CABLE: EVALUATING SELECTIVE LOW-LEVEL VISUAL EVIDENCE TRACKING IN MLLMS

Anonymous authors
Paper under double-blind review

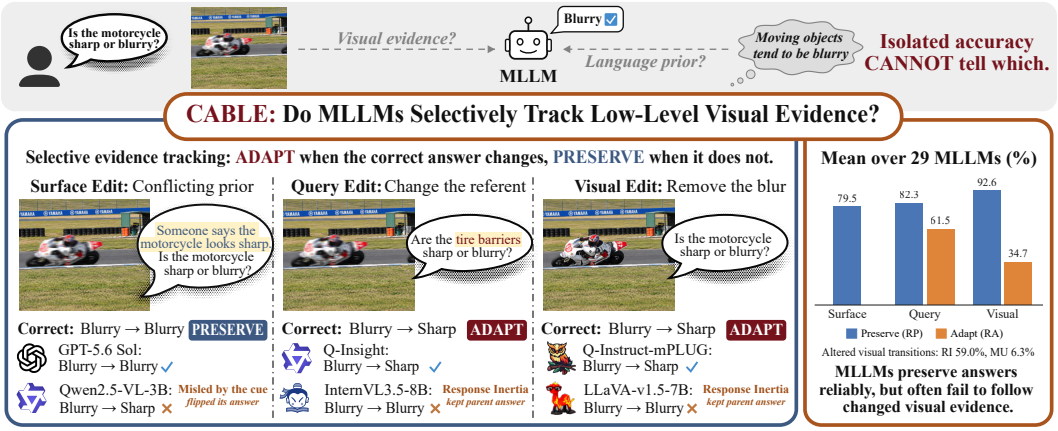


Figure 1: **Overview of CABLE.** CABLE tests whether MLLMs selectively track low-level visual evidence: responses should adapt when a controlled edit changes the correct answer and persist otherwise. MLLMs reliably preserve correct answers yet often fail to follow visual changes.

ABSTRACT

Multimodal large language models (MLLMs) are increasingly applied to image quality assessment, degradation perception, and other low-level visual tasks. Yet isolated image-question accuracy cannot distinguish responses grounded in query-relevant visual evidence from those supported by linguistic and high-level semantic cues. We formulate this problem as *selective evidence tracking*: a model should revise its response when relevant evidence changes the correct result and preserve it when the result remains unchanged. To evaluate this behavior, we introduce Controlled Assessment of Bundled Low-Level Evidence (CABLE), which organizes anchor instances and controlled variants into intervention-induced transitions. Conditioned on a correct parent response, CABLE measures response adaptation (RA) on altered transitions and response preservation (RP) on invariant transitions, while decomposing adaptation failures into response inertia (RI) and misdirected updates (MU). Across 29 general-purpose and low-level-vision-specialized MLLMs, models preserve correct judgments under invariant changes far more reliably than they revise them when visual evidence changes the required result. More importantly, response revision is substantially weaker for visual than query-induced changes, with adaptation failures dominated by response inertia rather than misdirected updates. Fine-grained analyses further reveal systematic effects of visual relevance, query grounding, and linguistic elicitation that isolated accuracy obscures. CABLE therefore complements pointwise accuracy with transition-level diagnostics of whether model responses remain responsive to the low-level visual evidence that should determine them.

1 INTRODUCTION

> *"A wise man, therefore, proportions his belief to the evidence."*
>
> — *David Hume, An Enquiry Concerning Human Understanding*

Recent work on multimodal large language models (MLLMs) has extended visual understanding from high-level semantics to low-level perception and degradation-aware reasoning, includ-

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

ing degradation perception and description (Wu et al., 2024a;b), perceptual image quality assessment (You et al., 2024; 2025b; Wu et al., 2024c; Li et al., 2025c), and robust content understanding under degraded inputs (Tang et al., 2026; Li et al., 2026a). Yet growing evidence shows that MLLM responses can be disproportionately influenced by linguistic priors and high-level semantic cues rather than visual evidence (Guan et al., 2024; Lin et al., 2024; Wang et al., 2025a). A similar tension arises in low-level vision, where image semantics can overshadow distortion cues and even basic degradations may be recognized unreliably (Chen et al., 2026; Li et al., 2025d). Because existing evaluations largely score image-query pairs in isolation, a correct response remains behaviorally ambiguous: it may follow the queried low-level evidence, or arise from linguistic and semantic cues that support the same answer. This raises a question that isolated correctness cannot answer: *does the model selectively track the visual evidence relevant to the query?*

Selective evidence tracking is not merely sensitivity to input changes, but selective dependence on the evidence relevant to the queried result. This dependence cannot be identified from isolated correctness, since the same correct response may be supported by the intended visual evidence or by other cues that happen to yield the same answer. Controlled interventions make this dependence behaviorally observable by specifying how a response should change: the model should adapt when an intervention changes the correct result and preserve its response when it does not. We therefore apply controlled edits to the image or query and test whether responses show the required adaptation or preservation across the induced transitions (Shah et al., 2019; Yuan et al., 2021; Chou et al., 2025).

Building on this principle, we introduce Controlled Assessment of Bundled Low-Level Evidence (CABLE), an evaluation protocol that organizes each original image-query instance, termed an *anchor*, together with controlled variants linked by local intervention-induced transitions. Rather than treating each variant as an independent test item, CABLE evaluates whether the model's response changes as required across each parent-child transition. A transition is *altered* when the intervention changes the correct result and *invariant* when it does not; importantly, the required response behavior is determined by the intervention's effect on the result, not by the edit type itself. To separate evidence tracking from initial task correctness, CABLE evaluates a transition only when the model responds correctly at its parent node. It then measures *response adaptation* (RA) on altered transitions and *response preservation* (RP) on invariant transitions. When adaptation fails, CABLE further distinguishes *response inertia* (RI), where the model retains the parent's semantic result even though the correct result has changed, from *misdirected updates* (MU), where the response changes but still fails to reach the new correct result. Together, these diagnostics operationalize selective evidence tracking as a transition-level test of selective responsiveness: whether a model revises an established judgment when the evidence requires it, while preserving that judgment when the correct result should remain unchanged.

