In-Context Learning Can Help Vision-Language Models Overcome Training Priors

Kun Wang **Xindi Wu** **Sanghyuk Chun** **Olga Russakovsky** **Esin Tureci**
Princeton University
{bw5889, xw5676, sanghyukc, olgarus, eturci}@princeton.edu

Abstract

Vision-language models (VLMs) learn strong statistical regularities during training, which can make them fail to properly respond to visual evidence when input images violate those regularities. It is challenging to disambiguate these prior-induced errors from errors caused by the model missing the underlying visual capabilities. In this work, we use controlled visual in-context learning to encourage the model to focus on the visual evidence, and demonstrate that this alone is frequently sufficient to correct the errors – suggesting that the models in fact do possess the relevant visual capabilities. Specifically, we show that across five vision-centric tasks, demonstrations yield large gains on prior-conflicting counterfactual examples ($+27.2\%$ for Qwen3-VL 235B and $+15.1\%$ for Qwen3-VL 32B) while leaving accuracy on prior-aligned examples nearly unchanged. These findings generalize across five VLM families with gains from $+9.7\%$ to $+30.3\%$. We analyze visual representations and find that demonstrations indeed redirect attention toward grounding cues while reducing reliance on cues that support the learned prior. Finally, we find that in-context learning is limited when the underlying visual capability is weak or absent. Together, these results show that current VLMs possess visual grounding capabilities that standard evaluations may overlook, underscoring the need for controlled evaluation design in diagnosing and improving model capability. Code: `https://github.com/princetonvisualai/vlm_icl_tp`.

1 Introduction

In humans, visual intelligence necessitates grounding abstract knowledge and reasoning directly within the perceived image. Yet, vision-language models (VLMs) frequently falter when an image departs from the dominant patterns of their training data [1–4]: when shown an image depicting a three-legged peacock and asked how many legs are visible, models often answer "2" as shown in Figure 1(a). Existing work addresses these failures by intervening mechanistically [5–8], changing training procedures [9–12], or improving visual input preprocessing [13–15].

These approaches aim to address visual grounding failures by providing models with stronger visual signals and capabilities, treating such failures as evidence that the necessary capabilities are missing. Recent work, however, shows that the relevant visual evidence is already encoded in the internal representations of VLMs [16, 17]. This leaves an open question: are these failures caused by the model's inability to incorporate visual information, or by a stronger learned expectation that overrides the accessible grounding signal?

In-context learning (ICL) provides a natural test: providing demonstrations of other animals with answers (e.g., a five-legged dog, a two-legged parrot) informs the model that the task is to count the legs visible in the image [18]. If errors persist despite such demonstrations, the model likely lacks the visual capability to count legs in the first place. If demonstrations correct the error, the model can already count legs but its learned knowledge about the animal overrides the visual evidence.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

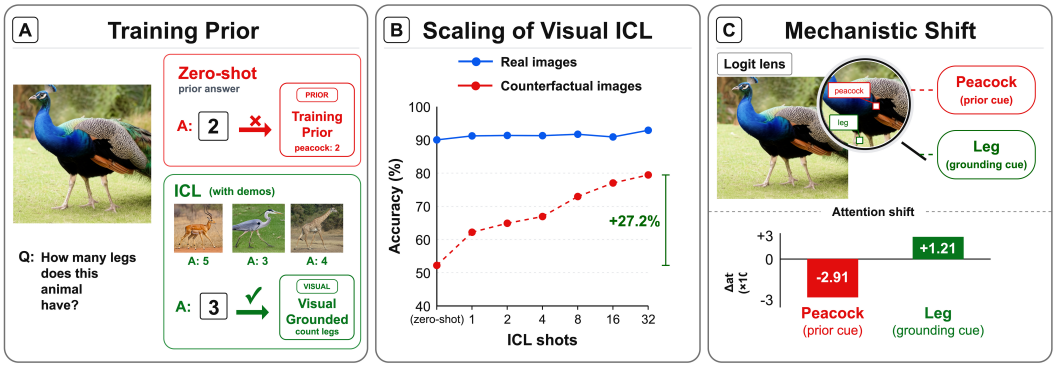


Figure 1: **Visual ICL overcomes training prior dominance. (A)** A VLM answers "2" for a three-legged peacock under zero-shot prompting; label-balanced demonstrations shift the answer to the visually grounded "3." **(B)** Scaling ICL shots yields a +27.2 pp accuracy gain on counterfactual images while real images remain near ceiling. **(C)** Mechanistically, demonstrations redirect attention from the prior cue ("peacock") to the grounding cue ("leg").

Although ICL is well established as a steering mechanism in language models [19–23], prior work has reported that visual ICL lags behind its language counterpart [24, 25]. These studies have made important progress, but they leave open the question of whether demonstrations can shift a model toward visually grounded answers. Many existing studies test surface-level image–label associations, such as mapping pictures of lions to the label "dog" [24, 25], where the task to be learned does not focus on visual perception and instead can be collapsed into a text-only mapping problem. Others retrieve demonstrations from general VQA pools [26–29], where loose query–demo relevance tends to reduce ICL to format learning. Thus, it is still an open question whether, and for which tasks, demonstrations can actually move a model toward grounded visual evidence.

We address this question by using ICL as a diagnostic tool to study a phenomenon we call *training prior dominance*, where a model's learned statistical regularity from training overrides visual evidence during generation. We construct a benchmark that separates prior-consistent from prior-conflicting examples and supports label-balanced demonstrations, allowing us to control the demonstration context through task-relevant examples and answer choices. With these controls, we find that many visual grounding failures are not caused by an inability to incorporate visual information, but by overpowering training priors that models can suppress through ICL. We further analyze this effect mechanistically, showing how demonstrations shift the models' attention toward visually grounded evidence, and we test whether demonstrations can also teach capabilities that are absent. Together, these results suggest that addressing prior-driven failures does not require architectural modification or enhanced visual features; effective mitigation can instead focus on managing the influence of training priors during inference to keep the model anchored to the visual evidence.

Our main contributions are: (1) a controlled benchmark that separates prior-consistent from prior-conflicting examples and supports label-balanced demonstrations for visual ICL; (2) empirical and mechanistic evidence that visual ICL improves accuracy most on prior-conflicting examples, distinguishing a visual grounding shift account from simple format learning; and (3) diagnostic analyses that test whether visual ICL teaches new visual capabilities or reveals capabilities already available but dominated by training priors.

2 Problem Setting

To test whether ICL overcomes training prior dominance, we compare examples where learned expectations agree with visual evidence to examples where they conflict.

Training Priors and Visual Grounding. We use **training prior** to denote a statistical regularity learned from the training distribution that biases predictions on a downstream task. This includes language and world-knowledge priors [1, 30, 31], such as typical object attributes, as well as visual or spatial priors [32, 33], such as common poses and layouts.

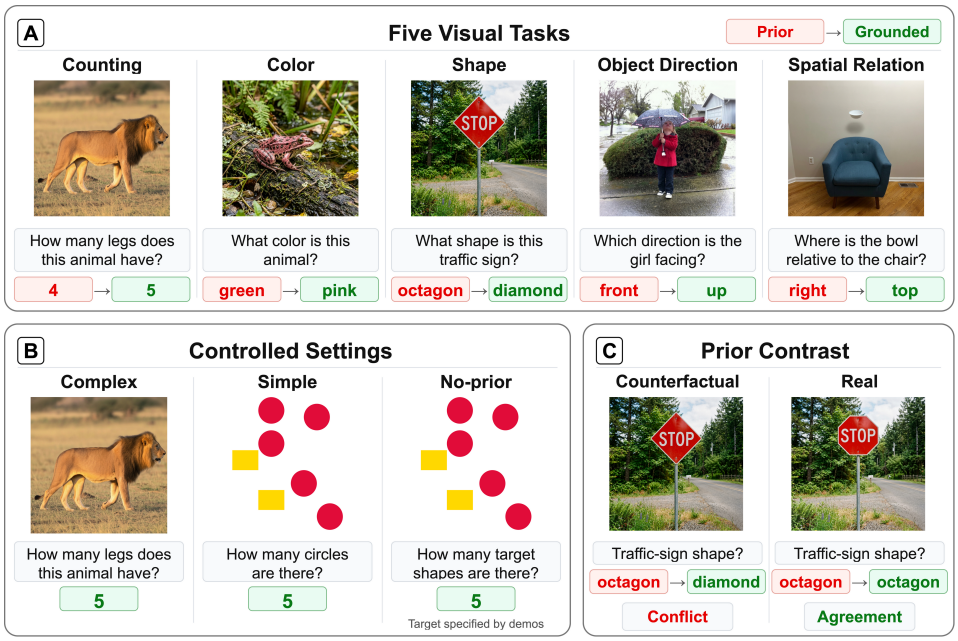


Figure 2: **Data design.** (A) Our benchmark covers five vision-centric tasks, each contrasting a prior answer with the grounded visual answer. (B) Each task contains complex, simple, and no-prior settings to separate prior override from ordinary visual task learning. (C) We probe the training prior by comparing counterfactual examples, where the prior and visual answers conflict, with real examples, where they agree.

A model can answer an animal-leg question by recognizing the animal and recalling its typical leg count, or by localizing and counting its visible legs. These compositions of capabilities agree on a typical two-legged peacock but disagree on the three-legged peacock in Figure 1(a).

Prior-Consistent and Prior-Conflicting Examples. For an image–question pair (I, Q), we call the image-supported answer the *grounded answer*. When a learned regularity favors an identifiable answer, we call it the *prior answer*. An example is *prior-consistent* when these answers agree and *prior-conflicting* when they differ. Prior-consistent examples alone cannot distinguish a correct shortcut from visual grounding.

Visual ICL as a Diagnostic. Given a query (I_q, Q_q) and k demonstrations $\mathcal{D} = \{(I_i, Q_i, A_i)\}_{i=1}^{k}$, visual ICL predicts an answer A without changing model parameters. We view ICL as shifting probability mass over latent compositions of capabilities [34–36]:

$$p(A \mid I_q, Q_q, \mathcal{D}) = \sum_{s \in \mathcal{S}} p(A \mid I_q, Q_q, s)\, p(s \mid I_q, Q_q, \mathcal{D}), \tag{1}$$

where s denotes a candidate capability composition and $\mathcal{S}$ is the space of such compositions. We use s_{prior} and s_{ground} to denote a prior-based composition and a visually grounded composition, respectively. Zero-shot prompting may place high mass on s_{prior} when the learned regularity is salient. Demonstrations that consistently exhibit visual grounding can shift mass toward s_{ground}.

This view predicts that ICL gains should concentrate on prior-conflicting examples if demonstrations help override training priors, rather than merely teach the response format. Section 3 develops this diagnostic into a benchmark with closed-label visual tasks, label-balanced demonstrations, and controlled settings for comparing prior-consistent and prior-conflicting examples.

3 ICL Construction for Visual Perception and Reasoning

Prior visual ICL benchmarks fail to separate visual grounding from a learned training prior or response-format imitation [28, 24, 27], while prior-probing benchmarks document training prior reliance but lack the controls needed for ICL evaluation [30, 1, 31, 2]. Our work bridges these gaps.

Each example carries a *grounded answer* determined by image evidence and, when applicable, a *prior answer* favored when that evidence is ignored; comparing ICL gains on *prior-consistent* examples (answers agree) with *prior-conflicting* examples (answers disagree) isolates training prior override from format or label learning. To ensure this diagnostic is valid, we use label-balanced demonstrations so gains cannot be explained by frequency shortcuts (Section 3.1), difficulty-controlled settings that isolate how the training prior affects performance (Section 3.2), and vision-centric closed-label tasks where the task requires visual grounding (Section 3.3).

3.1 Data Design Requirements

The benchmark uses closed-label, vision-centric tasks with balanced answer labels and matched control settings, so the accuracy gains from k-shot ICL over the zero-shot baseline cannot be explained solely by response-format imitation or answer-frequency shifts. We construct each example as an (I, Q, A) triple and use this label-balanced candidate pool for demonstration sampling.

To probe the training prior with minimal confounding factors, we use controlled counterfactual editing: we change only the queried visual property while preserving the object identity, scene context, and prompt format. For example, Figure 2 (C) edits an octagonal stop sign into a diamond-shaped sign while keeping the rest of the image fixed, making the grounded shape conflict with the canonical stop-sign prior. To avoid a source-domain confound, prior-consistent images are also generated or regenerated with Gemini and passed through the same validation, balancing, and downsampling pipeline as the prior-conflicting images. Two annotators independently label each complex image while blind to the intended label. We retain an image only if both annotators select the intended label and confirm that it meets the quality criteria. Appendix A.2.1 gives the construction procedure, task-specific rejection criteria, and validation statistics.

3.2 Settings

To examine visual ICL and its ability to override the training prior, we use difficulty-controlled settings that vary whether a prior answer is identifiable. Each task is therefore evaluated in three settings of increasing complexity, shown in Figure 2 (B). The *simple* setting tests the visual operation on abstract stimuli without an identifiable prior answer. The *no-prior* setting further abstracts the prompt by replacing object names with placeholders such as "target object," following the ICL view of demonstrations as task-rule evidence [34, 37, 21, 22]; we therefore interpret its 0-to-k gains as in-context task specification with language priors removed, not as an unambiguous zero-shot visual baseline. The *complex* setting is the main evaluation: examples preserve object identity and task format while varying whether the grounded answer agrees with the prior answer. In *real* images the two agree; in *counterfactual* images the queried visual evidence is changed so that the prior answer is wrong. Complex examples thus test training prior override, while simple and no-prior examples control for ordinary visual competence and task-format learning.

3.3 Tasks

We evaluate five closed-label visual tasks: counting, color identification, shape identification, object direction, and spatial relation. Each of these tasks supports exact-match scoring, label-balanced demonstrations, and controlled prior-conflicting examples that change only the queried visual evidence. Together, they cover two shortcut regimes relevant to training prior dominance: counting, color identification, and shape identification test language or world-knowledge priors tied to object identity, while object direction and spatial relation test visual-configuration priors tied to common poses and layouts. For each task, the answer label set is fixed across settings, and accuracy is measured by exact match after task-specific normalization. Table 2 lists sources, labels, and counts for each task and setting, including the counting datasets used in Section 4.3. The original complex setting contains 1,160 examples: 650 real (56.0%) and 510 counterfactual (44.0%). We also evaluate an expanded version with approximately 400 complex examples per task; all additions are counterfactual (Appendix A.3.3).

Counting. The task asks for the number of visible animal legs; the complex setting violates category priors such as two-legged birds or four-legged mammals [2].

Color Identification. The task asks for the displayed animal color, with animals rendered in colors that conflict with their canonical real-world coloration [1].

Shape Identification. The task asks for the displayed traffic-sign shape, where the sign identity remains recognizable but the rendered shape may differ from the canonical one [1].

Object Direction. The task asks for the camera-relative orientation (front, back, left, right, up, down) of an object whose canonical pose acts as a shortcut; the complex setting emphasizes rare orientations such as up and down [33].

Spatial Relation. The task asks for the relation of a target object to an ordered reference object, where the depicted layout conflicts with the most common configuration [32].

The selected source entries contain zero up/down examples for object direction and zero top/bottom examples for spatial relation. These source-label gaps motivate the rare-configuration tests; Appendix A.2.1 reports model-side prior diagnostics, since model-specific training frequencies are unavailable.

3.4 In-Context Demonstrations

Demonstrations are sampled from the same task and setting as the query unless otherwise stated, with query images excluded from the demonstration pool. We evaluate 0, 1, 2, 4, 8, 16, and 32 shots. For each nonzero shot count, we pre-sample 10 fixed contexts per task and setting and share them across queries. We report mean accuracy and standard deviation across these contexts; each query-context pair is scored from one sampled output, so the standard deviation also includes decoding noise. Contexts are label-balanced whenever possible, and complex-setting contexts also balance real and counterfactual examples when possible; this controls for label-frequency effects and prevents gains from being explained by changes in the marginal answer distribution. All shot counts use the same instruction template, with only the number of demonstrations changed; Appendix A.2.2 gives full prompt and sampling details, and Figure 8 lists task-specific templates.

3.5 Models and Inference

For performance analysis, we evaluate Qwen3-VL variants from 2B to 235B parameters [10]. To test generality beyond one model family, we also evaluate Qwen2.5-VL [9], Qwen3.5 [38], InternVL3.5 [39], and Gemma-4 [40]. Mechanistic experiments cover Qwen3-VL 4B and 32B and Gemma-4 E4B and 31B; masking experiments use the larger model from each family. All models use the same prompt templates, demonstration sets, and answer normalization. Appendix A.2.2 provides checkpoints and inference backends, sampling parameters, and normalization rules.

