill's physical consistency before re-answering the identification question. The second-stage answer is scored with the same judges as the main evaluation.

Inpainting Stage 1: Fill Occluded Region

```
This image has a black region covering something.  Generate a new
version of this image with the hidden object naturally visible in
place of the black region, keeping everything else in the image
exactly the same.
```

Inpainting Stage 2: Confront and Re-Answer

```
The first image is the original:  a region of it is covered by a
solid mask, hiding an object.  The second image is a generated
version of that image, where the hidden object has been filled in.

Does this infill make sense physically, given the properties of
reflections and shadows?  Now, what is the occluded object?  Be as
specific as you can.
```

B Performance Across Occlusion Categories

Test Design. The category structure of OccludeBench (specular reflection, diffuse reflection, shadow on the direct side; technology, household, culture, activities, nature on the inferential side) raises the natural question of whether the direct–inferential gap is really a gap between image *types*, or whether it is driven by one or two unusually difficult subcategories. To rule the latter out, we run an omnibus chi-square test of independence for each model, separately on direct and inferential images, with category as the grouping variable and per-image correctness as the outcome. The null hypothesis is that accuracy is uniform across the cue subcategories within an image type.

Takeaway. The direct–inferential gap is not driven by one or two difficult cue subcategories. Because accuracy is roughly uniform across categories within each image type, the gap reflects a difference in evidence requirements rather than in image difficulty, semantic domain, or annotation noise.

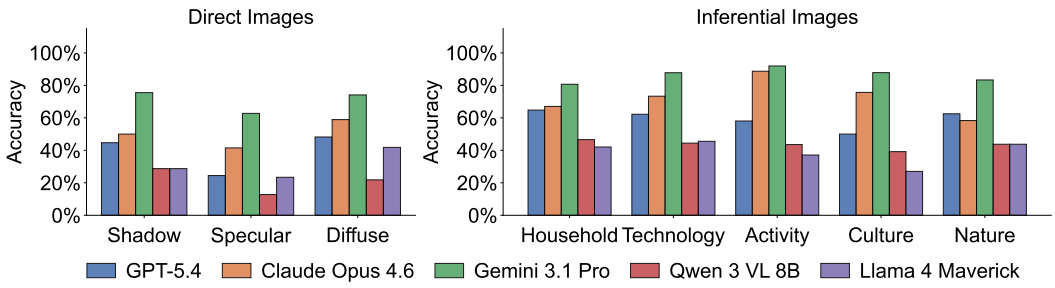


Figure 9. Per-category accuracy across occlusion cue categories for each model, separated by image type. VLMs do not exhibit statistically significant performance differences across subcategories within the direct (left) and inferential (right) image categories, suggesting that any semantic differences within the dataset do not have a major impact on performance.

Model	**Direct** p	**Inferential** p
GPT-5.4	**0.0037**	0.7488
Claude Opus 4.6	0.2020	0.1242
Gemini 3.1 Pro	0.2275	0.7190
Qwen 3 VL 8B	0.0557	0.9393
Llama 4 Maverick 17B	0.0524	0.2715

Table 5. Omnibus chi-square p-values testing whether accuracy varies across occlusion cue categories. Bold indicates $p < .05$.

C Dataset Breakdown and Sensitivity Analysis

Dataset Breakdown and Accuracy. Out of 378 images, 353 (93.4%) are collected from the Internet, 19 (5.0%) come from Visual Genome, and 6 (1.6%) are taken by the authors. The Visual Genome and author-taken images were included as a small source-diversity check rather than as equally sized comparison groups. Model accuracy is broadly similar across the three sources as shown in Table 6.

Model	**Internet**	**Visual Genome**	**Author-taken**	$\chi^2 p$
gemini-3.1-pro	75%	79%	67%	0.83
claude-opus-4-6	58%	58%	0%	**0.016**
gpt-5.4	46%	26%	50%	0.23
Qwen3-VL-8B	30%	5%	33%	0.068
Llama-4-Maverick	34%	5%	0%	**0.008**

Table 6. Chi-square p-values and accuracy across different image sources. Bold indicates $p < .05$.