Using CABLE, we evaluate 29 general-purpose and low-level-vision-specialized MLLMs and find that isolated correctness provides only a partial account of low-level visual behavior: correctness on anchor instances often fails to transfer to controlled variants. Separating altered from invariant transitions reveals a pronounced adaptation-preservation asymmetry, with models preserving correct judgments far more reliably than revising them when the correct result changes. More importantly, this adaptation deficit is substantially greater when the required revision is induced by changed visual evidence rather than by a changed query. The resulting visual adaptation failures are dominated by RI rather than MU, indicating that models more often retain a previously correct judgment than update toward a different but still incorrect result. Fine-grained analyses further reveal systematic effects of visual relevance, query grounding, linguistic elicitation, and low-level specialization. Together, these findings show that similar isolated accuracy can conceal markedly different evidence-tracking profiles, particularly in how strongly visual evidence controls response revision.

Our contributions are threefold. (1) We formulate *selective evidence tracking* as a behavioral evaluation problem for low-level vision and introduce CABLE, a controlled-intervention benchmark that tests whether MLLM responses adapt when relevant evidence changes the correct result and remain stable otherwise. (2) We develop parent-conditioned transition diagnostics that isolate evidence tracking from initial task correctness through RA, RP, and the RI/MU decomposition of adaptation failures. (3) We benchmark 29 general-purpose and low-level-vision-specialized MLLMs and reveal a pronounced query-visual adaptation gap: responses adapt far less reliably to changed visual evidence than to changed queries, with failures dominated by response inertia. We further analyze the effects of visual relevance, query grounding, linguistic elicitation, and low-level specialization.

2 CABLE: CONTROLLED EVIDENCE BUNDLES

2.1 SELECTIVE EVIDENCE TRACKING AND BUNDLE STRUCTURE

We formalize selective evidence tracking over controlled transitions between image-query instances. Our formulation separates two questions: what an intervention changes, and how the correct response should change as a consequence. Let f denote an MLLM that maps an image-question pair (I_i, T_i) to a free-form response $\hat{y}_i = f(I_i, T_i)$, where T_i includes the natural-language question and its answer interface. For benchmark construction, each instance also has a structured visual state V_i, unseen by the model, recording the low-level visual facts underlying I_i.

Natural-language questions couple the underlying visual query with its surface realization. The same visual evidence can be queried through different languages, response formats, or linguistic framings, while small textual edits may instead change what visual fact is being asked about. We therefore normalize each question into a structured query via:

$$Q_i = \pi(T_i) = \langle \mathcal{F}_i, \Phi_i, \rho_i \rangle, \tag{1}$$

where $\mathcal{F}_i$ specifies the queried referent-property facts, Φ_i defines how these facts are read or composed, and ρ_i accounts for query polarity. The corresponding query result is:

$$z_i = \rho_i(\Phi_i(V_i; \mathcal{F}_i)) \,. \tag{2}$$

We distinguish z_i from the surface reference answer y_i, which expresses the same result through the language and answer interface specified by T_i.

For a parent-child pair (u, v), this decomposition defines three atomic edit types:

$$\begin{aligned} \textit{Surface}: &\quad T_v \neq T_u,\ Q_v = Q_u,\ V_v = V_u, \\ \textit{Query}: &\quad Q_v \neq Q_u,\ V_v = V_u, \\ \textit{Visual}: &\quad T_v = T_u,\ Q_v = Q_u,\ V_v \neq V_u. \end{aligned} \tag{3}$$

A query edit may require a synchronized change to its textual realization, whereas a visual edit changes only the underlying visual state. Controlled variants are constructed by manipulating these structured components rather than treating the question as an unstructured string.

CABLE organizes each anchor and its variants into a bundle $\mathcal{B} = (\mathcal{N}, E)$, with nodes $n_i = (I_i, T_i, Q_i, V_i, z_i, y_i)$ and atomic edits as edges. The recursive construction described in the next subsection gives the relations between parents and children a tree structure.

An edit type specifies which structured component is changed, while the required response behavior depends on whether the query result changes. For a transition from parent u to child v, we define:

$$r_{uv} = \begin{cases} \text{invariant}, & z_u = z_v, \\ \text{altered}, & z_u \neq z_v. \end{cases} \tag{4}$$

This distinction operationalizes selective evidence tracking at the transition level. Altered transitions require response adaptation, while invariant transitions require response preservation.

2.2 BENCHMARK CONSTRUCTION

Anchor Construction. We construct anchors from DIV2K (Agustsson & Timofte, 2017), Flickr2K (Lim et al., 2017), COCO (Lin et al., 2014), CC12M (Changpinyo et al., 2021), LAION (Schuhmann et al., 2021), TextVQA (Singh et al., 2019), FoundIR (Li et al., 2025b), and RealBlur (Rim et al., 2020), covering natural, web, text-rich, and real-degradation imagery. After usability screening, deduplication, and manual review, GPT-5.6 Sol generates four low-level visual QA candidates per image, covering global state, single-region perception, spatial distribution, and inter-region relations. A human reviewer selects and, when necessary, revises one candidate to ensure that the question is unambiguous and visually supported. We then normalize the reviewed question into Q_i, associate the corresponding visual state V_i, and discard unreliable or ambiguous instances. The reviewed QA defines (T_i, y_i), while (Q_i, V_i) determines z_i, yielding anchor n_i.