4 In-Context Learning Overcomes Training Priors

Building on our diagnostic framing of ICL, this section asks when and why visual demonstrations help VLMs suppress training priors and rely on image-grounded evidence. We ask whether demonstrations improve performance in general and on prior-conflicting examples, how they change the use of visual evidence in VLMs, and whether they can strengthen new or weak capabilities.

4.1 Visual ICL Improves Performance on Prior-Conflicting Examples

We begin with the first diagnostic question: do demonstrations induce format learning, or help the model override a learned training prior when visual evidence conflicts with the prior-favored answer? If ICL mainly induces format learning, then examples with the same prompt structure and answer space should improve similarly. If ICL overrides training priors, then gains should concentrate where the visually grounded answer contradicts learned regularities.

Controlled visual ICL scales with model size and shot count. Figure 3 reports accuracy across shot counts for Qwen3-VL models from 2B to 235B parameters. Accuracy increases with more demonstrations across all three settings, and the larger models continue to gain even from 16 to 32 shots. This trend shows that VLMs can use visual demonstrations as in-context task evidence, with stronger scaling at larger model sizes.

The setting-level comparison helps separate task and format learning from training prior override. In particular, we observe that zero-shot accuracy on *simple* is already high. Given the same task

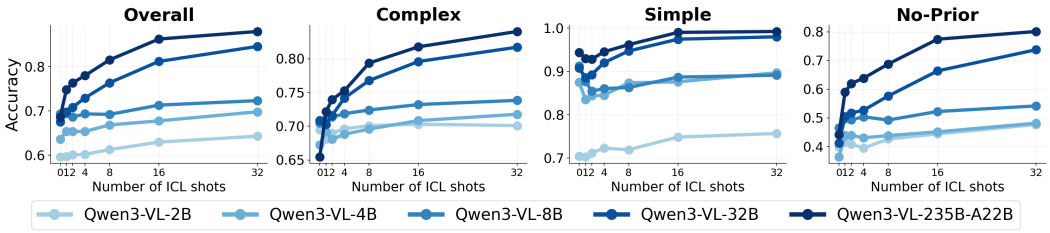


Figure 3: **Qwen3-VL scales with in-context demonstrations.** Accuracy vs. number of in-context demonstrations for Qwen3-VL model family on complex, simple, and no-prior settings across tasks.

format between settings, the larger gains on *complex* than on *simple*, especially for the 32B and 235B models, suggest that demonstrations do more than teach the response format. They help large models perform task learning and may encourage the use of a visually grounded composition.

To validate that this trend is not specific to Qwen3-VL, we extend the comparison to five VLMs at roughly 30B scale (Table 1). Every family improves from 0-shot to 16-shot prompting on the aggregate complex setting, with gains ranging from $+9.7\%$ for Qwen3-VL to $+30.3\%$ for Qwen3.5. At the per-task level (Table 7), complex deltas remain positive in 23 of 25 task-family cells at $\sim$30B. Full accuracy results across families, tasks, and settings are in Appendix A.3.

Table 1: In-context scaling of $\sim$30B model families on the complex setting. Accuracies and deltas are reported with % and micro-averaged over query entries.

Model	0-shot	16-shot	Δ_{16-0}
Qwen3-VL 32B	69.9%	79.6%	+9.7%
Qwen2.5-VL 32B	49.1%	64.7%	+15.5%
Qwen3.5 27B	44.1%	74.5%	**+30.3%**
InternVL3.5 38B	52.4%	70.6%	+18.3%
Gemma-4 31B	**75.2%**	**86.6%**	+11.5%

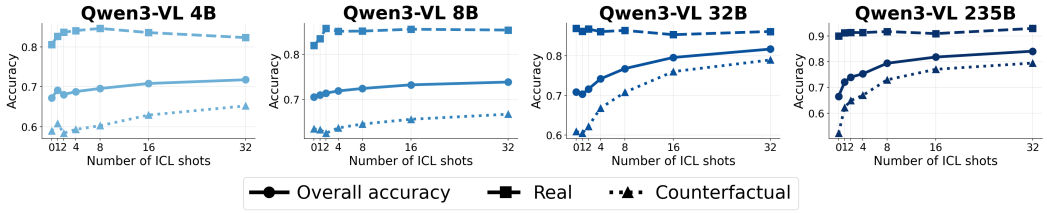
ICL gains concentrate on counterfactual examples. We next decompose this performance gain of ICL into real and counterfactual examples in the complex setting. As shown in Figure 2 (C), the two sets of examples differ only in whether the visually grounded answer agrees with or contradicts the learned prior. If ICL mainly induces format learning, 0-to-16-shot gains should be similar on both subsets. If ICL helps suppress training prior dominance, gains should concentrate on counterfactual examples.

Figure 4(a) supports the training prior-dominance account: the asymmetry is between real and counterfactual gains within each model, not between the 32B and 235B curves. For the 235B model, 16-shot prompting raises counterfactual accuracy by $+27.2\%$ while leaving real accuracy effectively unchanged ($+2.9\%$). The 32B model shows the same asymmetry (counterfactual: $+15.1\%$, real: -1.6%). At 4B and 8B, gains are smaller and less selective, suggesting that smaller models use demonstrations more for format learning than for overriding a learned training prior. Further results on small models' invariance to shuffled labels suggest that they barely condition on demonstration content [19] (Appendix A.3.4).

Figure 4(b) breaks the 0-to-16-shot change down by task. At 32B and 235B, counterfactual gains exceed real gains across all five tasks. Counting is the most challenging task in our benchmark and illustrates this most starkly: counterfactual accuracy rises by $+32.6\%$ at 32B and $+29.4\%$ at 235B while real accuracy edges down by 7.0% at 32B and 6.3% at 235B, a pattern we trace in Section 4.3 to ICL's dependence on a reliable underlying visual capability. Color identification, shape identification, and spatial relation follow the same pattern while maintaining performance on real examples, with gains scaling monotonically with model capacity. Object direction is the most scale-dependent task in our benchmark. At 32B and 235B it produces the largest absolute counterfactual gain ($+39.8\%$, $+21.0\%$), but at 8B, 16-shot counterfactual accuracy drops below 0-shot. This scale-dependence is inconsistent with a simple label-availability hypothesis, which would improve all model sizes, and consistent with a capability-dependent override of the training prior.

Gains persist on the expanded benchmark. On the expanded complex setting, Qwen3-VL 32B improves from 0 to 16 shots on all five tasks, with gains of 7.2 to 24.8 percentage points (Table 9). The real share falls from 56.0% to approximately 33%, changing the subset weights in overall accuracy; Appendix A.3.3 reports dataset statistics and the available context standard deviations.

(a) Aggregate ICL scaling.



(b) Per-task ICL diagnostics.

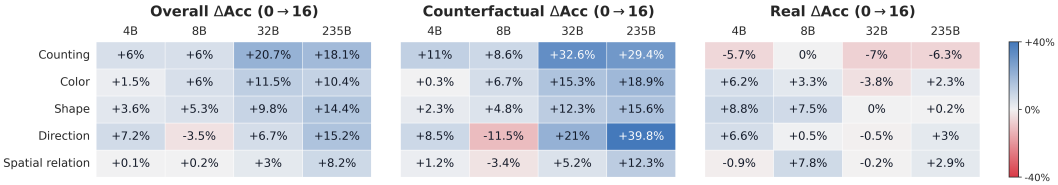


Figure 4: **Scaling and diagnostic analysis of Qwen3-VL across model sizes on the *complex* setting. (a)** Aggregate accuracy on the real and counterfactual subsets with varying numbers of in-context learning (ICL) shots, for the 4B, 8B, 32B, and 235B Qwen3-VL models. **(b)** Per-task 0-to-16-shot accuracy change (ΔAcc) on the overall, counterfactual, and real subsets, broken down by task (counting, color, shape, direction, spatial relation) and model size. Positive values (blue) indicate tasks where ICL improves accuracy; negative values (red) indicate degradation.

4.2 Mechanistic Evidence for Training Prior Override Through Visual Grounding

Sec. 4.1 identifies where visual ICL helps: gains concentrate on counterfactual examples, where the image contradicts the model's learned training prior. We next test whether demonstrations shift answer generation from image tokens supporting the learned training prior toward tokens supporting the visually correct answer.

We define *prior cues* as image tokens whose decoded semantics support the answer favored by the learned training prior, and *grounding cues* as tokens whose decoded semantics support the correct answer in the current image. To identify these cues, we use logit lens [41] and LatentLens [42] to assign approximate semantic labels to image-token representations. From the running three-legged peacock example in Figure 1 (c), tokens decoded as "peacock" are prior cues, while tokens decoded as visible legs are grounding cues. For object direction and spatial relation, no single prior answer is defined, so we use the object named in the caption as a proxy prior cue. This cue may correlate with the answer without specifying a direct shortcut. In general, prior and grounding cues need not be mutually exclusive: a single visual token can decode to both kinds of semantics. We resolve such cases by assigning any overlapping token to the prior-cue set (Appendix A.4), so the two cue sets are disjoint by construction.

With this setup, we test our hypothesis that visual ICL works by reweighting evidence: as the number of demonstrations grows, answer generation should pull less signal from prior-cue tokens and more signal from grounding-cue tokens. The token-disjoint construction above lets us measure each pull separately. Figure 5 lays out the two tests we use: an attention analysis that compares 0-shot and 16-shot weight on the two cue groups, and a masking ablation that asks whether removing prior-cue tokens reproduces the effect of demonstrations.

Demonstrations shift attention toward grounding cues. We first test whether demonstrations change the visual evidence used during answer generation. If ICL overrides training prior dominance by selecting a visually grounded composition, then answer tokens should attend less to prior cues and more to grounding cues. We measure this shift by comparing answer-token attention under 0-shot and 16-shot prompting, averaging attention over prior cues, grounding cues, and all other image tokens.

Figure 6(a) shows this redistribution for Qwen3-VL 4B and 32B. Both models attend less to prior cues and more to grounding cues after demonstrations. Gemma-4 E4B and 31B show the same

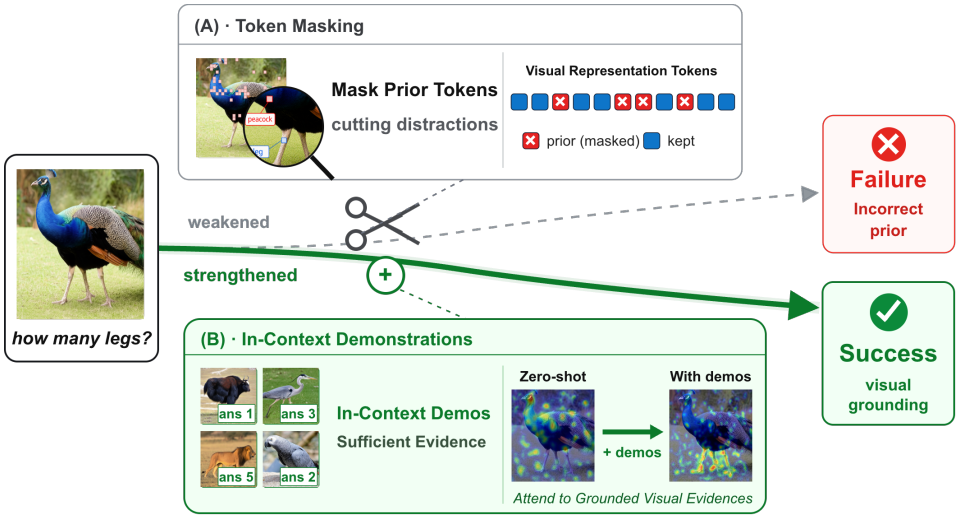
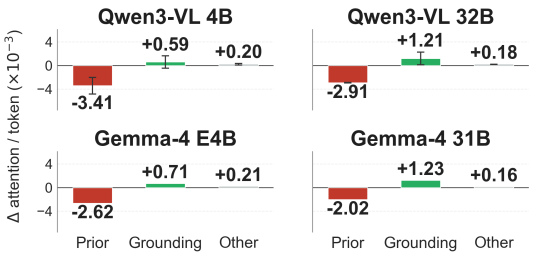


Figure 5: **Two complementary routes to visual grounding.** **(A)** Masking visual tokens that encode prior cues weakens the learned-training prior pathway. **(B)** In-context demonstrations strengthen attention to image-grounded evidence. Both interventions shift the prediction from a prior-driven error toward the visually grounded answer.

(a) Demonstrations shift answer-token attention



(b) Masking effects across model families

	Qwen3-VL 32B		Gemma-4 31B	
Task	**Random**	**Prior**	**Random**	**Prior**
Counting	−3.9%	+14.2%	−4.4%	+17.9%
Color ID	+3.5%	+15.0%	0.0%	+9.2%
Shape ID	−7.2%	+6.2%	−0.5%	+3.2%
Direction	−3.2%	+6.1%	−6.7%	+2.6%
Spatial rel.	+0.5%	+3.4%	−1.9%	+1.7%

Accuracy change from each model's 0-shot baseline, in percentage points.

Figure 6: **Demonstrations reduce reliance on prior cues.** (a) From 0-shot to 16-shot, VLMs shift answer-token attention away from prior cues and toward grounding cues. (b) On the complex setting with greedy decoding, prior-cue masking improves accuracy over both zero-shot and matched random-token masking for Qwen3-VL 32B and Gemma-4 31B on all five tasks. Table 13 reports all attention shifts and full masking accuracies.

pattern (Table 13(a)). These results are consistent with demonstrations redirecting answer generation toward visual evidence already represented in the image tokens.

Masking prior cues reduces prior-driven errors. The attention analysis shows that demonstrations redirect answer generation toward grounding cues, but attention shifts alone do not establish whether prior cues *cause* counterfactual errors. We therefore run two matched-budget ablations that block visual tokens during answer generation. The *prior-cue mask* blocks tokens flagged by LatentLens [42] as decoding to prior-cue semantics; the *random mask* blocks the same number of visual tokens drawn uniformly from non-cue locations. If zero-shot errors are driven by the semantic content of prior cues rather than by visual-token budget in general, the prior-cue mask should reduce the pull toward the prior-favored answer while the random control should not.

Figure 6(b) compares masking effects for Qwen3-VL 32B and Gemma-4 31B on the *complex* setting with greedy decoding. Prior-cue masking improves zero-shot accuracy and outperforms the matched random mask on all five tasks for both models. For Qwen3-VL 32B, the gains are 14.2 percentage points on counting, 15.0 on color identification, and 6.2 on shape identification. Gemma-4 31B gains 17.9, 9.2, and 3.2 points, respectively; its counting accuracy rises from 40.9% to 58.8%. Object direction and spatial relation also improve in both families, although the proxy cue makes the evidence

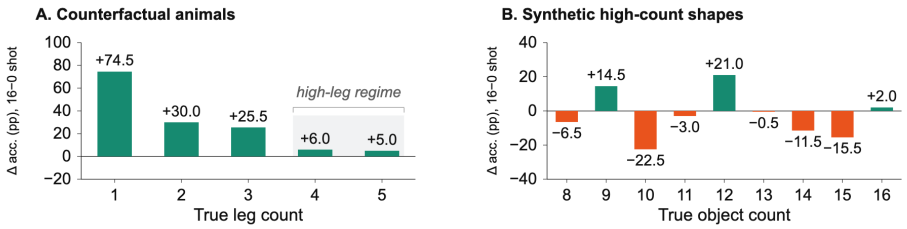


Figure 7: **ICL gains weaken for harder counting cases.** The plot shows the accuracy change from 0-shot to 16-shot prompting.

less direct for these tasks. Table 13(b) gives the full accuracies, including the corresponding 16-shot results.

The two model families provide consistent evidence for the proposed mechanism. Demonstrations shift attention from prior cues toward grounding cues, while the masking controls support a contribution of the identified prior cues to zero-shot errors. Together, these findings support the view that visual ICL shifts inference from s_{prior} toward s_{ground} by changing how the model uses available visual evidence (Sec. 2).

4.3 Visual ICL Relies on Reliable Underlying Visual Capabilities

We have shown that visual ICL helps when VLMs have access to relevant visual evidence but a training prior dominates the answer. These results raise a natural follow-up question about its limits: does ICL still help when the underlying visual capability itself is unreliable?

ICL gains weaken at higher leg counts. We first examine this question within counterfactual counting examples, breaking down ICL gains by the number of animal legs (Figure 7(a)). If prior conflict were the only obstacle, demonstrations should help uniformly across these examples, because each image still contains a misleading animal prior and a visually correct answer. Instead, ICL helps less for higher leg counts, especially 4- and 5-leg cases, where the model must rely on less robust visual counting. Overriding the training prior is therefore not sufficient on its own, which motivates a stronger hypothesis: ICL may help VLMs select a reliable capability that is already available, but it does not appear to make an unreliable capability robust given a few demonstrations.