Annotator Disagreement. After annotating images individually, we reviewed the annotated images and discussed whether the cues in the images are sufficient to identify the occluded object. We ended up discarding 41 images due to the annotator disagreement, which results in a 9.8% rejection rate. We did not have any disagreements on whether the image should be classified as inferential or direct, because it only depends on whether a shadow or reflection is the primary cue in the image.

Sensitivity to Bounding Box Properties. Annotators unanimously agreed that the occluded mask only occupies the object and has limited blocking of other objects, i.e., no part of the object is leaking or occluding other objects fully, and there should be no artifacts introduced from the manual annotation.

We change the background color of the polygon as well as the color of the border to see if that affects the accuracy, and find that the accuracy is not sensitive to the changes, as shown in Table 7.

Model	Category	n	Original Box	New Box (Blue Interior, Yellow Border)	Δ
gpt-5.4	Overall	378	49.5%	50.0%	+0.5
gpt-5.4	Direct	197	37.6%	40.1%	+2.5
gpt-5.4	Inferential	181	62.4%	60.8%	−1.7
claude-opus-4-6	Overall	378	63.0%	60.3%	−2.6
claude-opus-4-6	Direct	197	51.8%	46.2%	−5.6
claude-opus-4-6	Inferential	181	75.1%	75.7%	+0.6

Table 7. Comparison of accuracy between original bounding boxes and new bounding boxes by model and reasoning category.

D Difficulty Distribution Across Question Categories

To verify that our benchmark does not conflate category membership with question difficulty, we examine how per-image human accuracy is distributed within the *direct* (physical occlusion) and *inferential* (non-physical) question sets.

Neither category is dominated by polarized difficulty. Figure 10 shows the distribution of per-image accuracy across all 378 benchmark images (197 direct, 181 inferential), where per-image accuracy is defined as the fraction of Prolific participants who answered that image correctly. Both distributions span the full difficulty range, from images where nearly no participant answered correctly to images where nearly all did, with substantial mass in the middle. This indicates that neither category consists predominantly of trivially easy or impossibly hard questions.

The two categories do not differ significantly in difficulty. The mean per-image human accuracy is 63.4% for direct questions and 68.5% for inferential questions. A Mann-Whitney U test on the per-image accuracy distributions finds no statistically significant difference between the two categories ($U = 15977.5$, $p = 0.079$). Figure 11 further illustrates this: when images are ranked by difficulty, direct and inferential questions are interleaved throughout the difficulty spectrum rather than clustered at opposite ends.

Nearly every image is solvable. As a check that the benchmark does not rely on intrinsically unsolvable examples, at least one human participant correctly identified the occluded object in 94.2% of images (356/378), including 93.9% of direct and 94.5% of inferential examples.

Lexical Analysis of Direct and Inferential. We check the lexical prevalence of answer words for both direct and inferential questions to see if one is more difficult than the other from a textual perspective using metrics from Brysbaert et al. (2014). We find no difference in the concreteness of direct and inferential questions, as shown in Table 8.

Metric	Direct	Inferential
Zipf freq (Higher = more common)	4.48	4.22
Concreteness (1 = more abstract, 5 = more concrete)	4.65	4.54
Prevalence (fraction of people who know the word)	1.000	0.996
Out-of-norms (lower = fewer rare terms)	0.035	0.067

Table 8. Comparison of lexical metrics between direct and inferential categories.

Taken together, these results indicate that observed performance gaps between direct and inferential questions in model evaluations reflect genuine differences in task type rather than a confound introduced by systematically easier or harder questions in one category.