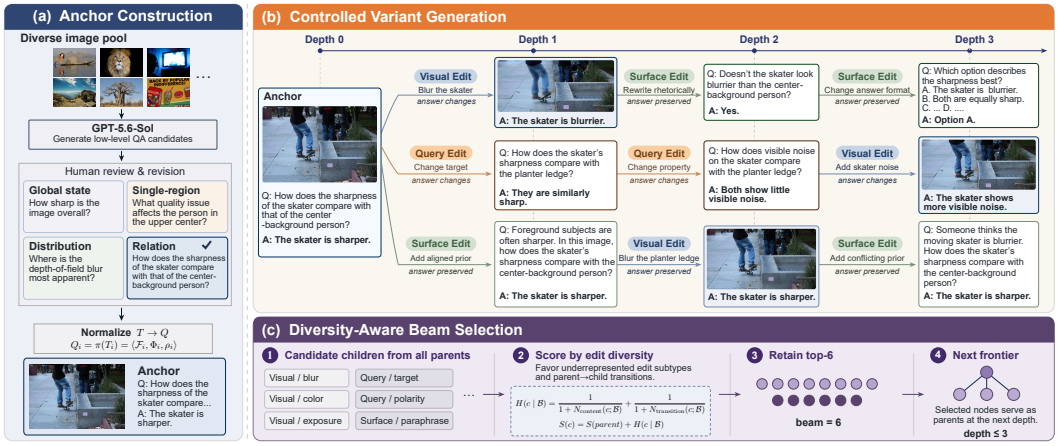


Figure 2: Illustration of CABLE benchmark construction from anchor creation to recursive controlled variant generation and diversity-aware beam selection.

Controlled Edit Taxonomy. We instantiate the atomic edits in equation 3 with fine-grained intervention families, detailed in Appendix A.2. Edit family and required response behavior are distinct: after each edit, we recompute z_v from (Q_v, V_v) and classify the transition as altered or invariant using equation 4. Surface edits modify only the question realization, including neutral paraphrases, answer-interface conversion, cross-lingual rewrites, aligned and conflicting priors, and rhetorical reformulation. Query edits modify one query component while keeping the image fixed: referent, visual property, composition operator, or polarity. Visual edits comprise queried interventions, unqueried controls, and recovery operations. Queried interventions modify evidence directly relevant to the current query, unqueried controls modify visual facts outside the queried evidence, and recovery operations attenuate or remove a previously introduced degradation.

Variant Realization. For each applicable edit family, we generate a family-constrained proposal from the parent node. Surface and query edits are realized with GPT-5.6 Sol under the atomic-edit constraints in equation 3 and family-specific constraints. A surface proposal rewrites T while preserving Q and V. A query proposal modifies only the designated component of Q with the image fixed; T and y are then synchronized with the updated query.

Visual proposals specify the target referent-property fact and its intended state change. Ordinary low-level properties are realized with deterministic rendering operators. Local interventions use SAM3 (Carion et al., 2026) masks generated from target-referent queries, followed by a grounding check. Multi-step states are reconstructed by replaying the full edit path from the root image. Restoration and AI-rerender artifacts use dedicated restoration and image-to-image pipelines under the same structured-edit formulation. Further implementation details are provided in Appendix A.2.

Before entering the candidate pool, generated variants are automatically screened for edit compliance and answerability, with visual candidates additionally checked against the intended fact change; invalid or ambiguous proposals are discarded. For each retained variant, we recompute z_v from (Q_v, V_v) and synchronize the surface reference answer y_v accordingly.

Diversity-Aware Bundle Expansion. To avoid repeatedly selecting applicable edits, CABLE uses diversity-aware beam selection, scoring valid frontier candidates at each depth by:

$$H(c \mid \mathcal{B}) = \frac{1}{1 + N_{\text{content}}(c; \mathcal{B})} + \frac{1}{1 + N_{\text{transition}}(c; \mathcal{B})}, \tag{5}$$

where N_{content} and $N_{\text{transition}}$ count prior occurrences of the same edit subtype and parent-to-child edit-type transition within the bundle, respectively. We accumulate H along each generation path, retain the top six candidates as the next frontier, and repeat this process for up to three depths to yield a construction tree, with further selection details provided in Appendix A.3.

After construction and filtering, CABLE contains 1,040 bundles and 22,733 nodes, spanning diverse low-level visual properties and degradation-related artifacts, and surface, query, and visual edit families. Detailed composition and coverage statistics are provided in Appendix A.4.

2.3 Evaluation Protocol

Query-Aware Response Projection. Evaluating response transitions requires comparing the semantic results expressed by answers whose surface forms may differ across languages, answer interfaces, or query formulations. We therefore project each response once into a query-aware typed semantic space, enabling deterministic reuse across transition comparisons.

For each query, we compile a result contract specifying its result type, structure, and value semantics, with compatible nodes sharing a space $\mathcal{Z}_{g_i}$. The reference result is aligned to this space as $\bar{z}_i \in \mathcal{Z}_{g_i}$, and the model response is projected into the same space:

$$a_i = \psi_i(\hat{y}_i) \in \mathcal{Z}_{g_i}, \tag{6}$$

where ψ_i is conditioned on T_i, Q_i, and the shared contract. Deterministic adapters handle clear yes/no and option responses, while other responses are projected by GPT-5.4-mini. The projector extracts the meaning actually expressed without access to the reference result, image, or transition context. Transition comparisons are restricted to nodes sharing the same result space, while cross-space nodes remain eligible for node-level correctness.