Demonstrations do not make high-cardinality counting robust. Counting offers a direct test of this hypothesis because high-cardinality counting remains difficult for VLMs even when objects are simple and clearly visible[43–46]. We therefore test counting on 450 synthetic 2D-shape images, with 50 examples for each count from 8 through 16 (Table 2). These examples remove animal identity, biological plausibility, and learned animal priors, so the main challenge is the counting capability itself. If demonstrations teach robust high-cardinality counting, then demonstrations for a given count should consistently improve accuracy on test examples with that count. Contrary to this prediction, ICL gains vary across counts (Figure 7(b)), and some counts, such as 10 and 15, fall below zero-shot performance. Even after removing the confounding effect of learned priors, demonstrations do not reliably improve high-cardinality counting. These results support a capability-selection view: ICL can redirect a reliable capability suppressed by a prior, but does not consistently strengthen an unreliable capability from a few demonstrations.

5 Conclusion

We revisited and challenged the assumption that visual grounding failures in VLMs reflect missing visual capabilities. Using a controlled benchmark that separates prior-consistent from prior-conflicting examples and supports label-balanced demonstrations, we show that many of these failures are better understood as *training prior dominance*, where the model encodes the relevant visual evidence and the ability to utilize it but defaults to a stronger learned expectation when answering. Task-relevant multimodal demonstrations reliably suppress this effect on prior-conflicting examples, and our mechanistic analyses confirm that demonstrations shift the model's reliance from priors toward

visually grounded evidence. However, demonstrations do not reliably strengthen weak capabilities. This distinction also matters beyond accuracy. When a VLM defaults to priors instead of perception, those priors can encode stereotypical biases; identifying when a failure is correctable by re-anchoring models to visual evidence is essential for fairness in downstream applications [47, 48, 4].

These findings reframe a subset of visual grounding failures as test-time correctable. ICL thereby serves as a diagnostic that separates *training prior dominance*, which test-time intervention can address, from capability absence, which still calls for changes to data, training, or architecture [12, 11, 10]. Furthermore, visual ICL is a training-free but not compute-free path to adaptation. The interleaved multimodal 32-shot setting consumes substantial context, increases memory use, and adds latency. Practical continual-adaptation systems will therefore need to select demonstrations carefully, compress or cache visual contexts, or distill successful contexts into model weights [49–52].

Acknowledgments and Disclosure of Funding

This work is supported by the Princeton First Year Fellowship, the Francis Robbins Upton Fellowship awarded to SC, and the Apple Scholars in AI/ML PhD Fellowship awarded to XW. We are grateful to Allison Chen and Tyler Zhu for insightful discussions and valuable feedback that helped shape this work.

References

[1] Tiange Luo, Ang Cao, Gunhee Lee, Justin Johnson, and Honglak Lee. Probing visual language priors in vlms. *arXiv preprint arXiv:2501.00569*, 2024.

[2] An Vo, Khai-Nguyen Nguyen, Mohammad Reza Taesiri, Vy Tuong Dang, Anh Totti Nguyen, and Daeyoung Kim. Vision language models are biased. *arXiv preprint arXiv:2505.23941*, 2025.

[3] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9568–9578, 2024.

[4] Mohammad Asadi, Jack W O'Sullivan, Fang Cao, Tahoura Nedaee, Kamyar Fardi, Fei-Fei Li, Ehsan Adeli, and Euan Ashley. Mirage the illusion of visual understanding. *arXiv preprint arXiv:2603.21687*, 2026.

[5] Seil Kang, Jinyeong Kim, Junhyeok Kim, and Seong Jae Hwang. Your large vision-language model only needs a few attention heads for visual grounding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9339–9350, 2025.

[6] Jianfei Zhao, Feng Zhang, Xin Sun, Chong Feng, and Zhixing Tan. Tell model where to look: Mitigating hallucinations in mllms by vision-guided attention. *arXiv preprint arXiv:2511.20032*, 2025.

[7] David Wan, Jaemin Cho, Elias Stengel-Eskin, and Mohit Bansal. Contrastive region guidance: Improving grounding in vision-language models without training. In *European Conference on Computer Vision*, pages 198–215. Springer, 2024.

[8] Yanshu Li, Jianjiang Yang, Ziteng Yang, Bozheng Li, Ligong Han, Hongyang He, Zhengtao Yao, Yingjie Victor Chen, Songlin Fei, Dongfang Liu, et al. Make lvlms focus: Context-aware attention modulation for better multimodal in-context learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 6610–6618, 2026.

[9] Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025.

[10] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[11] Lingchen Meng, Jianwei Yang, Rui Tian, Xiyang Dai, Zuxuan Wu, Jianfeng Gao, and Yu-Gang Jiang. Deepstack: Deeply stacking visual tokens is surprisingly simple and effective for lmms. *Advances in Neural Information Processing Systems*, 37:23464–23487, 2024.

[12] Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, Gaël Liu, Francesco Visin, Kathleen Kenealy, Lucas Beyer, Xiaohai Zhai, Anton Tsitsulin, Robert Busa-Fekete, Alex Feng, Noveen Sachdeva, Benjamin Coleman, Yi Gao, Basil Mustafa, Iain Barr, Emilio Parisotto, David Tian, Matan Eyal, Colin Cherry, Jan-Thorsten Peter, Danila Sinopalnikov, Surya Bhupatiraju, Rishabh Agarwal, Mehran Kazemi, Dan Malkin, Ravin Kumar, David Vilar, Idan Brusilovsky, Jiaming Luo, Andreas Steiner, Abe Friesen, Abhanshu Sharma, Abheesht Sharma, Adi Mayrav Gilady, Adrian Goedeckemeyer, Alaa Saade, Alex Feng, Alexander Kolesnikov, Alexei Bendebury, Alvin Abdagic, Amit Vadi, András György, André Susano Pinto, Anil Das, Ankur Bapna, Antoine Miech, Antoine Yang, Antonia Paterson, Ashish Shenoy, Ayan Chakrabarti, Bilal Piot, Bo Wu, Bobak Shahriari, Bryce Petrini, Charlie Chen, Charline Le Lan, Christopher A. Choquette-Choo, CJ Carey, Cormac Brick, Daniel Deutsch, Danielle Eisenbud, Dee Cattle, Derek Cheng, Dimitris Paparas, Divyashree Shivakumar Sreepathihalli, Doug Reid, Dustin Tran, Dustin Zelle, Eric Noland, Erwin Huizenga, Eugene Kharitonov, Frederick Liu, Gagik Amirkhanyan, Glenn Cameron, Hadi Hashemi, Hanna Klimczak-Plucińska, Harman Singh, Harsh Mehta, Harshal Tushar Lehri, Hussein Hazimeh, Ian Ballantyne, Idan Szpektor, Ivan Nardini, Jean Pouget-Abadie, Jetha Chan, Joe Stanton, John Wieting, Jonathan Lai, Jordi Orbay, Joseph Fernandez, Josh Newlan, Ju yeong Ji, Jyotinder Singh, Kat Black, Kathy Yu, Kevin Hui, Kiran Vodrahalli, Klaus Greff, Linhai Qiu, Marcella Valentine, Marina Coelho, Marvin Ritter, Matt Hoffman, Matthew Watson, Mayank Chaturvedi, Michael Moynihan, Min Ma, Nabila Babar, Natasha Noy, Nathan Byrd, Nick Roy, Nikola Momchev, Nilay Chauhan, Noveen Sachdeva, Oskar Bunyan, Pankil Botarda, Paul Caron, Paul Kishan Rubenstein, Phil Culliton, Philipp Schmid, Pier Giuseppe Sessa, Pingmei Xu, Piotr Stanczyk, Pouya Tafti, Rakesh Shivanna, Renjie Wu, Renke Pan, Reza Rokni, Rob Willoughby, Rohith Vallu, Ryan Mullins, Sammy Jerome, Sara Smoot, Sertan Girgin, Shariq Iqbal, Shashir Reddy, Shruti Sheth, Siim Põder, Sijal Bhatnagar, Sindhu Raghuram Panyam, Sivan Eiger, Susan Zhang, Tianqi Liu, Trevor Yacovone, Tyler Liechty, Uday Kalra, Utku Evci, Vedant Misra, Vincent Roseberry, Vlad Feinberg, Vlad Kolesnikov, Woohyun Han, Woosuk Kwon, Xi Chen, Yinlam Chow, Yuvein Zhu, Zichuan Wei, Zoltan Egyed, Victor Cotruta, Minh Giang, Phoebe Kirk, Anand Rao, Kat Black, Nabila Babar, Jessica Lo, Erica Moreira, Luiz Gustavo Martins, Omar Sanseviero, Lucas Gonzalez, Zach Gleicher, Tris Warkentin, Vahab Mirrokni, Evan Senter, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, Yossi Matias, D. Sculley, Slav Petrov, Noah Fiedel, Noam Shazeer, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Jean-Baptiste Alayrac, Rohan Anil, Dmitry, Lepikhin, Sebastian Borgeaud, Olivier Bachem, Armand Joulin, Alek Andreev, Cassidy Hardin, Robert Dadashi, and Léonard Hussenot. Gemma 3 technical report, 2025. URL `https://arxiv.org/abs/2503.19786`.

[13] Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13084–13094, 2024.

[14] Haozhan Shen, Kangjia Zhao, Tiancheng Zhao, Ruochen Xu, Zilun Zhang, Mingwei Zhu, and Jianwei Yin. Zoomeye: Enhancing multimodal llms with human-like zooming capabilities through tree-based image exploration. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 6613–6629, 2025.

[15] Aleksandar Shtedritski, Christian Rupprecht, and Andrea Vedaldi. What does clip know about a red circle? visual prompt engineering for vlms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11987–11997, 2023.

[16] Stephanie Fu, Tyler Bonnen, Devin Guillory, and Trevor Darrell. Hidden in plain sight: Vlms overlook their visual representations. *arXiv preprint arXiv:2506.08008*, 2025.

[17] Benlin Liu, Amita Kamath, Madeleine Grunde-McLaughlin, Winson Han, and Ranjay Krishna. Visual representations inside the language model. *arXiv preprint arXiv:2510.04819*, 2025.

[18] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[19] Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, and Tengyu Ma. Larger language models do in-context learning differently, 2023. URL `https://arxiv.org/abs/2303.03846`.

[20] Martin Pawelczyk, Seth Neel, and Himabindu Lakkaraju. In-context unlearning: Language models as few-shot unlearners. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024.

[21] Jane Pan, Tianyu Gao, Howard Chen, and Danqi Chen. What in-context learning "learns" in-context: Disentangling task recognition and task learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8298–8319, 2023.

[22] Jannik Kossen, Yarin Gal, and Tom Rainforth. In-context learning learns label relationships but is not conventional learning. *arXiv preprint arXiv:2307.12375*, 2023.

[23] Andrew K Lampinen, Arslan Chaudhry, Stephanie CY Chan, Cody Wild, Diane Wan, Alex Ku, Jörg Bornschein, Razvan Pascanu, Murray Shanahan, and James L McClelland. On the generalization of language models from in-context learning and finetuning: a controlled study. *arXiv preprint arXiv:2505.00661*, 2025.

[24] Yongshuo Zong, Ondrej Bohdal, and Timothy Hospedales. Vl-icl bench: The devil in the details of multimodal in-context learning. *arXiv preprint arXiv:2403.13164*, 2024.

[25] Yu Wang and Sharon Li. Why multimodal in-context learning lags behind? unveiling the inner mechanisms and bottlenecks, 2026. URL `https://arxiv.org/abs/2604.13403`.

[26] Shuo Chen, Jianzhe Liu, Zhen Han, Yan Xia, Daniel Cremers, Philip Torr, Volker Tresp, and Jindong Gu. True multimodal in-context learning needs attention to the visual context. *arXiv preprint arXiv:2507.15807*, 2025.

[27] Folco Bertini Baldassini, Mustafa Shukor, Matthieu Cord, Laure Soulier, and Benjamin Piwowarski. What makes multimodal in-context learning work? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1539–1550, 2024.

[28] Chengyue Huang, Yuchen Zhu, Sichen Zhu, Jingyun Xiao, Moises Andrade, Shivang Chopra, and Zsolt Kira. Mimicking or reasoning: Rethinking multi-modal in-context learning in vision-language models. *arXiv preprint arXiv:2506.07936*, 2025.

[29] Nan Xu, Fei Wang, Sheng Zhang, Hoifung Poon, and Muhao Chen. From introspection to best practices: Principled analysis of demonstrations in multimodal in-context learning. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3299–3324, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.170. URL `https://aclanthology.org/2025.naacl-long.170/`.

[30] Kang-il Lee, Minbeom Kim, Seunghyun Yoon, Minsung Kim, Dongryeol Lee, Hyukhun Koh, and Kyomin Jung. Vlind-bench: Measuring language priors in large vision-language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4129–4144, 2025.

[31] Michal Golovanevsky, William Rudman, Michael A. Lepori, Amir Bar, Ritambhara Singh, and Carsten Eickhoff. Pixels versus priors: Controlling knowledge priors in vision-language models through visual counterfacts. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 24837–24852, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.1262. URL `https://aclanthology.org/2025.emnlp-main.1262/`.

[32] Amita Kamath, Jack Hessel, and Kai-Wei Chang. What's "up" with vision-language models? investigating their struggle with spatial reasoning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9161–9175, 2023.

[33] Ji Hyeok Jung, Eun Tae Kim, Seoyeon Kim, Joo Ho Lee, Bumsoo Kim, and Buru Chang. Is 'right'right? enhancing object orientation understanding in multimodal large language models through egocentric instruction tuning. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14257–14267. IEEE, 2025.

[34] Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit bayesian inference. *arXiv preprint arXiv:2111.02080*, 2021.

[35] Yufeng Zhang, Fengzhuo Zhang, Zhuoran Yang, and Zhaoran Wang. What and how does in-context learning learn? bayesian model averaging, parameterization, and generalization. *arXiv preprint arXiv:2305.19420*, 2023.

[36] Tomoya Wakayama and Taiji Suzuki. In-context learning is provably bayesian inference: A generalization theory for meta-learning, 2025. URL `https://arxiv.org/abs/2510.10981`.

[37] Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the 2022 conference on empirical methods in natural language processing*, pages 11048–11064, 2022.

[38] Qwen Team. Qwen3.5: Accelerating productivity with native multimodal agents, February 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

[39] 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 preprint arXiv:2508.18265*, 2025.

[40] Gemma Team. Gemma 4. 2026. URL `https://ai.google.dev/gemma/docs/core/model_card_4`.

[41] nostalgebraist. interpreting gpt: the logit lens. LessWrong / AI Alignment Forum, 2020. URL `https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens`.

[42] Benno Krojer, Shravan Nayak, Oscar Mañas, Vaibhav Adlakha, Desmond Elliott, Siva Reddy, and Marius Mosbach. Latentlens: Revealing highly interpretable visual tokens in llms, 2026. URL `https://arxiv.org/abs/2602.00462`.

[43] Vaidehi Bagaria, Bijo Sebastian, and Nirav Kumar Patel. Recursive belief vision language action models. *arXiv preprint arXiv:2602.20659*, 2026.

[44] Evžen Wybitul, Evan Ryan Gunter, Mikhail Seleznyov, and David Lindner. Vista dataset: Do vision-language models understand sequential tasks? *arXiv preprint arXiv:2411.13211*, 2024.

[45] Xuyang Guo, Zekai Huang, Zhenmei Shi, Zhao Song, and Jiahao Zhang. Your vision-language model can't even count to 20: Exposing the failures of vlms in compositional counting. *arXiv preprint arXiv:2510.04401*, 2025.

[46] Simone Alghisi, Gabriel Roccabruna, Massimo Rizzoli, Seyed Mahed Mousavi, and Giuseppe Riccardi. [del re] constructing vlms' reasoning in counting. *arXiv preprint arXiv:2510.19555*, 2025.

[47] Laleh Seyyed-Kalantari, Haoran Zhang, Matthew BA McDermott, Irene Y Chen, and Marzyeh Ghassemi. Underdiagnosis bias of artificial intelligence algorithms applied to chest radiographs in under-served patient populations. *Nature medicine*, 27(12):2176–2182, 2021.

[48] Chahat Raj, Bowen Wei, Aylin Caliskan, Antonios Anastasopoulos, and Ziwei Zhu. Vignette: Socially grounded bias evaluation for vision-language models. *arXiv preprint arXiv:2505.22897*, 2025.

[49] Charlie Snell, Dan Klein, and Ruiqi Zhong. Learning by distilling context. *arXiv preprint arXiv:2209.15189*, 2022.

[50] Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, et al. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*, 2026.

[51] Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-distillation enables continual learning, 2026. URL `https://arxiv.org/abs/2601.19897`.

[52] Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models, 2026. URL `https://arxiv.org/abs/2601.18734`.

[53] R Thomas McCoy, Shunyu Yao, Daniel Friedman, Mathew D Hardy, and Thomas L Griffiths. Embers of autoregression show how large language models are shaped by the problem they are trained to solve. *Proceedings of the National Academy of Sciences*, 121(41):e2322420121, 2024.