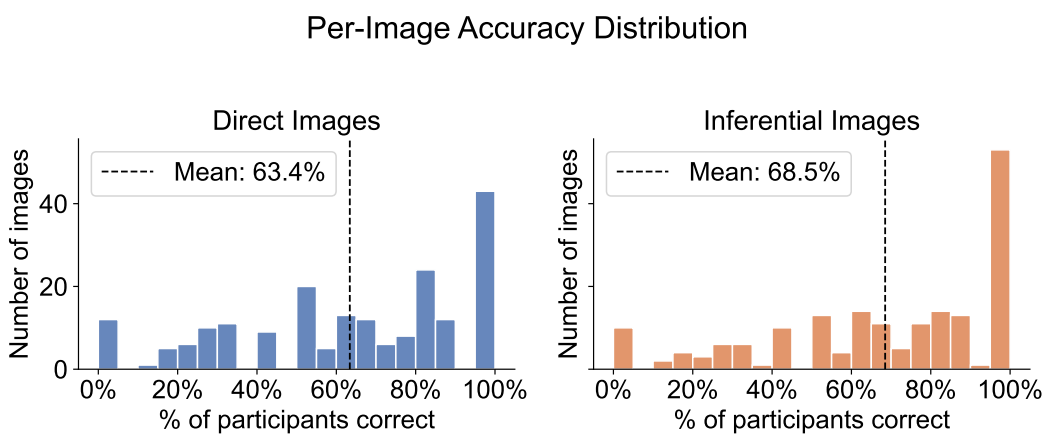


Figure 10. Per-image human accuracy distributions for direct and inferential questions. Both distributions span the full difficulty range, with no concentration at the extremes.

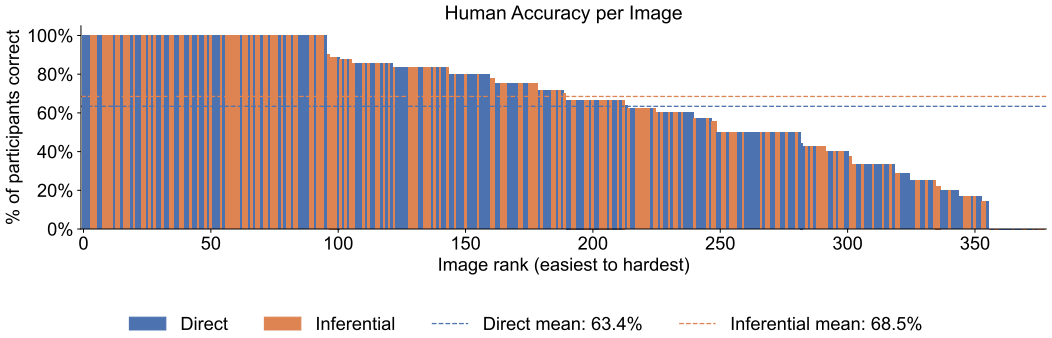


Figure 11. Images ranked by human accuracy (easiest to hardest), colored by category. Direct and inferential questions are interleaved across the full difficulty spectrum. Dashed lines indicate per-category means (direct: 63.4%, inferential: 68.5%).

E Tool Use Analysis

Per-Tool Frequencies. Figure 12 shows tool-usage frequency for GPT-5.4 and Claude Opus 4.6 on direct images. GPT-5.4 leans heavily on `crop` and `zoom` and repeatedly narrows its field of view; Claude Opus 4.6 distributes calls more evenly across the available tools and rarely crops, alternating between geometric transforms instead.

Budget Exhaustion. GPT-5.4 nearly always exhausts its 5-call budget; Claude Opus 4.6 rarely does. Read together with Figure 5 in the main text, which shows that Claude's accuracy *declines* monotonically from 62% at two calls to 28% at five, this suggests that additional tool calls reflect indecision rather than progress. For both models, the correct-answer group uses fewer tools on average than the incorrect-answer group.

Implication. Tool-call volume is a poor proxy for visual grounding effort. The target of a call (whether it lands on the correct projection region) matters more than the count, and even agentic harnesses with unrestricted code execution and high reasoning fail to close the direct–inferential gap.

F Details on Human Study

F.1 Human Study Interface

This appendix documents the participant-facing interface used to collect the human baseline reported in Section 2. Figures 13 and 14 show the loading screen and an exemplar question.