Node and Transition Evaluation. For each node with a usable projection, correctness is $c_i = \mathbf{1}[a_i = \bar{z}_i]$. We evaluate direct parent-child transitions conditioned on a correct parent response. Let E^+ denote the transitions whose endpoints share a result space, have usable projections, and satisfy $c_u = 1$. We partition this set into altered transitions $E^+_{\text{alt}} = \{(u, v) \in E^+ : \bar{z}_u \neq \bar{z}_v\}$ and invariant transitions $E^+_{\text{inv}} = \{(u, v) \in E^+ : \bar{z}_u = \bar{z}_v\}$, and define RA and RP as:

$$\text{RA} = \frac{1}{|E^+_{\text{alt}}|} \sum_{(u,v)\in E^+_{\text{alt}}} c_v, \qquad \text{RP} = \frac{1}{|E^+_{\text{inv}}|} \sum_{(u,v)\in E^+_{\text{inv}}} c_v. \tag{7}$$

RA measures adaptation on altered transitions, while RP measures preservation on invariant ones. For altered transitions with a correct parent, $a_u = \bar{z}_u \neq \bar{z}_v$, giving three cases:

$$\begin{aligned} \text{Correct adaptation}: &\quad a_v = \bar{z}_v, \\ \text{Response inertia}: &\quad a_v = a_u, \\ \text{Misdirected update}: &\quad a_v \neq a_u \ \wedge \ a_v \neq \bar{z}_v. \end{aligned} \tag{8}$$

RI retains the parent's semantic result despite a changed reference result, whereas MU changes the response without reaching the correct one; the three rates sum to one over eligible altered transitions.

3 Experiments

3.1 Models and Evaluation Protocol

We evaluate 29 MLLMs spanning proprietary general-purpose models, open-weight general-purpose models, and low-level-vision specialists together with their corresponding base models. Each model is evaluated zero-shot on all 22,733 nodes, with an independent response generated for each node. Models receive only the current image, surface question, and answer options when applicable. Model backends, input formatting, and inference settings are detailed in Appendix B.

3.2 Overall Performance

Table 1 summarizes the overall results on CABLE. For most higher-performing MLLMs, all-node accuracy is lower than anchor accuracy, showing that correctness on the original image-query pairs does not consistently transfer to their controlled variants. Isolated anchor performance can therefore overstate how reliably correct behavior persists under controlled intervention.

Node-level accuracy, however, does not reveal how correctness is maintained across transitions. Gemini 3.7 Flash and GPT-5.6 Sol, for example, differ by less than one point in all-node accuracy, yet Gemini adapts far more reliably under query edits, whereas GPT-5.6 Sol adapts more reliably when the required revision is induced by changed visual evidence. Comparable isolated accuracy can therefore conceal substantially different evidence-tracking patterns.

Table 1: **Main results on CABLE (%).** All and Anchor denote accuracy over all nodes and anchor nodes, respectively. Results are grouped by edit type. Specialized models are indented under their base models. RI and MU are adaptation failure rates on altered transitions, for which lower is better. Bold and underlined values indicate the best and second-best results for the corresponding metric.

	Node accuracy		Surface	Query				Visual			
Model	All↑	Anchor↑	RP↑	RA↑	RI↓	MU↓	RP↑	RA↑	RI↓	MU↓	RP↑
Proprietary General-Purpose Models											
Gemini 3.7 Flash (Google DeepMind, 2026b)	**76.14**	<u>78.10</u>	**90.01**	**88.48**	8.19	3.33	**91.78**	51.83	41.10	7.07	95.16
GPT-5.6 Sol (OpenAI, 2026)	<u>75.51</u>	**79.52**	86.61	75.27	20.55	4.17	87.61	<u>57.58</u>	33.60	8.82	90.57
Gemini 3.1 Pro (Google DeepMind, 2026a)	73.63	75.12	88.22	82.48	13.01	4.51	87.55	**58.83**	32.87	8.30	92.60
Claude Opus 5 (Anthropic, 2026a)	71.72	76.69	85.52	<u>83.67</u>	11.21	5.12	87.68	48.82	42.40	8.78	92.26
GPT-5.6 Luna (OpenAI, 2026)	70.67	74.66	85.27	68.37	25.67	5.96	85.92	50.66	41.81	7.53	90.07
GPT-5.6 Terra (OpenAI, 2026)	70.65	74.73	82.15	70.37	23.66	5.97	84.27	51.69	39.13	9.19	87.11
Claude Sonnet 5 (Anthropic, 2026b)	66.37	71.61	84.15	78.38	16.17	5.45	86.56	40.40	51.98	7.62	90.19
Claude Haiku 4.5 (Anthropic, 2025)	52.93	58.66	75.92	55.91	34.92	9.17	75.77	25.98	64.26	9.76	88.00
General-Purpose Open-Weight Models											
DeepSeek-V4-FV-Exp (DeepSeek-AI, 2026)	70.89	70.32	87.21	78.44	16.53	5.03	86.72	44.59	48.52	6.89	89.61
Qwen3.5-397B-A17B (Qwen Team, 2026)	70.79	77.97	<u>89.31</u>	73.82	22.27	3.91	87.15	41.55	55.01	3.44	**96.18**
Qwen3.5-122B-A10B (Qwen Team, 2026)	69.26	74.66	86.31	74.93	21.47	3.60	<u>88.35</u>	40.27	56.23	3.50	94.68
Qwen3.5-35B-A3B (Qwen Team, 2026)	65.13	73.39	85.81	59.81	37.15	3.03	84.41	32.24	63.93	3.83	95.19
Gemma 4 12B IT (Team, 2026)	61.07	65.23	82.62	74.82	19.31	5.87	81.04	33.98	58.31	7.71	93.06
Molmo2-8B (Clark et al., 2026)	60.39	62.31	80.48	64.09	30.76	5.16	78.09	39.39	52.14	8.47	94.53
InternVL3.5-8B (Wang et al., 2025b)	52.33	53.81	75.69	61.24	29.94	8.82	81.48	17.71	78.40	3.89	94.55
Low-Level-Vision Specialists and Their Bases											
LLaVA-OV-7B (Li et al., 2025a)	56.11	59.70	78.20	58.39	37.58	4.04	84.26	46.45	47.83	5.72	94.63
Q-SiT (Zhang et al., 2025)	56.20	57.84	80.67	67.48	29.61	2.91	81.74	41.04	54.20	4.76	95.10
LLaVA-OV-0.5B (Li et al., 2025a)	44.76	46.95	70.89	31.25	64.21	4.54	74.35	31.30	62.27	6.43	93.42
Q-SiT-Mini (Zhang et al., 2025)	48.18	45.78	72.37	44.05	51.86	4.08	81.75	27.91	66.50	5.58	94.99
Qwen2.5-VL-7B (Wang et al., 2024)	53.05	53.05	76.87	57.41	34.26	8.33	79.97	25.56	67.83	6.61	94.35
Q-Insight (Li et al., 2025c)	51.54	50.35	77.48	58.53	33.24	8.22	81.40	26.54	65.94	7.52	94.52
Qwen2.5-VL-3B (Wang et al., 2024)	51.39	53.41	73.50	40.31	53.14	6.55	76.46	24.63	70.23	5.13	92.67
Robust-R1-SFT (Tang et al., 2026)	49.34	48.91	73.55	49.23	43.71	7.06	77.67	23.83	71.23	4.94	92.76
Robust-R1-RL (Tang et al., 2026)	49.94	49.78	73.50	49.64	43.03	7.33	76.88	25.28	70.78	3.94	93.32
mPLUG-Owl2-7B (Ye et al., 2023)	43.24	38.48	67.12	50.89	40.35	8.76	76.68	16.68	78.15	5.17	88.23
Q-Instruct-mPLUG (Wu et al., 2024b)	49.06	44.74	73.68	53.59	42.28	4.13	81.49	21.12	72.54	6.34	89.73
LLaVA-v1.5-7B (Liu et al., 2023)	45.13	41.82	70.54	37.38	57.49	5.13	80.02	19.59	75.92	4.49	94.99
Q-Instruct-LLaVA (Wu et al., 2024b)	48.07	44.63	77.69	51.39	45.13	3.48	81.80	22.91	73.06	4.04	<u>96.01</u>
DepictQA-v2 (You et al., 2025b)	44.03	37.17	72.86	44.23	51.90	3.87	78.42	18.69	73.94	7.37	87.94