[54] Yasaman Razeghi, Robert L Logan IV, Matt Gardner, and Sameer Singh. Impact of pretraining term frequencies on few-shot numerical reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 840–854, 2022.

[55] Jiacheng Liu, Mayi Xu, Qiankun Pi, Wenli Li, Ming Zhong, Yuanyuan Zhu, Mengchi Liu, and Tieyun Qian. Format as a prior: Quantifying and analyzing bias in llms for heterogeneous data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 32060–32068, 2026.

[56] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022.

[57] Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Yu Qiao, et al. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? In *European Conference on Computer Vision*, pages 169–186. Springer, 2024.

[58] Haz Sameen Shahgir, Khondker Salman Sayeed, Abhik Bhattacharjee, Wasi Uddin Ahmad, Yue Dong, and Rifat Shahriyar. Illusionvqa: A challenging optical illusion dataset for vision language models. *arXiv preprint arXiv:2403.15952*, 2024.

[59] Francois Chollet, Mike Knoop, Gregory Kamradt, Bryan Landers, and Henry Pinkard. Arc-agi-2: A new challenge for frontier ai reasoning systems. *arXiv preprint arXiv:2505.11831*, 2025.

[60] Liang Chen, Weichu Xie, Yiyan Liang, Hongfeng He, Hans Zhao, Zhibo Yang, Zhiqi Huang, Haoning Wu, Haoyu Lu, Yiping Bao, et al. Babyvision: Visual reasoning beyond language. *arXiv preprint arXiv:2601.06521*, 2026.

[61] Chenwei Wu, Li Erran Li, Stefano Ermon, Patrick Haffner, Rong Ge, and Zaiwei Zhang. The role of linguistic priors in measuring compositional generalization of vision-language models. In *Proceedings on "I Can't Believe It's Not Better: Failure Modes in the Age of Foundation Models" at NeurIPS 2023 Workshops*, pages 118–126. PMLR, 2023.

[62] Kelsey Allen, Ishita Dasgupta, Eliza Kosoy, and Andrew K Lampinen. The in-context inductive biases of vision-language models differ across modalities. *arXiv preprint arXiv:2502.01530*, 2025.

[63] Patrick Amadeus Irawan, Ikhlasul Akmal Hanif, Muhammad Dehan Al Kautsar, Genta Indra Winata, Fajri Koto, and Alham Fikri Aji. Vision language models are confused tourists. *arXiv preprint arXiv:2511.17004*, 2025.

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

[65] Shuo Chen, Zhen Han, Bailan He, Jianzhe Liu, Mark Buckley, Yao Qin, Philip Torr, Volker Tresp, and Jindong Gu. Can multimodal large language models truly perform multimodal in-context learning? In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 6000–6010. IEEE, 2025.

[66] Yanshu Li. Advancing multimodal in-context learning in large vision-language models with task-aware demonstrations. *arXiv preprint arXiv:2503.04839*, 2025.

[67] 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 *European conference on computer vision*, pages 740–755. Springer, 2014.

[68] Gemini Team Google. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. URL `https://arxiv.org/abs/2312.11805`.

[69] Google DeepMind. Gemini 3.1 pro - model card, February 2026. URL `https://deepmind.google/models/model-cards/gemini-3-1-pro/`. Published 19 February 2026; accessed 2026-04-20.

[70] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pages 611–626, 2023.

[71] Mário Antunes, Tyler Estro, Pranav Bhandari, Anshul Gandhi, Geoff Kuenning, Yifei Liu, Carl Waldspurger, Avani Wildani, and Erez Zadok. Kneeliverse: A universal knee-detection library for performance curves. *SoftwareX*, 30:102161, 2025. ISSN 2352-7110. doi: https://doi.org/10.1016/j.softx.2025.102161. URL `https://www.sciencedirect.com/science/article/pii/S2352711025001281`.

A Appendix

Appendix Contents

A.1 Related Work 16
A.1.1 In-Context Learning in Language Models 16
A.1.2 Visual Failures in Vision-Language Models: Capability or Prior? 16
A.1.3 Multimodal In-Context Learning 16
A.2 Further Dataset and Evaluation Settings 17
A.2.1 Dataset 17
A.2.2 Evaluation and Models 23
A.3 Further Evaluations 25
A.3.1 ICL Effect Across Model Scales 25
A.3.2 Task Performance Breakdown in Detail 25
A.3.3 Expanded Benchmark Evaluation 28
A.3.4 Masking and Length-Matched Control Ablations 28
A.3.5 Explicit Language Prior: Can ICL Overcome? 30
A.3.6 Does Reasoning Help? 30
A.3.7 Probability of Correct Answer Labels 30

A.4 Mechanistic Experiments . . . 31
A.4.1 Results Across Model Families . . . 31
A.4.2 Additional Setting Details . . . 32
A.4.3 Prior-Cue Masking Example . . . 34
A.5 License and Ethics Statement . . . 35

A.1 Related Work

A.1.1 In-Context Learning in Language Models

Large language models are strongly shaped by their training distribution. They are sensitive to pretraining frequencies, token statistics, and output-format priors, even when a prompt asks for a different behavior [53–55]. In-context learning offers one way to change this behavior at inference time. Since the first few-shot results in large language models [18, 56], several studies have described ICL as a form of implicit task inference, where demonstrations provide evidence about the task and shift the model toward predictors that explain the examples [34, 37, 36]. This shift can be strong enough to override default behavior. Language models can learn counterintuitive label mappings from context [19], suppress unwanted behavior [20], and infer task rules from demonstrations [21–23].

Because ICL leaves model weights fixed and changes only the context, it serves as a probe of what a model already knows: if demonstrations shift behavior, the underlying capability was present but not selected by default. We adopt this diagnostic framing for vision-language models, where a shift from a prior-favored answer to a visually grounded answer would indicate that the visual capability was already available.

A.1.2 Visual Failures in Vision-Language Models: Capability or Prior?

Vision-language models still fail on tasks that require precise visual understanding, including chart reasoning [57], optical illusions [58], inductive visual reasoning [59], and basic visual perception [60]. Most existing remedies treat these failures as missing visual capability and respond by editing attention or internal representations [5–8], by altering the training recipe or visual encoder [9–12], or by preparing the input through cropping, zooming, or visual prompting [13–15]. These approaches differ in mechanism but share an assumption: the model cannot yet use the relevant visual evidence.

A separate line of work documents that VLMs often rely on training priors instead of the image. Some models answer visual questions from linguistic associations even when the image gives contrary evidence [61, 4], and predictions are highly sensitive to textual cues that pull the answer away from visual evidence [1, 30]. The bias extends beyond language: VLMs have clear preferences on which visual information to incorporate [62] and exploit shortcut compositions when solving visual tasks [2, 63, 31]. Recent analyses further show that useful visual information can be present in the model yet not used [16, 17].

Our work is motivated by a question raised by Luo et al. [1], Irawan et al. [63], Golovanevsky et al. [31]: can VLMs overcome training priors when supplied with in-context demonstrations that anchor their predictions in visual evidence? While Vo et al. [2] report that demonstrations yield little benefit, we argue that these null results stem from inadequate experimental controls. In particular, the insufficient number of demonstrations and limited diversity of answer labels hinder a model's ability to generalize.

A.1.3 Multimodal In-Context Learning

Models that interleave text and images make multimodal in-context learning possible [64]. Whether current VLMs perform genuine visual ICL remains contested. Several studies report that multimodal ICL is driven mainly by text, prompt format, or context cues rather than by demonstrated visual evidence [27, 65, 8], and that apparent gains can come from shallow strategies such as copying, matching, or majority voting [27, 28]. Multimodal ICL also lags text-only ICL on average [25] and is brittle under changes in perception, reasoning, or context length [24, 26, 66].

The most relevant work to our study demonstrates that VLMs possess some capacity to overcome semantic priors and follow unseen tasks from in-context demonstrations [29]. However, we argue that

the ICL settings in this study inadvertently invite shortcut learning rather than true visual grounding. Simple strategies, such as flipping answer labels or adopting an "I don't know" response format, constitute low-effort shortcuts that models can acquire directly from demonstrations without any reliance on visual evidence. Beyond this, the study's scope is largely limited to demonstration retrieval in general VQA benchmarks, with minimal control over task-specific factors central to our investigation.

A.2 Further Dataset and Evaluation Settings

A.2.1 Dataset

Task Definitions and Setup. This appendix expands the task and setting definitions from Section 3.3. We use the main-paper terminology throughout: each example has a *grounded answer* determined by the image and, when applicable, a *prior answer* favored when the model ignores that evidence. In the *complex* setting, *real* examples are prior-consistent because these answers agree, while *counterfactual* examples are prior-conflicting because they disagree. The five closed-label tasks are counting, color identification, shape identification, object direction, and spatial relation. The *simple* setting tests the visual operation on abstract stimuli, the *no-prior* setting replaces target names with placeholders so demonstrations specify the task rule, and the *complex* setting tests training prior override. Together, the tasks cover two shortcut regimes relevant to training prior dominance: counting, color identification, and shape identification test language or world-knowledge priors tied to object identity, while object direction and spatial relation test visual-configuration priors tied to common poses and layouts.

Counting. Following Vo et al. [2], this task asks for the number of visible animal legs in the *complex* setting. Real examples match common species leg counts, while counterfactual examples change the visible count. The *simple* and *no-prior* settings use synthetic circle-counting images; *no-prior* replaces the named target with the placeholder "target shapes." Answer labels are integers in $[1, 8]$ for the *simple* and *no-prior* settings and $[1, 5]$ for the *complex* setting.

Color Identification. Following Luo et al. [1], this task asks for the displayed color of the target object. The *complex* setting uses animals whose visible color either matches the species prior or conflicts with it. The *simple* and *no-prior* settings use synthetic colored shapes; *no-prior* fixes the target as the single circle and asks for the color of the "target shape." All three settings use {green, orange, brown, black, pink, red}, derived from the canonical colors of the six target animals in the *complex* setting.

Shape Identification. Following Luo et al. [1], this task asks for the displayed shape of the target object. The *complex* setting uses traffic signs whose rendered shape either matches the sign prior or conflicts with it. The *simple* and *no-prior* settings use synthetic 2D shapes; *no-prior* fixes the hidden target as the red object and asks for the shape of the "target object." Answer labels are {circle, triangle, rectangle, pentagon, octagon, diamond}.

Object Direction. Following Jung et al. [33], this task asks for the camera-relative orientation of a target object. The *complex* setting stresses visual-configuration priors by adding uncommon up and down orientations to the front, back, left, and right directions covered by the source benchmark. The *simple* and *no-prior* settings use SVG objects with red and green markers indicating the front-facing direction; *no-prior* uses more abstract non-arrow targets. Answer labels are {front, back, left, right, up, down}.

Spatial Relation. Following Kamath et al. [32], this task asks for the relation of an ordered target object to a reference object. The *complex* setting preserves the object pair while changing whether the depicted relation agrees with a frequent spatial layout. The *simple* and *no-prior* settings use synthetic colored shapes; *no-prior* fixes object A as a red circle and object B as a blue triangle, then queries the relation using only the placeholders object A and object B. All settings use the same answer labels: {left, right, front, behind, top, bottom}.

Prompts for Tasks

Counting

[Complex] How many legs does this animal have? Answer with a number in curly brackets. e.g., {4}.

[Simple] How many circles are there? Answer with a number in curly brackets. e.g., {6}.

[No-prior] How many target shapes are there? Answer with a number in curly brackets. e.g., {6}.

Color Identification

[Complex] What color is the animal in this image? Choose from: {green, orange, brown, black, pink, red}. Answer with the color in curly brackets, e.g., {red}.

[Simple] What color is the ⟨shape⟩? Choose from: {green, orange, brown, black, pink, red}. Answer with the color in curly brackets, e.g., {red}.

[No-prior] What color is the target shape? Choose from: {green, orange, brown, black, pink, red}. Answer with the color in curly brackets, e.g., {red}.

Shape Identification

[Complex] What shape is the traffic sign in this image? Answer with one of: {circle, triangle, rectangle, pentagon, octagon, diamond}.

[Simple] What shape is the ⟨color⟩ object? Choose from: {circle, triangle, rectangle, pentagon, octagon, diamond}. Answer with the shape name in curly brackets, e.g., {circle}.

[No-prior] What shape is the target object? Choose from: {circle, triangle, rectangle, pentagon, octagon, diamond}. Answer with the shape name in curly brackets, e.g., {circle}.

Object Direction

[Complex] From the perspective of the camera, which orientation is the object in the photo facing? Choose from: {front}, {back}, {left}, {right}, {up}, {down}. Answer with the direction word in curly brackets, e.g., {front}.

[Simple] / [No-prior] From the perspective of the camera, which orientation is the target object in the photo facing? Choose from: {front}, {back}, {left}, {right}, {up}, {down}. Answer with the direction word in curly brackets, e.g., {front}.

Spatial Relation

[Complex] What is the spatial relation of the ⟨objA⟩ relative to the ⟨objB⟩? Choose from the following options: {left}, {right}, {front}, {behind}, {top}, {bottom}. Answer with only the relation in curly brackets, e.g., {left}.

[Simple] What is the spatial relation of the ⟨colorA shapeA⟩ relative to the ⟨colorB shapeB⟩? Choose from the following options: {left}, {right}, {front}, {behind}, {top}, {bottom}. Answer with the relation in curly brackets, e.g., {left}.

[No-prior] What is the spatial relation of object A relative to object B? Choose from the following options: {left}, {right}, {front}, {behind}, {top}, {bottom}. Answer with the relation in curly brackets, e.g., {left}.

Figure 8: Prompts used for querying models across tasks and settings.

Dataset Statistics. Table 2 reports sources, labels, and counts for all tasks, with real/counterfactual statistics for each complex task. It also lists the counterfactual counting subset and the 450 high-count shape images used in Section 4.3.

Training Priors for Object Direction and Spatial Relation. These tasks probe preferences for common poses and layouts, with the preferred answer depending on the object or object pair. The selected EgoOrientBench entries [33] contain only front, back, left, and right labels; up and down

Table 2: Dataset statistics. (a) Original benchmark sources, labels, and counts, including real and counterfactual (CF) subsets. The real/CF split applies only to complex examples. Citations identify image sources or, for color and shape, task designs. Procedural images use Pillow or SVG; generated, regenerated, and edited images use Gemini. Regeneration reproduces a source image; HV denotes human validation. (b) Counting datasets used in Section 4.3; high-count shapes have 50 images per label.

(a) Sources, Labels, and Original Counts

Task	Setting	Source	Labels	Count	Real	CF	Edited	HV
Counting	Simple	Procedural (Pillow)	$\{1, \ldots, 8\}$	320				
Counting	No-prior	Procedural (Pillow)	$\{1, \ldots, 8\}$	320				
Counting	Complex	VLMs are Biased (regenerated) [2]	$\{1, \ldots, 5\}$	200	90	110	✓	✓
Color ID	Simple	Procedural (Pillow)	{G, O, Br, Bk, Pk, R}	240				
Color ID	No-prior	Procedural (Pillow)	{G, O, Br, Bk, Pk, R}	240				
Color ID	Complex	Generated + Edited; task design [1]	{G, O, Br, Bk, Pk, R}	240	120	120	✓	✓
Shape ID	Simple	Procedural (Pillow)	6 shapes	240				
Shape ID	No-prior	Procedural (Pillow)	6 shapes	240				
Shape ID	Complex	Generated + Edited; task design [1]	6 shapes	240	120	120	✓	✓
Direction	Simple	Procedural (SVG)	6 directions	240				
Direction	No-prior	Procedural (SVG)	6 directions	240				
Direction	Complex	EgoOrientBench (regenerated) [33]	6 directions	240	160	80	✓	✓
Spatial	Simple	Procedural (Pillow)	6 relations	240				
Spatial	No-prior	Procedural (Pillow)	6 relations	240				
Spatial	Complex	WhatsUp; COCO-/GQA-spatial [32, 67] (regenerated)	6 relations	240	160	80	✓	✓

(b) Further Counting Studies (Section 4.3)

Dataset	Source	Labels	Count	Study
Animal-leg counting	CF counting subset above	$\{1, \ldots, 5\}$	110	Figure 7(a)
High-count shapes	Synthetic 2D shapes	$\{8, \ldots, 16\}$	450	Figure 7(b)

each have zero examples. In the original complex direction split, edited up/down examples form the entire counterfactual subset: $80/240 = 33.3\%$ of the queries. For spatial relation, the selected entries from each source (WhatsUp VLM, COCO-spatial, and GQA-spatial) also contain zero top and zero bottom examples. WhatsUp's controlled images use unusual object combinations to reduce object-dependent biases [32], but the selected entries still leave these labels uncovered. These zero counts describe source coverage before counterfactual editing; the constructed benchmark covers all six labels for each task. In the original complex setting, both tasks contain 160 real and 80 counterfactual examples (Table 2).