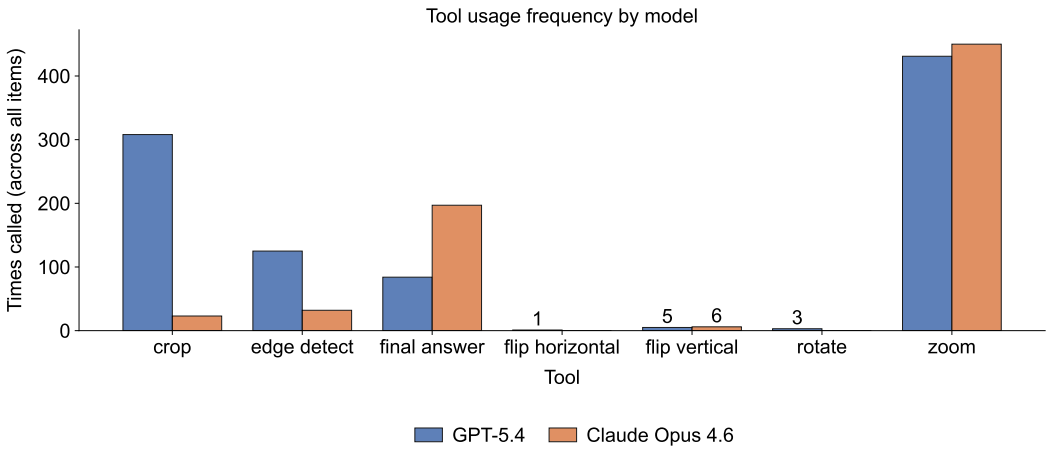


Figure 12. Tool usage frequency by model on direct images.

Loading Screen. Figure 13 shows the screen participants see before the task begins. It is where compensation, study scope, and consent information are first shown.

Per-Question Screen. Figure 14 shows the layout of a single question. Each participant sees six questions in total: three direct and three inferential, sampled approximately uniformly across the dataset, with item order randomized independently per participant. Items are shown one at a time as an image plus a free-form text box, matching the open-ended response format used for the model evaluations.

Compensation. Participants are recruited and paid through Prolific (Prolific, 2025) at $12.50/hour, as reported in Section 2. The median completion time of two minutes thirty seconds is consistent with the per-question design (six short free-form responses) and was used to set the rate.

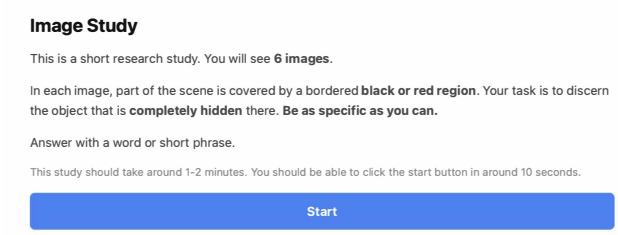


Figure 13. Loading screen of the human baseline study interface.

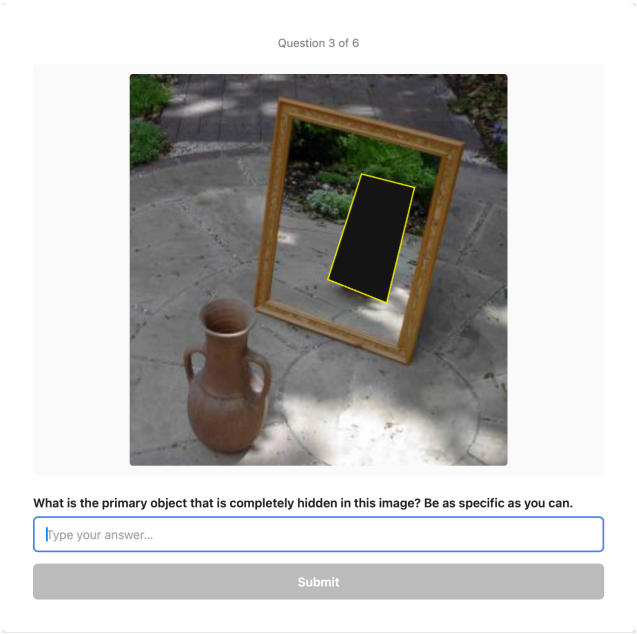


Figure 14. Example question shown to participants in the human baseline study.

F.2 Learning Effects in the Survey

We order each participant's six responses by timestamp and regress accuracy and response time on question position. Accuracy is stable across positions as shown in Table 9, staying within a 62–70% band with no significant increasing/decreasing trend (Cochran-Armitage p = 0.108), and response times are flat throughout. Thus, we conclude that allowing participants to answer multiple questions in the survey does not improve performance.

Q	Accuracy
1	0.699
2	0.653
3	0.642
4	0.651
5	0.673
6	0.620

Table 9. Accuracy and sample size (n) across the order of questions answered by humans.