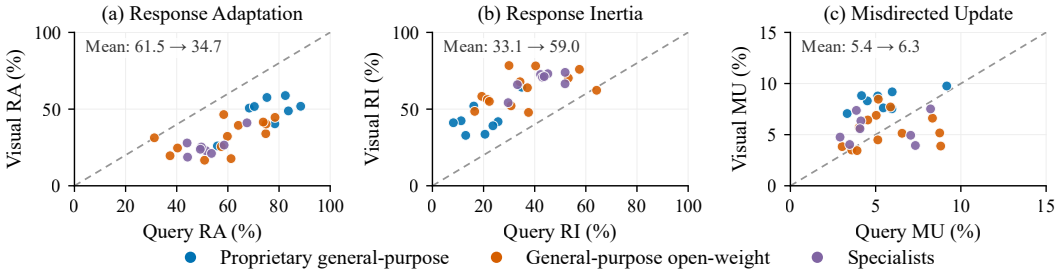


Figure 3: **Query-visual adaptation asymmetry.** Visual interventions show lower RA and higher RI than query interventions, while MU remains comparatively stable.

Across models, invariant transitions are generally preserved more reliably than altered transitions are adapted to. This asymmetry is particularly pronounced for visual interventions: models that often revise their responses after a query change frequently retain the previous result when the required revision is instead induced by changed visual evidence. We next examine this query-visual adaptation gap and the failure modes underlying it.

3.3 Adaptation and Response Inertia

The adaptation-preservation asymmetry in Table 1 is most pronounced under visual interventions. Figure 3 further reveals a systematic query-visual adaptation gap: visual RA is lower than query RA for 28 of 29 models across all three model groups, with the mean decreasing from 61.5% to

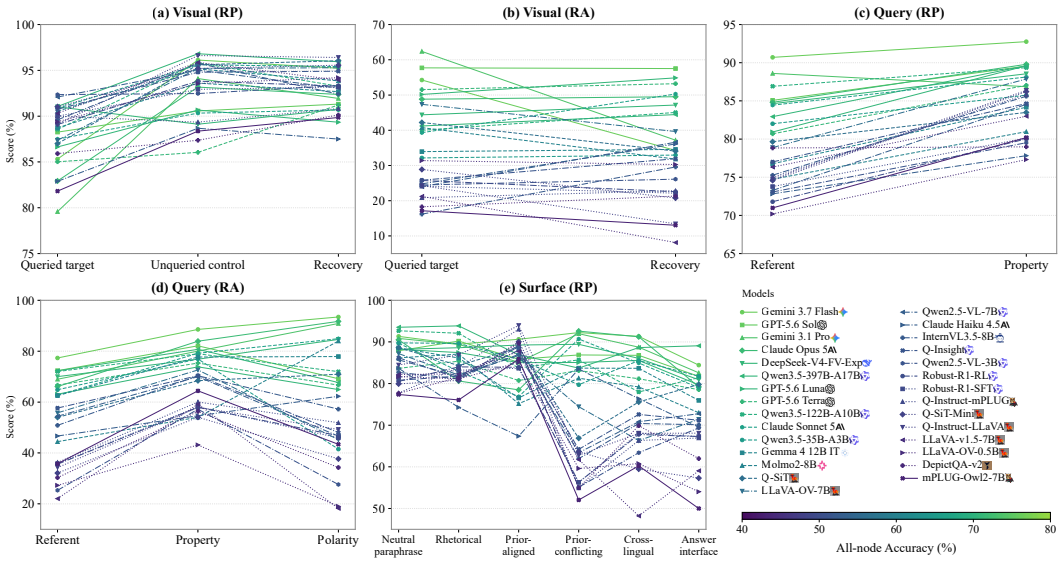


Figure 4: **RP and RA across controlled intervention subtypes.** Unqueried-control edits are shown only for RP, polarity edits only for RA, and query-operator edits are omitted because they change the result space, precluding direct comparison of RA and RP.