We hypothesize that VLM training similarly favors common configurations, but model-specific training frequencies are unavailable and cannot be inferred from these source counts. To test this preference, we probe Qwen3-VL 32B directly. For object direction, we replace the image with object-identity text from its source caption, such as "A girl carrying an umbrella." Table 3 reports zero-shot accuracy with images and with this caption-only input. With images, the real-to-counterfactual accuracy gap is 47.4 percentage points for direction and 42.0 points for spatial relation. Caption-only direction accuracy remains above uniform six-label chance (16.7%) on real examples (42.1%), but falls below it on counterfactuals (2.7%). This asymmetry is consistent with a preference for common configurations without visual evidence.

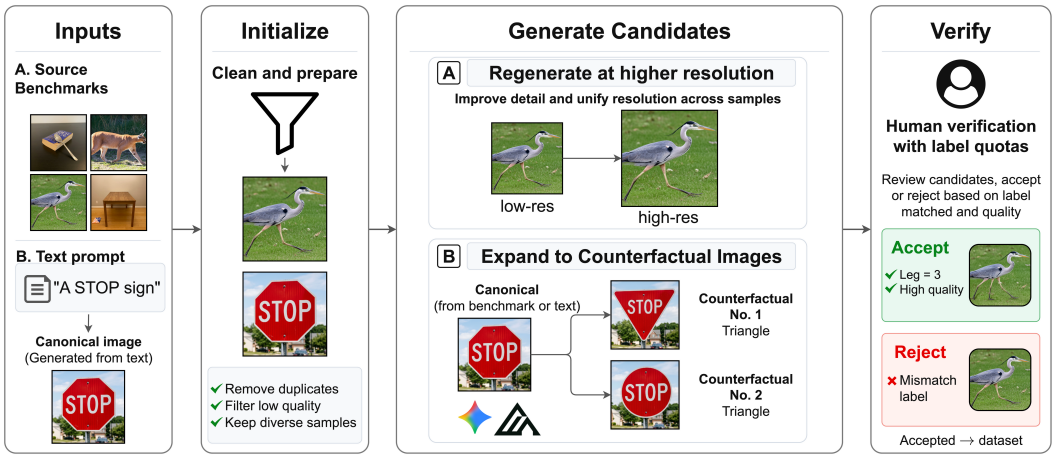


Figure 9: **Dataset construction.** For each task, images are rendered, generated, regenerated, or edited across the simple, no-prior, and complex settings, then paired with closed answer labels and validation controls for label-balanced ICL evaluation.

Both the *simple* and *no-prior* settings retain each task's six-label answer set. In the *simple* setting, Qwen3-VL 32B reaches zero-shot accuracies of 78.2% for direction and 80.8% for spatial relation (Table 7). These controls argue against a general inability to use the answer sets. The direction split remains a rare-orientation stress test because its real and counterfactual subsets use disjoint labels. Their accuracy gap therefore cannot separate orientation rarity from object-dependent priors.

Task	Input	Real	Counterfactual	Overall
Object Direction	Image	93.3	45.9	77.5
	Caption Only	42.1	2.7	29.0
Spatial Relation	Image	83.2	41.2	69.2

Table 3: **Configuration-prior diagnostics.** Zero-shot accuracy (%) for Qwen3-VL 32B on the original complex setting. Caption-only input replaces the direction image with object-identity text from its source caption. Overall accuracy weights the 160 real and 80 counterfactual examples in each task.

Data Construction and Validation. Our benchmark follows a common construction pipeline across tasks: initialize a candidate pool, expand the pool when prior-conflicting labels are needed, manually verify each candidate, and retain only verified samples in the cleaned benchmark. The pipeline varies by task and setting (Figure 9). Synthetic *simple* and *no-prior* settings are rendered with deterministic programs. The *complex* settings start from existing benchmark images or generated prior-consistent images, then use targeted image editing to produce prior-conflicting examples. After construction and validation, all retained images are downsampled to 512×512 pixels.

***Simple* and *no-prior* settings.** All *simple* and *no-prior* images are synthetically rendered with Pillow or SVG on a 512×512 canvas. No image generation model is used.

- **Counting.** Synthetic scenes contain same-color target circles and non-circle distractors. The *no-prior* prompt replaces "circles" with "target shapes."
- **Color identification.** Synthetic scenes contain a colored target shape and distractor shapes. The *no-prior* prompt fixes the target as the single circle and asks for the "target shape."
- **Shape identification.** Synthetic scenes contain a target shape and distractors in different colors. The *no-prior* prompt fixes the hidden target as the red object and asks for the "target object."

- **Object direction.** SVG target objects are rendered with distractors and front-direction markers. The *simple* setting uses arrow-like targets, and the *no-prior* setting uses abstract non-arrow targets.
- **Spatial relation.** Two colored shapes are placed in one of six spatial relations, with optional distractors. The *no-prior* prompt fixes object A as a red circle and object B as a blue triangle, then refers only to object A and object B.

Interpreting the *no-prior* zero-shot baseline. The *no-prior* setting is a task-rule inference control, not the main evidence for training prior override. Its prompts replace target names with placeholders, so demonstrations provide the hidden binding; this can make zero-shot queries underspecified when distractors are present. The primary training prior override evidence comes from the *complex* real/counterfactual split, where the queried target is explicit and the grounded answer either agrees with or conflicts with the prior answer.

***Complex* settings.** Images in the *complex* settings are either prior-consistent real examples or prior-conflicting counterfactual examples. Their base images come from existing benchmarks or are generated from scratch, and are then regenerated or counterfactually edited to produce the desired real/counterfactual split. We control for source-domain artifacts by generating or regenerating the prior-consistent images with Gemini rather than comparing raw originals against Gemini-edited counterfactuals. Thus, the real and counterfactual splits both come from the Gemini construction pipeline and share the same human validation, label balancing, and downsampling steps. All generated, regenerated, and edited images use Gemini [68, 69] at 1024×1024 resolution unless otherwise noted.

- **Counting.** Base images are drawn from the Animals topic of Vo et al. [2], filtered for counting questions with resolution $\geq$1024 px. Selected base images are regenerated at 1024×1024 to place prior-consistent examples in the same Gemini output domain as edited examples. For each target leg count in $\{1, 2, 3, 4, 5\}$, we edit base images so the animal has exactly the target number of legs. Edited images are manually verified with a per-label cap of 40, and unverified samples are removed.
- **Color identification.** We generate 10 canonical images per animal for six species with distinct canonical colors (frog/green, tiger/orange, capybara/brown, raven/black, flamingo/pink, scarlet ibis/red) at 512×512. These canonical Gemini generations form the real split. Each canonical image is then edited into the five non-original colors, producing counterfactual variants where the animal color conflicts with the species prior. Edited images are manually verified with a per-color cap of 40.
- **Shape identification.** We generate 10 canonical images per traffic sign for six sign types with distinct canonical shapes (stop/octagon, yield/triangle, railroad crossing ahead/circle, school zone crossing/pentagon, speed limit/rectangle, warning/diamond) at 512×512. These canonical Gemini generations form the real split. Each canonical image is then edited into the five non-original shapes. Edited images are manually verified with a per-shape cap of 40.
- **Object direction.** Base images are drawn from EgoOrientBench [33], filtered for `type=choose` entries with cardinal directions (front, back, left, right). Selected base images are regenerated at 1024×1024, and these regenerated images form the real split for the original directions. For each of the two missing directions (up, down), we edit base images to reorient the object accordingly (100 per direction). Edited images are manually verified with a per-direction cap of 40 across all six directions.
- **Spatial relation.** Base images and captions are drawn from WhatsUp VLM [32] (subsets A and B), COCO spatial [67], and GQA spatial. Object pairs and spatial relations are parsed from caption text. The prior-consistent base images are regenerated with Gemini and form the real split. For each base sample, a local language model generates up to two counterfactual captions that change only the spatial relation between the same objects, choosing relations that make the resulting configuration rare or unnatural. Each counterfactual caption is then used to edit the regenerated base image so the objects match the new spatial arrangement. Edited images are manually verified with a per-relation cap of 40 across all six relations.

Human Validation and Filtering. Two in-house annotators independently label every image in the *complex* settings, both blind to the intended label. Each annotator sees the image, task question, and answer options, then selects an answer in the same multiple-choice format used for model evaluation. We retain an image only if both annotators select the intended label and confirm that it meets the quality criteria. Any disagreement between annotators or with the intended label leads to rejection rather than adjudication.

Each annotator checks that the target object or relation is visible and unambiguous, distractors leave only one valid answer, and edits preserve object identity and scene realism. Text, markings, and object parts needed to answer the question must remain legible. We reject images with ambiguous, occluded, or cropped targets, visible editing boundaries, texture discontinuities, label-changing artifacts, prompt–image mismatches, or blur that prevents reliable judgment. The task-specific checks are:

- **Counting.** The animal's visible legs must support an unambiguous count. We reject images with unclear leg boundaries or no clear indication of individual legs.
- **Spatial Relation and Object Direction.** The depicted relation or orientation must correspond to a single answer option. We reject mixed directions, such as a person in three-quarter view looking at the camera.
- **Color and Shape Identification.** The main object's displayed color or shape must be unambiguous. We reject generated images that incorrectly mix attributes.

After filtering, we cap the retained images per label to improve label balance. Applying this protocol to the original dataset did not change the retained images.

Table 4 reports validation statistics for the original counterfactual portion of the *complex* setting. Across the five tasks, we accepted 510 of 936 candidates (54.5%) and rejected 426. These acceptance rates combine label agreement with the image-quality checks above; they do not measure inter-annotator agreement across all candidates.

Table 4: Human validation statistics for the original counterfactual portion of the complex setting. Acceptance requires both annotators to select the intended label independently and confirm that the image meets the quality criteria.

Task	Accepted	Rejected	Candidates	Acceptance (%)
Counting	110	192	302	36.4
Color ID	120	15	135	88.9
Shape ID	120	13	133	90.2
Object Direction	80	78	158	50.6
Spatial Relation	80	128	208	38.5
All Tasks	**510**	**426**	**936**	**54.5**

Counterfactual Generation Failures. Acceptance rates are lowest for counting (36.4%), spatial relation (38.5%), and object direction (50.6%), compared with color (88.9%) and shape identification (90.2%). These rates show the difficulty of generating unambiguous counterfactual images. They are consistent with generation models favoring familiar attributes and configurations, although the rejection counts also include ambiguity and image-quality failures.

Generation and editing prompts. Table 5 lists the exact prompts used for image generation and counterfactual editing in *complex* settings. All prompts are passed to Gemini [68, 69]. Placeholders in braces (e.g., `{animal_name}`) are filled at generation time. For regeneration from existing images, the source image is supplied as visual input and the instruction asks Gemini to reproduce the scene at the target resolution while preserving object identity, the answer-defining attribute or relation, background, lighting, and camera angle. This regeneration step leaves the real label unchanged; it only places prior-consistent examples in the same image-model output domain as counterfactual examples.

Table 5: Generation and editing prompts used for image construction in the *complex* setting.

Task	Stage	Prompt
Counting	Edit	`{new_total_legs} leg(s) total. Edit this image so the {animal_name} has exactly {new_total_legs} leg(s). {arrangement hint}. Every leg must match the size and texture of the original legs. The result must be photo-realistic with all {new_total_legs} leg(s) clearly visible and distinct. Preserve the same background, facial features, lighting, and camera angle.`
Direction	Edit	`Edit this image so the {object_name} faces {target_direction}ward. {front_part} should point toward the {top|bottom} of the frame, with the {object_name} {pose_desc}. The {object_name} must remain clearly visible and recognizable. The result must be photo-realistic. Preserve the same background, lighting, object size, and camera angle.`
Spatial	Caption	`You generate counterfactual captions by changing ONLY the spatial relation between the same two objects. Caption: {original_caption}. Choose a preposition from {allowed_preps} that makes the resulting caption feel the most unnatural. Produce {num_variants} candidate captions.`
Spatial	Edit	`Edit this image. From the viewer's perspective, the {obj1} is {relation} the {obj2}. Both objects must remain fully visible. Maintain photorealistic appearance. Preserve the same background, lighting, and object appearance.`
Color ID	Generate	`Generate a photorealistic image of a {animal_name}. The {animal_name} should be the central object in the image. Include realistic natural scenery around the animal. The animal must be clearly visible.`
Color ID	Edit	`Edit this image: change the color of the {animal_name} from {original_color} to {target_color}. The {animal_name} must remain clearly recognizable. Preserve the same background, lighting, and camera angle.`
Shape ID	Generate	`Generate a photorealistic image of a {sign_name} traffic sign. The sign is {shape}-shaped and should be the central object in the image. Include realistic outdoor scenery. The sign text and markings must be clearly visible.`
Shape ID	Edit	`Edit this image: change the shape of the {sign_name} sign from {original_shape} to {target_shape}. Keep the sign's text, colors, and markings identical. The {target_shape}-shaped sign must be clearly visible. Preserve the same background, lighting, and camera angle.`

A.2.2 Evaluation and Models

Prompt Setup for In-Context Learning. Each demonstration is a triple of an image I, a question Q, and an answer A. The full prompt begins with a system prefix, followed by N demonstrations $(I_1, Q_1, A_1), \ldots, (I_N, Q_N, A_N)$ presented as interleaved image–answer pairs, and ends with the query pair (I, Q) whose answer the model must predict.

Demonstration sampling. We use $N \in \{0, 1, 2, 4, 8, 16, 32\}$ demonstrations, where $N{=}0$ is the zero-shot baseline. Demonstrations are drawn uniformly across the unique answer labels in two

steps. Let L be the number of unique labels. First, we sample $\lfloor N/L \rfloor$ demonstrations from each label so that every label is equally represented. The remaining $N \bmod L$ slots are then filled by sampling uniformly from the leftover pool across all labels. This label-balanced sampling prevents improvements from being explained by answer-label frequency in the demonstrations. For *complex* settings, each context also balances real and counterfactual demonstrations whenever the shot budget and candidate pool allow.

Demonstrations and queries are drawn from the same task and setting unless otherwise specified (e.g., experiments across settings in Section 4.1). Query images are strictly disjoint from the demonstration candidate pool. Each nonzero-shot query is evaluated under 10 fixed demonstration contexts sampled with this procedure and a fixed random seed. Accuracy is averaged over query–context pairs, so each query contributes equally after averaging over its contexts. When an analysis pools tasks or settings, we report the micro-average over evaluated query entries rather than an unweighted mean of task-level accuracies. We use the same contexts across model families and prompt conditions whenever applicable. These repeated contexts reduce sensitivity to which specific demonstrations a query is paired with, since some demonstration sets may align unusually well or poorly with a given query.

For object direction, the real/counterfactual query split is also tied to orientation labels: real queries use EgoOrientBench's front, back, left, and right directions, while edited counterfactual queries introduce up and down. We therefore interpret this task as a rare-orientation stress test rather than a fully label-matched real/counterfactual comparison. The ICL contexts remain label-balanced across all six directions; at 16 shots, each context contains at least two examples per direction, including up and down, with real and counterfactual examples mixed.

Output Scoring and Answer Distributions. For exact-match scoring, we first extract the text inside the last pair of curly braces, `{...}`, in the model output. We then lowercase this text and strip leading and trailing whitespace. We apply the same lowercase-and-strip normalization to the ground-truth answer. The prediction is correct only if the two normalized strings match exactly. If the model output has no curly-braced span, the prediction is marked incorrect.

For answer-distribution analyses in Appendix A.3.7, we score candidate labels using normalized token-logit probabilities. For each query, let $\mathcal{Y}$ be the task-specific candidate label set. We compute a normalized probability for each candidate answer by applying a softmax over the candidate label logits. We then report the probability assigned to the visually correct answer $P(y_{\text{visual}})$ and the prior/default answer $P(y_{\text{prior}})$. This measurement is separate from raw answer frequency over sampled demonstration sets.

Model Architectures and Configurations. We evaluate five model families spanning a wide range of parameter counts, covering both recent vision–language models and earlier generations used as baselines. The specific checkpoints are:

- **Qwen3-VL** [10]: 2B, 4B, 8B, 32B, and 235B parameters.
- **Qwen2.5-VL** [9]: 3B and 32B parameters.
- **Qwen3.5** [38]: 4B and 27B parameters.
- **InternVL3.5** [39]: 4B and 38B parameters.
- **Gemma-4** [40]: E4B and 31B parameters.

For each family we use the instruction-tuned checkpoint. For models that support an extended "thinking" mode, we disable it by default so that all comparisons use single-pass, non-reasoning inference. Results with thinking enabled are reported separately in Section A.3.6. Main behavioral evaluations use vLLM [70]. Mechanistic evaluations use Hugging Face Transformers. For benchmark evaluations, we use the sampling parameters recommended by each model's official model card (summarized in Table 6). For mechanistic experiments, we use greedy decoding so that internal probes (attention, hidden states, label-logit probabilities) are evaluated against a deterministic generation rather than a stochastic sample.

All images are resized to 512×512 pixels before inference. We use each model's default system prompt and default chat-template formatting. There are no task-specific prompt differences across model families beyond backend-required serialization.

Table 6: **Sampling parameters for VLM benchmark evaluation.** Values follow each model card's recommended decoding settings and are used for the vLLM behavioral runs. Mechanistic experiments (Hugging Face Transformers) instead use greedy decoding and therefore do not use the temperature, top-p, top-k, repetition, or presence parameters listed here.

Mode	Model family	Max tokens	Temp.	Top-p	Top-k	Rep. penalty	Presence penalty
Standard non-thinking	Qwen3-VL, Qwen2.5-VL, Qwen3.5, InternVL3.5	512	0.7	0.8	20	1.0	vLLM: 1.5; HF: omitted
Standard non-thinking	Gemma-4	512	1.0	0.95	64	1.0	vLLM: 0.0; HF: omitted
Thinking mode	Qwen3-VL, Qwen2.5-VL, Qwen3.5, InternVL3.5	1024	1.0	0.95	20	N/A	N/A
Thinking mode	Gemma-4	1024	1.0	0.95	64	N/A	N/A

Compute Resources. All experiments are run on a single node with 4 NVIDIA H200 GPUs. Behavioral ICL evaluations use vLLM [70] for batched multi-image inference, and mechanistic analyses use Hugging Face Transformers to access per-layer attention and hidden states. The full empirical ICL sweep, covering all five model families, both representative model sizes per family, all five tasks, all three settings, and shot counts $N \in \{0, 1, 2, 4, 8, 16, 32\}$ with 10 demonstration contexts per task, setting, and shot count, consumed approximately 400 GPU-hours in aggregate. Mechanistic experiments and preliminary or discarded runs are not included in this estimate.

A.3 Further Evaluations

A.3.1 ICL Effect Across Model Scales

This section extends the scaling results presented in Section 4.1 with a more comprehensive analysis. We evaluate ICL performance across five model families: Qwen3-VL [10], Qwen2.5-VL [9], Qwen3.5 [38], InternVL3.5 [39], and Gemma-4 [40]. For each family, we select two representative sizes—4B and 30B—to examine how parameter count influences scaling behavior.

Figure 10 reports performance overall and across the *complex*, *simple*, and *no-prior* settings of the dataset. Two trends emerge. First, performance generally improves as the number of demonstrations grows, although Qwen2.5-VL 3B and Qwen3.5 show non-monotonic scaling. Second, larger models benefit more from additional demonstrations than their smaller counterparts.

Special Cases: Non-Monotonic Scaling On *complex* examples, Qwen2.5-VL 3B falls below zero-shot accuracy at 4 and 16 shots. Qwen3.5 4B improves at 4 shots, then declines on both real and counterfactual examples at 16; 27B retains most gains (Figure 10). One possible cause is long-context degradation: 16 or 32 multimodal demonstrations can push the context above 100K tokens.

A.3.2 Task Performance Breakdown in Detail

Table 7 reports task-level accuracies for each evaluated model family and setting, showing performance at 0-shot, 16-shot, and the difference between the two when both endpoints are available. For each family, we select two representative sizes—4B and 30B—to examine how parameter count influences scaling behavior.

Rows with both endpoints available show that the *complex* setting often gains from in-context demonstrations, although the size and direction of the gain vary by task and model. Gemma-4 is the strongest performer overall, achieving both the highest absolute accuracies and the largest gains from demonstrations on most tasks and settings. Within most model families, larger models generally benefit more from demonstrations, although task-level gains are not monotonic across all tasks and settings.

On the Real vs. Counterfactual Asymmetry. A natural concern about Figure 4(b) is that real-image accuracy on color and shape identification already exceeds 95.0% at zero shot for most $\geq$30B models, mechanically capping the absolute real-side Δ at only a few percentage points. The headline finding does not depend on these two tasks. Aggregate real accuracy on the *complex* setting is 86.9% at zero shot for Qwen3-VL 32B (Section 4.1), leaving roughly 13.0 percentage points of headroom that 16-shot prompting does not consume on the real subset. Counting, object direction, and spatial relation all have real-image zero-shot accuracy well below ceiling (Table 7), and the cf-vs-real asymmetry holds on these tasks as well.

Table 7: **Cross-family accuracies by task and setting.** Accuracies are percentages; standard deviations and Δ_{16-0} are in percentage points. For 16-shot rows, parentheses report standard deviations over sampled demonstration contexts. Best 0- and 16-shot entries per row are bold; for Δ_{16-0} the largest per-row improvement is bold. A dash indicates that a valid run was unavailable, and Δ_{16-0} is omitted unless both endpoints are available.

		InternVL3.5		**Qwen2.5-VL**		**Qwen3-VL**		**Qwen3.5**		**Gemma-4**	
Setting	**Shots**	**4B**	**38B**	**3B**	**32B**	**4B**	**32B**	**4B**	**27B**	**E4B**	**31B**
Counting											
No-prior	0	43.1	28.8	35.6	41.9	32.5	43.8	**58.8**	49.7	35.0	23.8
	16	49.8(±29.4)	76.5(±26.3)	19.9(±25.9)	70.8(±20.0)	35.6(±18.1)	71.9(±18.6)	16.8(±30.0)	**80.4**(±22.6)	28.6(±22.5)	72.4(±25.8)
	Δ_{16-0}	+6.6	+47.8	-15.7	+28.9	+3.1	+28.2	-41.9	+30.8	-6.4	**+48.5**
Simple	0	92.5	63.8	87.5	84.4	88.1	93.8	95.0	**95.6**	63.8	54.4
	16	85.2(±18.0)	89.1(±10.2)	26.7(±23.8)	82.9(±9.0)	82.1(±13.2)	**95.5**(±3.6)	23.3(±33.5)	94.6(±8.8)	65.6(±29.0)	83.8(±17.6)
	Δ_{16-0}	-7.2	+25.3	-60.8	-1.4	-6.1	+1.8	-71.7	-1.1	+1.8	**+29.4**
Complex	0	30.0	30.0	31.0	30.0	30.0	33.0	30.0	30.0	30.0	**42.0**
	16	30.7(±26.7)	30.6(±3.2)	26.1(±15.0)	31.0(±1.9)	36.0(±12.5)	53.7(±14.6)	25.9(±16.8)	46.6(±18.4)	29.8(±4.5)	**76.4**(±15.8)
	Δ_{16-0}	+0.7	+0.6	-4.9	+1.0	+6.0	+20.7	-4.1	+16.6	-0.2	**+34.4**
Color identification											
No-prior	0	**85.0**	50.8	65.0	52.5	50.8	52.5	61.7	60.0	55.8	54.2
	16	**84.8**(±17.0)	70.4(±19.9)	37.9(±34.3)	78.6(±18.3)	63.4(±17.2)	84.5(±17.2)	38.8(±33.5)	72.0(±30.9)	53.8(±17.6)	81.2(±20.6)
	Δ_{16-0}	-0.2	+19.6	-27.1	+26.1	+12.6	**+32.0**	-22.8	+12.0	-2.1	+27.1
Simple	0	**100.0**	**100.0**	98.3	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
	16	99.8(±0.4)	**100.0**(±0.0)	79.1(±25.9)	**100.0**(±0.0)	99.6(±0.7)	**100.0**(±0.0)	58.2(±45.0)	99.4(±1.8)	99.4(±1.3)	99.9(±0.3)
	Δ_{16-0}	-0.2	**+0.0**	-19.2	**+0.0**	-0.4	**+0.0**	-41.8	-0.6	-0.6	-0.1
Complex	0	83.5	83.7	47.2	79.1	**86.7**	80.8	56.4	78.8	77.9	84.3
	16	87.9(±4.6)	86.3(±8.5)	46.2(±41.9)	82.9(±9.0)	88.2(±6.5)	92.3(±6.5)	48.8(±40.8)	92.2(±8.9)	77.8(±20.3)	**93.9**(±5.5)
	Δ_{16-0}	+4.5	+2.7	-1.0	+3.8	+1.5	+11.5	-7.6	**+13.4**	-0.1	+9.6
Shape identification											
No-prior	0	63.3	60.0	59.2	59.2	55.8	61.7	60.0	57.5	58.8	**67.9**
	16	65.9(±16.1)	58.0(±19.6)	39.7(±23.7)	56.0(±14.0)	68.2(±21.1)	64.2(±19.6)	38.2(±43.7)	59.5(±26.3)	61.2(±19.4)	**81.1**(±23.7)
	Δ_{16-0}	+2.6	-2.0	-19.5	-3.2	+12.4	+2.6	-21.8	+2.0	+2.4	**+13.2**
Simple	0	99.2	98.3	98.3	98.3	**100.0**	**100.0**	**100.0**	**100.0**	90.8	**100.0**
	16	99.2(±0.9)	99.8(±0.8)	54.5(±39.1)	**100.0**(±0.0)	99.9(±0.3)	**100.0**(±0.0)	49.8(±47.2)	97.2(±7.0)	94.9(±8.2)	98.9(±2.6)
	Δ_{16-0}	+0.1	+1.4	-43.8	+1.7	-0.1	+0.0	-50.2	-2.8	**+4.1**	-1.1
Complex	0	83.0	93.7	48.7	83.2	**94.2**	87.5	22.7	82.3	89.5	**94.2**
	16	85.4(±5.7)	96.6(±3.4)	27.0(±18.3)	87.6(±9.6)	97.8(±3.7)	97.3(±2.0)	29.2(±42.0)	98.9(±2.2)	89.2(±13.7)	**99.9**(±0.3)
	Δ_{16-0}	+2.4	+2.9	-21.7	+4.4	+3.6	+9.8	+6.6	**+16.7**	-0.3	+5.8
Direction											
No-prior	0	**26.7**	15.0	16.7	25.0	20.0	22.5	19.7	14.2	17.2	15.8
	16	26.0(±26.5)	37.9(±23.5)	18.9(±9.0)	34.2(±28.2)	37.9(±34.1)	71.4(±28.4)	19.7(±28.1)	73.8(±32.9)	28.8(±16.3)	**92.0**(±13.3)
	Δ_{16-0}	-0.7	+22.9	+2.2	+9.2	+17.9	+48.9	+0.1	+59.7	+11.6	**+76.2**
Simple	0	68.3	50.0	43.3	70.0	77.5	78.2	72.5	15.0	55.0	**79.2**
	16	79.4(±20.3)	89.2(±14.3)	50.4(±28.1)	89.4(±8.6)	83.1(±6.3)	99.7(±1.1)	33.2(±38.9)	99.9(±0.3)	75.1(±17.0)	**100.0**(±0.0)
	Δ_{16-0}	+11.1	+39.2	+7.1	+19.4	+5.6	+21.5	-39.3	**+84.9**	+20.0	+20.8
Complex	0	56.7	70.0	34.2	52.5	58.3	**77.5**	64.2	28.3	38.3	**77.5**
	16	56.8(±19.7)	73.2(±11.9)	29.2(±30.2)	55.8(±8.9)	65.6(±11.1)	84.2(±8.5)	22.3(±37.9)	72.8(±23.3)	44.0(±18.4)	**87.8**(±8.5)
	Δ_{16-0}	+0.2	+3.2	-5.0	+3.2	+7.2	+6.7	-41.8	**+44.5**	+5.7	+10.3
Spatial relation											
No-prior	0	26.7	20.0	27.5	21.9	24.2	25.0	25.0	28.3	24.5	**33.3**
	16	23.6(±6.0)	29.2(±12.3)	11.4(±18.9)	22.2(±14.4)	23.8(±3.1)	38.0(±17.2)	17.1(±28.4)	40.8(±27.6)	21.4(±18.0)	**45.6**(±25.0)
	Δ_{16-0}	-3.1	+9.2	-16.1	+0.4	-0.3	**+13.0**	-7.9	+12.4	-3.1	+12.2
Simple	0	74.2	60.8	63.3	70.8	71.7	80.8	85.0	**90.8**	63.3	80.8
	16	71.9(±7.2)	76.3(±8.1)	20.2(±32.3)	79.1(±8.2)	75.2(±6.5)	92.8(±4.0)	35.2(±39.3)	94.2(±7.3)	61.5(±11.3)	**97.8**(±1.7)
	Δ_{16-0}	-2.2	+15.5	-43.2	+8.2	+3.6	+11.9	-49.8	+3.4	-1.8	**+17.0**
Complex	0	58.3	53.3	46.7	61.7	60.8	69.2	69.2	71.7	53.3	**74.2**
	16	51.1(±7.8)	59.6(±9.1)	36.2(±8.8)	60.4(±7.5)	60.9(±6.1)	72.2(±4.9)	64.4(±17.5)	**78.2**(±11.3)	44.1(±16.2)	73.4(±9.4)
	Δ_{16-0}	-7.2	+6.2	-10.4	-1.2	+0.1	+3.0	-4.8	**+6.5**	-9.2	-0.8

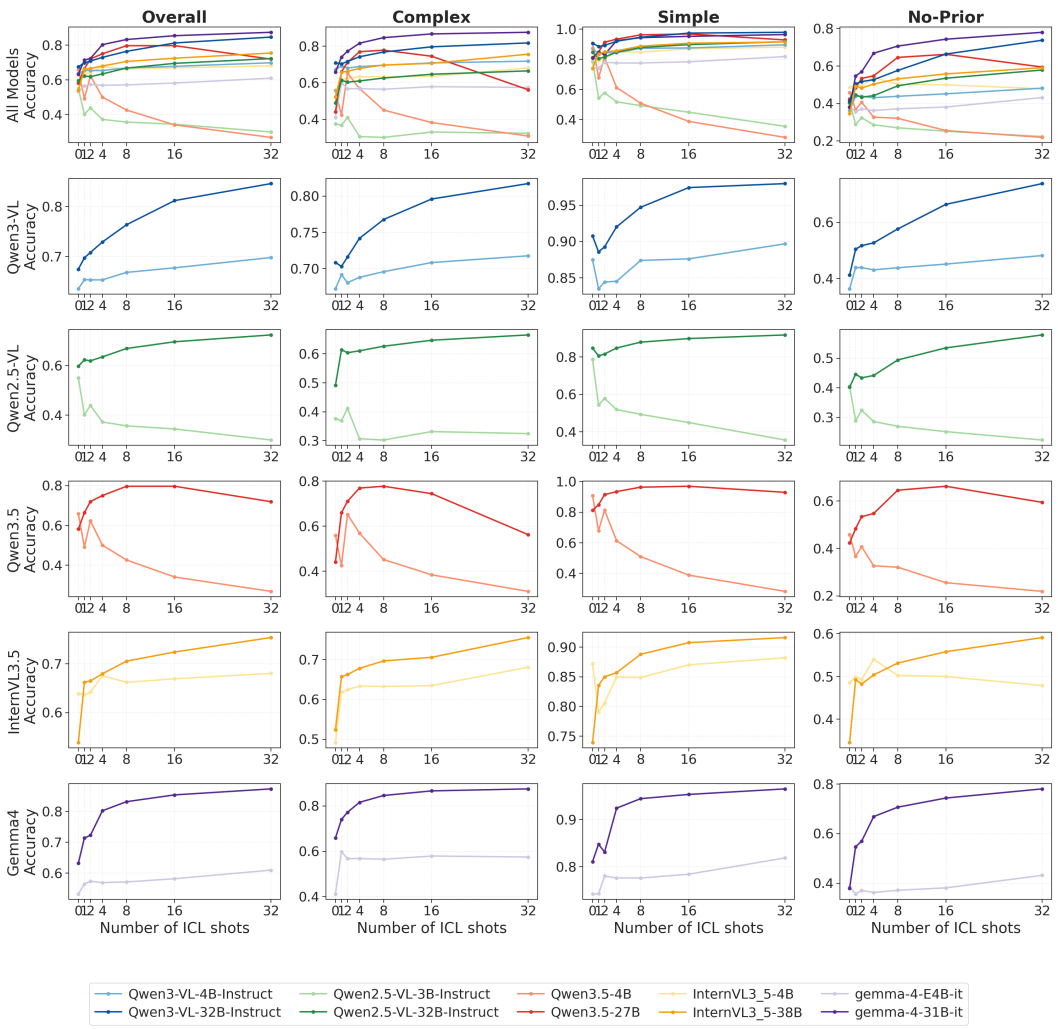


Figure 10: **ICL scaling across model families and settings.** Rows: model families (aggregated, then Qwen3-VL, Qwen2.5-VL, Qwen3.5, InternVL3.5, Gemma-4). Columns: settings. Qwen3-VL scales consistently; Qwen2.5-VL less so. Qwen3.5 and InternVL3.5 degrade beyond ∼8–16 shots. Gemma-4 achieves the strongest and most stable scaling.