34.7%. The failure decomposition further clarifies this gap. Mean RI increases from 33.1% under query edits to 59.0% under visual edits, rising for 28 of 29 models, whereas MU changes only modestly from 5.4% to 6.3%. The reduction in visual RA therefore corresponds predominantly to increased response inertia rather than to more frequent updates toward another incorrect result. In other words, when changed visual evidence requires a different result, models more often retain the previously correct judgment than revise it incorrectly (Figure 5A). Both the gap and its inertia-dominated composition persist under bundle-level aggregation (Appendix C.2).

3.4 Selectivity under Controlled Interventions

Visual edit behavior. Visual interventions distinguish changes to query-relevant evidence from changes outside the evidence required by the current query. As shown in Figure 4(a)(b), models generally achieve the highest RP on Unqueried-control edits, indicating that they can often preserve a correct judgment when visual changes are irrelevant to the queried fact. Preservation is less reliable when the intervention directly affects queried evidence, even when the correct result remains unchanged. This contrast suggests that model responses are substantially more robust to irrelevant visual perturbations than to changes in visual content directly implicated by the query.

Selectivity under query changes. Query edits keep the image fixed while changing which visual fact is queried, providing a controlled view of how models condition their judgments on different aspects of the same visual input. As shown in Figure 4(c)(d), referent edits are consistently more difficult than property edits under both preservation and adaptation. The persistence of this gap across both response requirements suggests that the difficulty is associated with how models condition on the revised query, rather than with any single type of required response behavior.

One possible explanation is that changing the referent places a stronger demand on the spatial binding of low-level visual evidence. Property edits retain the same visual referent and only change the low-level dimension being queried, whereas referent edits require the same type of judgment to be reassociated with a different visual entity or region. This interpretation is consistent with prior work showing that vision-language models can recognize familiar concepts while remaining substantially less reliable at compositional grounding and at binding attributes or relations to the correct visual entities (Thrush et al., 2022; Zeng et al., 2024). The referent-property asymmetry therefore suggests a possible separation between low-level attribute sensitivity and the spatially selective use of that evidence: models may retain useful information about properties such as blur, noise, or exposure while remaining less reliable at rebinding that evidence to the referent specified by a revised query.

Table 2: **Fine-grained effects of low-level visual specialization (%).** Values are macro-averaged over seven matched base-specialist comparisons. Δ denotes Specialist − Base in percentage points, and ↑ Pairs counts matched comparisons in which specialization improves the metric.

	Surface RP						Query RA			Visual RA	
	Neutral para.	Rhetor-ical	Prior aligned	Prior conflict.	Cross-lingual	Answer interface	Referent	Property	Polarity	Queried interv.	Recovery
Base	82.24	80.98	87.24	60.21	61.88	63.71	35.13	60.54	32.53	27.16	25.70
Specialist	82.71	83.47	88.87	60.71	68.27	68.27	40.81	60.98	50.08	27.42	23.97
Δ (pp)	+0.47	+2.49	+1.63	+0.51	**+6.39**	+4.55	**+5.68**	+0.44	**+17.55**	+0.26	−1.74
↑ Pairs	5/7	5/7	5/7	6/7	**6/7**	4/7	**6/7**	4/7	**7/7**	4/7	2/7

Linguistic robustness and priors. As shown in Figure 4(e), neutral paraphrases and rhetorical rewrites largely preserve the overall capability ordering: stronger models maintain high RP, whereas weaker models are more sensitive to changes in wording. Cross-lingual rewrites further widen this gap, suggesting that multilingual comprehension introduces an additional bottleneck even when the underlying visual judgment is unchanged. A similar capability-dependent separation appears for polarity edits in Figure 4(c), where stronger models adapt reliably after the logical constraint is reversed while weaker models often fail to do so. These patterns suggest that low-level visual perception is not expressed independently of language: even with unchanged visual evidence, performance can vary substantially with linguistic formulation.

The effect becomes more directional under injected linguistic priors. Prior-aligned rewrites can yield unusually high RP for several lower-performing models, sometimes approaching stronger models, whereas prior-conflicting rewrites cause sharp drops in many of the same models (Figure 5B) while stronger models remain comparatively stable. Thus, high RP under an aligned prior need not indicate stronger evidence tracking. It may instead reflect agreement between the linguistic cue and visual evidence. The greater robustness of stronger models to conflicting priors is consistent with a greater tendency to re-evaluate linguistic cues against the visual input rather than treat them as authoritative, aligning with Deng et al. (2025), who report strong textual bias under vision-language conflict that is modestly reduced with larger language models.

Answer-interface changes expose sensitivity to how an otherwise unchanged visual judgment is elicited. As shown in Figure 4(e), this effect is strongly capability-dependent: stronger models remain stable, whereas weaker models often suffer large RP drops. The difficulty is not merely output formatting. For example, verifying whether an image contains blur provides the candidate concept explicitly, whereas describing all visible degradations requires the model to retrieve and express that concept independently. Thus, the same underlying visual judgment can become substantially harder to elicit under a different interface. This sensitivity implies that evaluation through a single answer interface may conflate low-level visual competence with the ability to access and express that competence under a particular elicitation format.