Headroom-normalized error reduction, $\mathrm{ERR} = (\mathrm{Acc}_{16} - \mathrm{Acc}_0)/(1 - \mathrm{Acc}_0)$, measures the fraction of zero-shot error removed by demonstrations, using fractional accuracies. For Qwen3-VL 32B on *complex* color identification, real-image accuracy falls from 91.7% to 87.9%, while counterfactual accuracy rises from 70.0% to 96.7%. The corresponding ERR values are -45.8% and $+89.0\%$: demonstrations increase real-image error by 45.8% but reduce counterfactual error by 89.0%. Thus, the counterfactual advantage persists after accounting for the different amounts of available headroom.

Paired Errors of Gemma-4 and Qwen3-VL. To locate where the higher accuracy of Gemma-4 31B comes from, we compare it with Qwen3-VL 32B on the same *complex* queries (Table 8). On real examples, agreement stays stable at 89.6% at 0 shots and 88.9% at 16 shots. On counterfactual examples, 16-shot prompting raises shared correctness from 51.7% to 72.6% and lowers shared errors from 23.7% to 10.4%, so agreement increases by 7.7 points as the two models converge on the grounded answer. Gemma-4-only successes exceed Qwen3-VL-only successes in every row, and the gap is widest on counterfactual examples (15.4% versus 9.2% at 0 shots, 13.5% versus 3.4% at 16 shots). Counting accounts for much of this advantage: on counterfactual counting queries, Gemma-4

31B answers 21.3% correctly at 0 shots versus 4.2% for Qwen3-VL 32B, and 70.1% versus 35.8% at 16 shots. Overall counting accuracy is 42.0% versus 33.0% at 0 shots and 76.4% versus 53.7% at 16 shots, the largest per-task gap in Table 7. Demonstrations therefore move both models toward the grounded answer on examples where the training prior conflicts with the image, not only the weaker model.

Table 8: **Paired correctness of Gemma-4 31B and Qwen3-VL 32B.** Both models answer the same *complex* queries, split into real and counterfactual examples, at 0 and 16 shots. Agreement is the share of cases where both models are correct or both are incorrect. Percentages use the same aggregation as Figure 4(a), and entries are rounded independently.

Subset	Shots	Both correct (%)	Gemma-4 only (%)	Qwen3-VL only (%)	Both incorrect (%)	Agreement (%)	Δ Agreement (pts)
Real	0	82.4	5.9	4.5	7.2	89.6	—
	16	80.9	6.6	4.5	8.1	88.9	−0.7
Counterfactual	0	51.7	15.4	9.2	23.7	75.4	—
	16	72.6	13.5	3.4	10.4	83.1	+7.7

A.3.3 Expanded Benchmark Evaluation

We expand each task to approximately 400 complex examples by adding only counterfactual images. This changes the real-example share from 56.0% to approximately 33%; exact expanded per-task counts are unavailable. Table 9 reports a partial evaluation with Qwen3-VL 32B at 0 and 16 shots. Accuracy improves with 16 demonstrations on all five tasks, although these results cover only one model and two shot counts. Because overall task accuracy weights the real and counterfactual subsets by their counts, comparisons with the original benchmark also reflect the change in subset composition.

Version	Complex Examples	Real	Counterfactual	Real Share
Original	1,160 total	650	510	56.0%
Expanded	$\sim$400 per task	650	not reported	$\sim 33\%$

(a) Dataset statistics

Task	0-shot (%)	16-shot (%)	Δ (pp)
Counting	23.5	48.3 ± 12.1	+24.8
Object Direction	65.2	79.5 ± 12.8	+14.3
Spatial Relation	62.8	70.0 ± 7.2	+7.2
Color ID	72.9	89.1 ± 8.9	+16.2
Shape ID	83.2	96.4 ± 3.1	+13.2

(b) Partial expanded-benchmark results

Table 9: Expanded complex-setting dataset and partial results. Exact expanded counterfactual and total counts are unavailable. The partial evaluation covers Qwen3-VL 32B at two shot counts; 16-shot entries are mean accuracy $\pm$ standard deviation across ten fixed demonstration contexts, while zero-shot has no context variance. Accuracies are percentages; standard deviations and Δ are in percentage points (pp).

A.3.4 Masking and Length-Matched Control Ablations

Table 10 tests whether the gains come from added context, prompt format, or indiscriminate token removal. These control ablations use Qwen3-VL 32B on the *complex* setting with greedy decoding unless noted in the table. At 16 shots, masking grounding cues in the query image reduces accuracy by between 20.7 and 30.1 percentage points across counting, color identification, and shape identification. By contrast, prior-cue masking on real images changes zero-shot accuracy by at most 4.7 points. This contrast is consistent with demonstrations steering the model toward grounding cues rather than providing a generic masking benefit.

Task	0-shot	16-shot	Ground mask	Blank
Counting	32.1	54.4	24.3	28.4
Color ID	80.3	92.2	68.2	76.2
Shape ID	86.4	94.5	73.8	81.5

(a) Grounding-mask and blank-image controls at 16 shots.

Task	0-shot	Prior mask
Counting	68.2	63.5
Color ID	95.6	93.2
Shape ID	96.3	95.6

(b) Prior-cue masking on real images.

Task	0-shot	Full	Unrelated	Vision only	Blank
Counting	32.1	54.4	43.5	38.9	28.4

(c) Length-matched demonstration content at 16 shots.

Model	0-shot	Shuffled	Δ
2B	63.7	63.4	−0.3
235B	81.1	57.5	−23.6

(d) Qwen3-VL endpoints with 32 shuffled demonstrations.

Table 10: **Masking and length-matched control ablations.** Panels (a), (b), and (c) report accuracy (%) for Qwen3-VL 32B on the *complex* setting with greedy decoding. Ground mask removes grounding-cue tokens from the query image; Blank replaces each demonstration image with a same-size blank image. Unrelated uses demonstrations from another split of the same task, while Vision only removes question text from both the demonstrations and query. Panel (d) reports *simple*-setting endpoints from the shuffled-label sweep in Figure 11; Δ is the change from zero-shot accuracy in percentage points. Compute limits restrict panel (c) to counting.

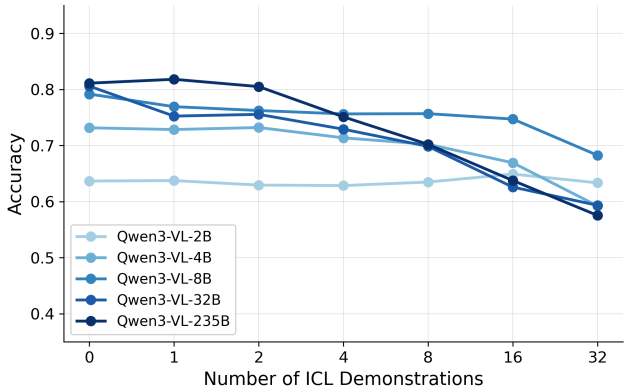


Figure 11: **Qwen3-VL unlearns under label shuffling.** In the *simple* setting, shuffled demonstration labels reduce performance below the zero-shot baseline, with larger drops for larger models.

The length-matched controls separate demonstration content from context length. Blank images reduce all three tasks below zero-shot accuracy. On counting, cross-split counting and vision-only demonstrations recover part of the full 16-shot gain, but both remain below the standard demonstrations.

Random label shuffling provides a further content control used in prior ICL work [37, 22, 19, 21]. It preserves the images and prompt format but breaks the mapping between each image and answer. Figure 11 shows that the resulting drop grows with model size. At 32 shuffled demonstrations, Qwen3-VL 235B falls from 81.1% to 57.5%, while 2B changes from 63.7% to 63.4%. These controls are inconsistent with an explanation based only on prompt length or format.

Cross-Task Demonstrations. Table 11 extends the counting controls to color and shape identification demonstrations. Animal-color demonstrations improve accuracy by 2.5 percentage points over zero shot, while the other cross-task conditions reduce accuracy. Same-task counting demonstrations transfer better across image domains, suggesting that added context alone does not explain the gains.

Table 11: **Cross-task demonstrations.** Accuracy on *complex* counting queries for Qwen3-VL 32B. Mixed color demonstrations are sampled uniformly across splits.

Demonstration Task	Images	Shots	Accuracy (%)
None	—	0	32.1
Counting	Complex (standard)	16	54.4
Counting	2D shapes	16	43.5
Color ID	Animals	16	34.6
Color ID	2D shapes	16	20.2
Color ID	Mixed splits	16	27.1
Shape ID	Traffic signs	16	25.2

A.3.5 Explicit Language Prior: Can ICL Overcome?

We next test whether ICL can override an *explicitly stated* language prior, following the setup of Luo et al. [1]. In this variant, a natural-language distractor statement appears immediately before the visual question on the *complex* setting of the three language-prior tasks: counting, color identification, and shape identification. The distractor templates state that `{animal} typically has {expected_bias} legs`, `{animal} is typically {expected_bias}`, or `{sign} typically has a {expected_bias} shape`. For example, a counting query becomes: "Dogs typically have 4 legs. How many legs does this animal have? Answer with a number in curly brackets, e.g., {4}." Demonstration prompts do not include distractor statements, and scoring follows the standard setup in Appendix A.2.2.

The counting case gives a compact stress test because the explicit sentence states the common animal-leg prior while the image may show a counterfactual leg count. Table 12 compares standard counting prompts with the explicit-prior variant. The distractor lowers zero-shot accuracy from 33.0% to 28.9%, showing that the textual cue makes the prior harder to ignore. With 16 demonstrations, accuracy still rises by 22.1% under the explicit prior, close to the 20.7% gain on the original prompts.

Counting	Original	Explicit Language Prior
0-shot	33.0	28.9
16-shot	53.7	51.0
Δ	$+20.7$	$+22.1$

Table 12: **Counting under an explicit language prior.** Accuracy (%) for standard counting prompts and prompts prepended with an explicit prior statement. The explicit prior lowers zero-shot accuracy, but the 0-to-16-shot gain remains comparable to the original prompt.

A.3.6 Does Reasoning Help?

We test whether reasoning improves in-context learning by enabling each model's thinking mode, which lets it generate reasoning tokens before producing an answer. We compare Qwen3-VL's Instruct and Thinking variants, and we toggle thinking mode on and off for Gemma-4. Reasoning runs use a 1024-token budget; non-reasoning runs use 512. Following Vo et al. [2], we expect textual reasoning to offer little benefit for visual tasks.

Figure 12 reports accuracy for both models at small and large scales. For Qwen3-VL, the Thinking variant matches the Instruct variant at both scales. For Gemma-4, the picture is split: E4B improves with thinking enabled, while 31B shows no consistent gain.

A.3.7 Probability of Correct Answer Labels

To gain deeper insight into model behavior, we examine the probability assigned to each ground-truth answer label. Specifically, we extract the output logit at the answer position and compute the probability of the correct token. For visualization, we focus on the *complex* settings of color identification and counting, which best illustrate how the model's output distribution shifts as in-context demonstrations are provided.

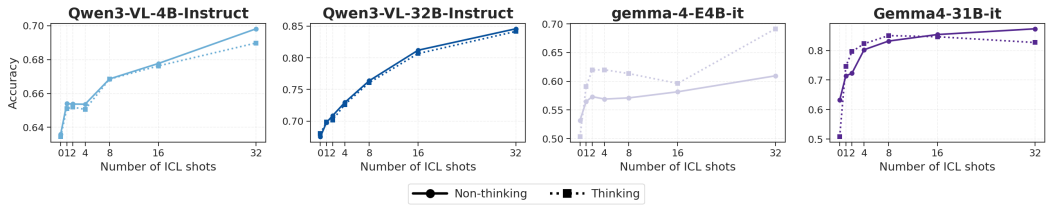


Figure 12: **Reasoning (thinking) vs. non-reasoning ICL accuracy.** Overall accuracy for Qwen3-VL-4B (Instruct vs. Reasoning), Qwen3-VL-32B (Instruct vs. Reasoning), Gemma-4-E4B-it, and Gemma-4-31B-it. Solid lines denote non-thinking mode; dotted lines denote thinking mode. Enabling thinking does not consistently improve ICL scaling.

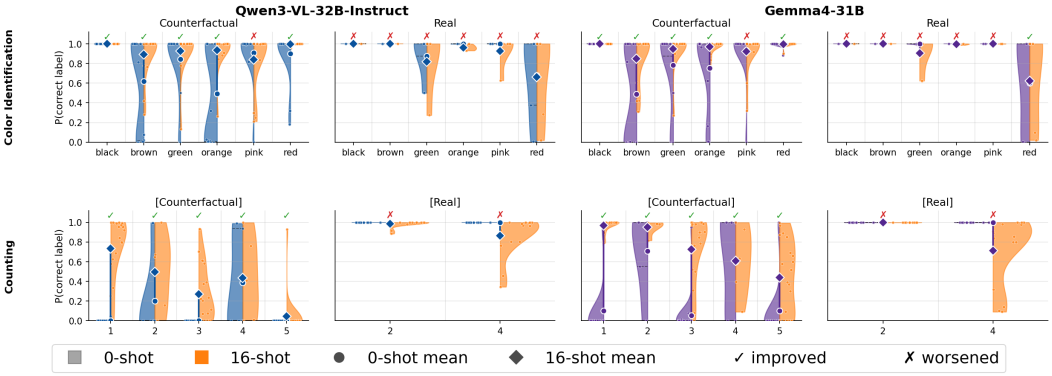


Figure 13: **Answer-label distribution shifts under ICL in the *complex* setting.** For Qwen3-VL-32B-Instruct and Gemma4-31B, we plot the probability assigned to the correct label at 0-shot and 16-shot for color identification (top row) and counting (bottom row), separated into counterfactual and real entries. Probability mass on counterfactual answers increases substantially with demonstrations, while probabilities on real entries remain largely intact.

Figure 13 shows how the answer probabilities evolve for Qwen3-VL 32B and Gemma-4 31B. Across both models, the probability of the correct answer on counterfactual images improves dramatically with additional demonstrations, while the probability on real images remains largely unchanged. Two patterns stand out:

- **Color identification (top row):** The correct-answer probability becomes sharply concentrated near 1.0, indicating that in-context demonstrations help the model overcome occasional errors induced by the training prior.
- **Counting:** At zero-shot, many correct answers that conflict with world knowledge encoded in the training prior receive a near-zero probability. The substantial improvement from 0-shot to 16-shot demonstrates the model's ability to override its strong training prior and learn the visual task from in-context demonstrations.

A.4 Mechanistic Experiments

This section gives full results and implementation details for the mechanistic analyses in Section 4.2. The attention analyses use the *complex* settings of counting and color identification, where each example has a visually correct answer and a competing prior answer. We report attention shifts for Qwen3-VL 4B and 32B and Gemma-4 E4B and 31B. Prior-cue masking covers all five tasks in the *complex* setting for Qwen3-VL 32B and Gemma-4 31B.

A.4.1 Results Across Model Families

Table 13(a) reports the change in attention per token from 0 to 16 shots. All four models reduce attention to prior cues and increase attention to grounding cues. For Gemma-4 E4B, these changes are

-2.62 and $+0.71$ in units of 10^{-3}; for Gemma-4 31B, they are -2.02 and $+1.23$. The comparison concerns the direction of each model's attention shift; per-token magnitudes also depend on the model's visual-token count.

Table 13(b) gives the full masking accuracies behind Figure 6(b), alongside the 16-shot results. Prior-cue masking exceeds both zero-shot accuracy and the matched random-mask control on every task for both models. Gemma-4 31B improves most on counting, from 40.9% to 58.8%, while 16-shot prompting reaches 73.9%. Its direction and spatial-relation gains under prior-cue masking are smaller: 2.6 and 1.7 percentage points, respectively. On spatial relation, prior-cue masking reaches 74.2%, compared with 73.4% at 16 shots and 72.5% at zero shot. Masking and demonstrations therefore improve accuracy in the same direction, but their relative gains vary by task.

Model	**Prior Cues**	**Grounding Cues**	**Other Tokens**
Qwen3-VL 4B	-3.41	$+0.59$	$+0.20$
Qwen3-VL 32B	-2.91	$+1.21$	$+0.18$
Gemma-4 E4B	-2.62	$+0.71$	$+0.21$
Gemma-4 31B	-2.02	$+1.23$	$+0.16$

(a) Change in attention per token from 0 to 16 shots ($\times 10^{-3}$).

Task	**0-shot**	**Random Mask**	**Prior Mask**	**16-shot**
Qwen3-VL 32B				
Counting	32.1%	28.2% (−3.9)	46.3% (+14.2)	**54.4%** (+22.3)
Color ID	80.3%	83.8% (+3.5)	**95.3%** (+15.0)	92.2% (+11.9)
Shape ID	86.4%	79.2% (−7.2)	92.6% (+6.2)	**94.5%** (+8.1)
Object Direction	73.2%	70.0% (−3.2)	79.3% (+6.1)	**82.9%** (+9.7)
Spatial Relation	64.8%	65.3% (+0.5)	68.2% (+3.4)	**71.0%** (+6.2)
Gemma-4 31B				
Counting	40.9%	36.5% (−4.4)	58.8% (+17.9)	**73.9%** (+33.0)
Color ID	82.5%	82.5% (0.0)	91.7% (+9.2)	**92.8%** (+10.3)
Shape ID	92.6%	92.1% (−0.5)	95.8% (+3.2)	**99.1%** (+6.5)
Object Direction	76.7%	70.0% (−6.7)	79.3% (+2.6)	**87.8%** (+11.1)
Spatial Relation	72.5%	70.6% (−1.9)	**74.2%** (+1.7)	73.4% (+0.9)

(b) Complex-setting accuracy (%) with greedy decoding; best per row in bold. Parentheses show percentage-point changes from each model's 0-shot baseline.

Table 13: **Mechanistic results across Qwen3-VL and Gemma-4.** (a) Changes in answer-token attention to prior cues, grounding cues, and other query-image tokens. Attention is normalized over query-image visual tokens and averaged per token within each group. (b) Zero-shot accuracy, zero-shot masking interventions, and 16-shot accuracy on all five tasks. The random mask matches the prior-cue mask's token budget. Object direction and spatial relation use object identity as a proxy prior cue.

A.4.2 Additional Setting Details

Cue-Token Metadata Fields. For each example, we instantiate cue descriptors and scoring labels from three metadata fields: `prior_cues`, `grounding_cues`, and `ground_truth`. The main analyses use two cue types: prior cues from `prior_cues` and grounding cues from `grounding_cues`. The `ground_truth` field defines the visually correct answer used for scoring. It is not, by itself, the prior-cue descriptor; it overlaps with the grounding descriptor only when the answer word also names the visible evidence, as in color identification.

For example, in a color-identification query with a red tiger, `prior_cues` contains `tiger`, `grounding_cues` contains the visible color `red`, and `ground_truth` is `red`. The prior cue is therefore the animal identity, while the grounding cue is the visible color evidence. In a counting query with a three-legged peacock, `prior_cues` contains `peacock`, `grounding_cues` contains

words such as `legs`, and `ground_truth` is 3. The prior cue is again the animal identity, while the grounding cue is the visible leg evidence needed to count.

For object direction and spatial relation, `prior_cues` contains the object named in the caption. These tasks have no single prior answer; object identity serves as a proxy cue that may correlate with the answer without specifying a direct shortcut. Masking therefore targets visual tokens that decode to this object identity.

Figure 14 shows representative examples of these fields in practice: the left panel is a counting example (one-legged swan, prior cue `swan`, grounding cue `legs`), and the right panel is a color-identification example (pink raven, prior cue `raven`, grounding cue `pink`), with prior-cue tokens marked by a red overlay on the query image.

Logit Lens and LatentLens. We use logit-lens [41] and LatentLens [42] probes to assign approximate semantic labels to visual-token representations. The probes are used only for analysis: they identify candidate prior-cue and grounding-cue visual tokens from the model's internal representations, but they do not provide supervision to the model during generation.

Adaptive thresholding with Kneedle. Logit lens and LatentLens do not assign a single label to each visual token; instead, they produce a ranked list of text tokens that are semantically compatible with the visual representation. Because score scales vary across layers, probes, models, and images, we avoid using a fixed similarity threshold. For each visual token, we retrieve the top-K matched text tokens, with $K = 50$, sort them by decreasing score, and apply the Kneedle algorithm [71] to the normalized score curve.

Kneedle identifies the rank at which the score curve changes from a steep drop to a flatter tail. We use this knee as an adaptive cutoff: all text tokens before the knee are preserved as the decoded semantic set for that visual token. If no stable knee is detected, we fall back to a fixed conservative cutoff and retain all the top-50 tokens.

We then mark a visual token as a prior cue if any preserved text token matches the prior-cue vocabulary, and as a grounding cue if any preserved text token matches the grounding-cue vocabulary. A single visual token can in principle satisfy both criteria; in such cases we assign the token to the prior-cue set. The prior-cue and grounding-cue sets are therefore disjoint by construction. This tie-breaking rule is conservative for the masking analysis in Sec. 4.2: when a token simultaneously decodes to prior-favoring and grounding semantics, the prior-cue mask removes it, ablating any grounding evidence it also carries. Improvements from prior-cue masking thus *lower-bound* the contribution of prior-cue semantics rather than overstate it.

Cue Sizes and Cutoff Sensitivity. On 200 *complex* counting images, 12.3% of visual tokens match prior cues and 8.4% match grounding cues in at least one probed early, middle, or late layer. These are across-layer coverage rates over Qwen3-VL's query-image visual tokens. With Kneedle disabled, retaining all $K = 50$ text matches, only 32 of approximately 350K visual tokens overlap before tie-breaking across the tested tasks and datasets. This checks overlap under the largest retained semantic sets, rather than the sensitivity of attention or masking effects to the cutoff.

Average Attention per Visual Token. For each query image, we identify cue-token sets using the metadata descriptors defined above. Let $\mathcal{V}_q$ denote all visual tokens for the query image, and let $\mathcal{C} \subseteq \mathcal{V}_q$ denote a cue-token set. We examine attention across early, middle, and late layers. For a model with L language-model layers, the early set contains the first three layers, the middle set contains the three layers centered at $\lfloor L/2 \rfloor$, and the late set contains the last three layers. For a selected layer ℓ and answer position a, we average attention from a to each query-image visual token across heads, giving $\bar{A}^{(\ell)}_{a,i}$ for token i. We compute the mean normalized attention per visual token for layer ℓ as

$$\alpha_\ell(\mathcal{C}_\ell) = \frac{1}{|\mathcal{C}_\ell|} \sum_{i \in \mathcal{C}_\ell} \frac{\bar{A}^{(\ell)}_{a,i}}{\sum_{j \in \mathcal{V}_q} \bar{A}^{(\ell)}_{a,j}}. \tag{2}$$

The denominator restricts the normalization to attention mass assigned to query-image visual tokens. The factor $1/|\mathcal{C}_\ell|$ makes cue sets of different sizes comparable. Thus $\alpha_\ell(\mathcal{C}_\ell)$ measures how much query-image attention an average token in cue set $\mathcal{C}_\ell$ receives at layer ℓ. We report the aggregate

mean over the selected early, middle, and late layers:

$$\alpha(\mathcal{C}) = \frac{1}{|\mathcal{L}_\mathcal{C}|} \sum_{\ell \in \mathcal{L}_\mathcal{C}} \alpha_\ell(\mathcal{C}_\ell), \tag{3}$$

where $\mathcal{L}_\mathcal{C}$ is the set of selected layers in which the corresponding cue set $\mathcal{C}_\ell$ is non-empty. Not all selected layers contain all cue types. When a prior or grounding cue set is empty in a layer, we omit that layer from the aggregate for that cue type rather than treating the missing cue as zero attention.

Qwen3-VL encodes images with a Vision Transformer using 16×16 patches, followed by a 2×2 spatial merge before the visual tokens are passed to the language model [10]. All images are resized to 512×512 pixels. This produces 32×32 raw patches and $16 \times 16 = 256$ merged visual tokens per image. Uniform attention over query-image visual tokens therefore corresponds to $1/256 \approx 0.0039$ attention per token.

A.4.3 Prior-Cue Masking Example

Token ablation blocks generated answer tokens from attending to selected query-image visual tokens during generation, but leaves these tokens unmasked during prefill. This distinction matters because logit-lens and LatentLens probing identify cue tokens from the unmasked visual-token representations; masking during prefill would alter the very representations used for cue identification. Generation-time masking thus holds the probed representation fixed while isolating the effect of selected cue tokens on answer generation.

The main intervention masks prior-cue tokens identified from the `prior_cues` field. We report accuracy for all five tasks using Qwen3-VL 32B and Gemma-4 31B with greedy decoding (Table 13(b)). Where a prior (default) answer is defined, we also measure how the candidate-normalized probability mass shifts between that answer and `ground_truth`. A matched random-token ablation accompanies this intervention as a control: for each query, we mask the same number of visual tokens as in the prior-cue mask for that query, drawn uniformly from visual tokens that are flagged as neither prior cues nor grounding cues, and average over multiple random seeds. Prior-cue masking outperforms this matched-budget random control on all five tasks in both models (Figure 6(b)). The object-identity proxy makes the evidence less direct for direction and relation, where Gemma's gains are also smaller than on counting, color, and shape.

Figure 14 shows two prior-cue masking examples from the *complex* setting. The left panel presents a one-legged swan from the counting task, where the prior cue is "swan" and the grounding cue is "leg." The answer-token probabilities show that both token masking and demonstrations shift the model away from the expected-bias answer (2) toward the ground-truth answer (1). The right panel presents a pink raven from the color-identification task, where the same pattern holds: both token masking and demonstrations shift the answer toward the ground-truth answer "pink."

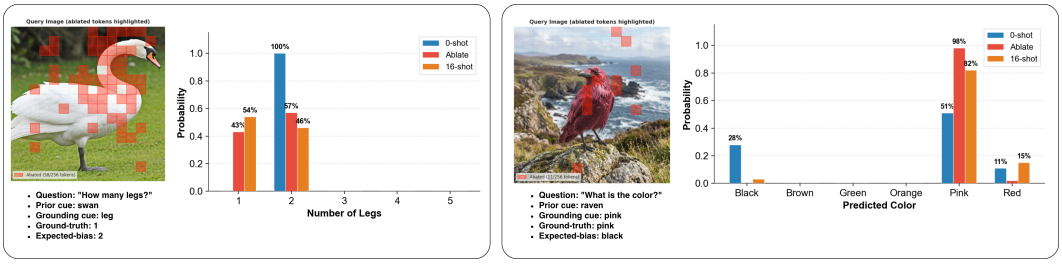


Figure 14: **Prior-cue masking examples.** Two queries from the *complex* setting with their identified prior-cue tokens (red overlay on the query image) and the candidate-normalized answer distributions at 0-shot, after prior-cue token ablation, and at 16-shot. Left: a one-legged swan, where the 0-shot model places nearly all mass on the prior-default answer "2"; ablating the prior-cue tokens shifts mass onto the ground-truth "1"; and 16-shot demonstrations produce a similar shift. Right: a pink raven, where 0-shot mass concentrates on the training prior default "black", and both prior-cue ablation and 16-shot ICL move probability mass onto the ground-truth "pink". Prior-cue ablation reproduces the qualitative effect of in-context demonstrations.

A.5 License and Ethics Statement

Our benchmark contains three categories of images: (1) procedurally rendered synthetic images, (2) real images regenerated with Gemini from base images in existing publicly available benchmarks, and (3) images generated from scratch or counterfactually edited with Gemini. Synthetic and generated images carry no privacy or consent concerns. The licenses and terms of the source benchmarks govern the base images used for regeneration; we cite each source dataset (Appendix A.2.1) and do not redistribute the original base images.

All generated and counterfactually edited images are disclosed as such in the release metadata. Each sample includes a source-type field indicating whether the image is procedural, regenerated, generated, or edited.

Our benchmark is designed to evaluate shortcut and prior behavior in VLMs. By construction, examples in the *complex* setting intentionally contain unusual or counter-stereotypical object attributes (e.g., a pink tiger, a one-legged dog). These examples do not represent real-world distributions and are not intended to reinforce stereotypical associations.

Human validation was an in-house dataset quality review by two authors. They independently selected answers while blind to the intended labels and applied the fixed quality criteria in Appendix A.2.1. We discarded images with label disagreements or quality failures; no crowd-sourced labeling or human-subject interaction was conducted.

Broader Impacts. On the positive side, our work provides a controlled diagnostic that separates training prior dominance, where the model has the relevant visual evidence but defaults to a learned expectation, from genuine capability absence. This distinction tells researchers and practitioners when a failure can be addressed at test time with demonstrations and when it instead calls for changes to data, training, or architecture, supporting more targeted and resource-efficient VLM development. This distinction is also fairness-relevant: a VLM that does not actually consult the image must substitute statistical priors for perception, and those priors can encode demographic and stereotypical biases, detrimental to downstream applications. Finally, ICL gains require long multimodal contexts, which increase inference compute and energy use (Section 5).

AI Use Statement. Generative AI tools assisted with data generation, coding the ICL pipeline, and editing the LaTeX source. The authors reviewed the resulting source and take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction (Section 1) state our main claims—that visual ICL substantially reduces training prior dominance on prior-conflicting examples and that it does not create capabilities that are absent—and these claims are supported by the empirical results in Section 4.1, the mechanistic analyses in Section 4.2, and the capability-limit analysis in Section 4.3.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 4.3 analyzes when ICL fails to compensate for weak underlying visual capabilities (e.g., high-cardinality counting), and the conclusion (Section 5) discusses the compute and context cost of long multimodal demonstrations along with the resulting need for demonstration selection or context distillation.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: Section 2 introduces a conceptual framework (training prior, task decomposition, ICL as Bayesian inference over prediction strategies) that motivates the empirical study, but the paper does not state or prove formal theorems.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Appendix A.2.1 documents the data construction pipeline, prompts, and human-validation criteria for every task and setting; Appendix A.2.2 specifies the prompt format, demonstration sampling, contexts, and shot counts; Appendix A.2.2 lists exact model checkpoints and inference backends; Appendix A.2.2 describes the exact-match scoring procedure.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We will release the benchmark (synthetic images, Gemini-regenerated, generated, and edited images, per-sample metadata with source-type fields, and task labels) together with the evaluation code. An anonymized version is included in the supplemental material at submission time.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Our evaluation is training-free; all evaluation details—shot counts $N \in \{0, 1, 2, 4, 8, 16, 32\}$, label-balanced demonstration sampling, the 10 fixed demonstration contexts shared across queries, image resolution, decoding parameters (Table 6), and per-family model checkpoints—are documented in Appendix A.2.2 and Appendix A.2.2.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: Each non-zero-shot accuracy is averaged over 10 fixed demonstration contexts, shared across queries and sampled with a fixed seed (Appendix A.2.2); error bars in the ICL-scaling figures are standard deviations across these contexts. Behavioral runs use stochastic decoding (Table 6) with one sampled output per query-context pair, so the error bars also include decoding noise and therefore upper-bound the variability due to the choice of demonstration set.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix A.2.2 reports the compute setup: a single node with 4 NVIDIA H200 GPUs, vLLM for behavioral evaluation and Hugging Face Transformers for mechanistic analyses, with the full empirical ICL sweep consuming approximately 400 GPU-hours.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics. The benchmark uses procedurally generated images and images generated or edited with a public image-generation API, including images regenerated from base images in existing public benchmarks; no human-subject data is collected and no personally identifying information is included (Appendix A.5).

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Appendix A.5 discusses both the positive impact—a diagnostic that distinguishes prior dominance (test-time correctable) from capability absence and informs whether test-time intervention or further training is appropriate—and the steps taken to avoid reinforcing stereotypical associations from the counter-stereotypical *complex* stimuli, which are explicitly flagged as non-representative of real-world distributions.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: We release an evaluation benchmark, not a generative model. The dataset contains synthetic, regenerated, and counterfactually edited images that pose no clear misuse risk, and each sample carries a source-type field (Appendix A.5) so downstream users can distinguish real, generated, and edited images.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All evaluated VLMs are cited in Appendix A.2.2 with their corresponding technical reports and licenses, and all source benchmarks are cited in Appendix A.2.1. License terms and re-distribution constraints are summarized in Appendix A.5; the source benchmarks' licenses and terms govern the base images used for regeneration, and we do not redistribute the original base images.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The new benchmark introduced in this paper, is documented in Section 3 and Appendix A.2.1, including task definitions, construction and validation pipelines, label sets, dataset statistics (Table 2), and the exact prompts used for image generation and counterfactual editing (Appendix A.2.1); a datasheet and per-sample metadata accompany the release.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: Two authors independently labeled images while blind to the intended labels and applied fixed quality criteria (Appendix A.2.1). We retained an image only when both selected the intended label and confirmed that it met these criteria. This was in-house dataset quality review; no crowd workers, external annotators, or human subjects participated.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The work does not involve human-subject data collection; all annotation was an in-house quality review of synthetic and image-model-generated stimuli, so no IRB review was required.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: We use Gemini [68, 69] as a non-standard core component for image generation, regeneration, and counterfactual editing in the *complex* settings of the benchmark; the exact prompts and per-task pipelines are documented in Appendix A.2.1 and Appendix A.2.1. The VLMs that are the subject of study (Appendix A.2.2) are evaluated through standard inference and are not used to develop the methodology itself.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.