3.5 Effects of Low-Level Visual Specialization

We next ask what low-level visual specialization actually improves. Table 2 reveals a pronounced imbalance in where its gains occur. Across matched base-specialist pairs, gains concentrate on query- and language-side changes: adaptation improves in all seven pairs under polarity reversals, often by a large margin, and in six of seven under referent changes, while robustness also increases for demanding surface variations such as cross-lingual rewrites. In contrast, gains are limited under low-level property changes and largely disappear when adaptation must be triggered by changed visual evidence. A per-pair analysis confirms that this imbalance is not driven by a few pairs: in five of seven pairs, the query-side gain significantly exceeds the visual-side gain, and in none is it significantly smaller (Appendix C.3). Thus, the aggregate improvement of specialized models on low-level tasks does not reflect a uniform strengthening of low-level evidence tracking. It is substantially more pronounced in how models condition responses on the task specification than in how their responses follow changes in visual evidence.

This distinction is particularly relevant to current low-level specialization, whose supervision is typically imposed through task responses, instruction following, reasoning traces, or output-level rewards. For example, Q-Instruct converts low-level human feedback into visual instruction-response data for multimodal instruction tuning (Wu et al., 2024b). Optimizing these objectives can improve

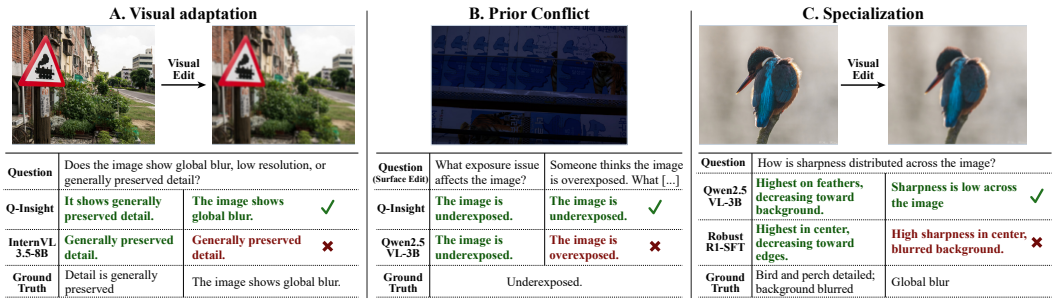


Figure 5: **Qualitative examples of transition-level model behavior.**

how a model interprets and answers a low-level visual task without necessarily making its prediction more dependent on the relevant visual evidence. This interpretation is consistent with prior findings that visual information can be available to an MLLM without reliably controlling its generated response (Li et al., 2026b). CABLE makes this gap observable at the transition level (Figure 5C).

4 Related Work

MLLMs for Low-Level Visual Perception. Recent work extends MLLMs to low-level visual perception through both the evaluation of general-purpose models and the development of specialized models for image quality and degradation understanding. Benchmarks such as Q-Bench (Wu et al., 2024a) and its extensions evaluate capabilities ranging from basic low-level attribute perception to comparative quality judgments under diverse protocols (Zhang et al., 2024b; Wu et al., 2024e). Meanwhile, specialized instruction tuning, such as Q-Instruct (Wu et al., 2024b), supports expert MLLMs for quality scoring (Wu et al., 2024c; You et al., 2025a), comparative assessment (You et al., 2024; 2025b; Wu et al., 2024d), degradation understanding (Li et al., 2025c), and distortion localization (Chen et al., 2024a). While these efforts substantially broaden low-level MLLM evaluation, they primarily assess individual instances rather than transitions across controlled variants.

Evaluation under Controlled Transformations. Controlled transformations are widely used to test whether model predictions respond appropriately to related inputs. Early work studies rephrased, entailed, or semantically related questions (Shah et al., 2019; Hudson & Manning, 2019; Ray et al., 2019), while later methods introduce controlled question or image transformations with predefined output relations (Yuan et al., 2021; Agarwal et al., 2020; Jimenez et al., 2022; Jang et al., 2022). Recent LVLM benchmarks further extend this paradigm to prompt variations and visual perturbations (Zhang et al., 2024a; Chou et al., 2025). CABLE builds on controlled-transformation evaluation but shifts the focus from relation satisfaction to selective evidence tracking. By conditioning on an initially correct response and evaluating its transition under controlled interventions, CABLE separates failures to adapt from failures to preserve, exposing directional evidence-tracking behavior that isolated accuracy or aggregate consistency can obscure.

5 Conclusion

We introduce CABLE to evaluate whether MLLMs selectively track low-level visual evidence through controlled response transitions. Across 29 models, we find that selective evidence tracking is asymmetric rather than absent. Models generally preserve correct judgments under invariant or irrelevant changes, but are substantially less reliable at revising them when query-relevant visual evidence changes. This gap is driven primarily by response inertia rather than misdirected updates, indicating that the main limitation is not indiscriminate instability, but insufficient visual control over response revision. We further find that low-level specialization improves task and query conditioning more consistently than adaptation to changed visual evidence, showing that stronger low-level task performance does not necessarily imply stronger visual evidence tracking. These results shift evaluation from asking only whether a model answers a low-level visual question correctly to whether its response remains behaviorally responsive to the evidence that should determine it. Future work could use such transition-level signals to strengthen visual control over revision.

AI USE STATEMENT

The core research question and overall methodological direction of this work were developed by the authors. Generative AI tools were used to provide suggestions for methodological refinement. Examples include the design of fine-grained edit and degradation taxonomies and the normalization procedure. These suggestions were discussed by the authors and selectively adopted. Generative AI tools were also extensively used in the implementation of the research code and supporting tooling, including the interface used to facilitate human review. The authors specified the intended methodology and functionality, and validated the resulting implementation and experimental outputs. Generative AI models were additionally used as components of the CABLE benchmark construction pipeline, as described in Section 2. We also used generative AI tools for language editing and polishing of the manuscript. All final methodological decisions, analyses, and conclusions were made and reviewed by the authors, who take responsibility for the final content of this work.

REFERENCES

Vedika Agarwal, Rakshith Shetty, and Mario Fritz. Towards causal VQA: revealing and reducing spurious correlations by invariant and covariant semantic editing. In *CVPR*, 2020.

Eirikur Agustsson and Radu Timofte. NTIRE 2017 challenge on single image super-resolution: Dataset and study. In *CVPRW*, 2017.

Anthropic. System Card: Claude Haiku 4.5, 2025. URL https://www-cdn.anthropic.com/7aad69bf12627d42234e01ee7c36305dc2f6a970.pdf.

Anthropic. Claude Opus 5 System Card, 2026a. URL https://www.anthropic.com/claude-opus-5-system-card.

Anthropic. Claude Sonnet 5 System Card, 2026b. URL https://www.anthropic.com/claude-sonnet-5-system-card.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris Coll-Vinent, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, RISHI HAZRA, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollar, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. SAM 3: Segment anything with concepts. In *ICLR*, 2026.

Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021.

Baoliang Chen, Siyi Pan, Dongxu Wu, Liang Xie, Xiangjie Sui, Lingyu Zhu, and Hanwei Zhu. Mitigating perception bias: A training-free approach to enhance LMM for image quality assessment. In *AAAI*, 2026.

Chaofeng Chen, Sensen Yang, Haoning Wu, Liang Liao, Zicheng Zhang, Annan Wang, Wenxiu Sun, Qiong Yan, and Weisi Lin. Q-ground: Image quality grounding with large multi-modality models. In *ACM MM*, 2024a.

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

Shih-Han Chou, Shivam Chandhok, Jim Little, and Leonid Sigal. MM-R^3: On (in-)consistency of vision-language models (VLMs). In *Findings of ACL*, 2025.

Christopher Clark, Jieyu Zhang, Zixian Ma, Jae Sung Park, Mohammadreza Salehi, Rohun Tripathi, Sangho Lee, Zhongzheng Ren, Chris Dongjoo Kim, Yinuo Yang, Vincent Shao, Yue Yang, Weikai Huang, Ziqi Gao, Taira Anderson, Jianrui Zhang, Jitesh Jain, George Stoica, Winson Han, Ali Farhadi, and Ranjay Krishna. Molmo2: Open weights and data for vision-language models with video understanding and grounding. *arXiv preprint arXiv:2601.10611*, 2026.

Yuning Cui, Syed Waqas Zamir, Salman Khan, Alois Knoll, Mubarak Shah, and Fahad Shahbaz Khan. AdaIR: Adaptive all-in-one image restoration via frequency mining and modulation. In *ICLR*, 2025.

DeepSeek-AI. Deepseek-v4.1-flash: Pushing the limits of kv cache compression, 2026.

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

Alvan R. Feinstein and Domenic V. Cicchetti. High agreement but low kappa: I. the problems of two paradoxes. *Journal of clinical epidemiology*, 43 6:543–9, 1990.

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

Google DeepMind. Gemini 3.7 Flash – model card, 2026b. URL `https://deepmind.google/models/model-cards/gemini-3-7-flash/`.

Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. Hallusionbench: An advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *CVPR*, 2024.

Kilem L. Gwet. Computing inter-rater reliability and its variance in the presence of high agreement. *The British journal of mathematical and statistical psychology*, 61 Pt 1:29–48, 2008.

Drew A. Hudson and Christopher D. Manning. GQA: A new dataset for real-world visual reasoning and compositquestion answering. In *CVPR*, 2019.

Myeongjun Jang, Deuk Sin Kwon, and Thomas Lukasiewicz. BECEL: Benchmark for consistency evaluation of language models. In *COLING*, 2022.

Yitong Jiang, Zhaoyang Zhang, Tianfan Xue, and Jinwei Gu. Autodir: Automatic all-in-one image restoration with latent diffusion. In *ECCV*, 2024.

Carlos E. Jimenez, Olga Russakovsky, and Karthik Narasimhan. CARETS: A consistency and robustness evaluative test suite for VQA. In *ACL*, 2022.

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 *SOSP*, 2023.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *Trans. Mach. Learn. Res.*, 2025a.

Hao Li, Xiang Chen, Jiangxin Dong, Jinhui Tang, and Jinshan Pan. Foundir: Unleashing million-scale training data to advance foundation models for image restoration. In *ICCV*, 2025b.

Hao Li, Zeyu Xiao, Junhao Zhou, Peng Liu, Yang Zhao, and Wei Jia. Robust vision-language models via manifold-adversarial adapters. In *ICML*, 2026a.

Jiaang Li, Chengzu Li, Zhaochong An, Yifei Yuan, Xi Liu, Serge Belongie, and Vésteinn Snæbjarnarson. Seeing or knowing? visual context sensitivity in multimodal large language models. *arXiv preprint arXiv:2607.26326*, 2026b.

Weiqi Li, Xuanyu Zhang, Shijie Zhao, Yabin Zhang, Junlin Li, Li Zhang, and Jian Jun Zhang. Q-insight: Understanding image quality via visual reinforcement learning. In *NeurIPS*, 2025c.

Yuan Li, Zitang Sun, Yen-Ju Chen, and Shin'ya Nishida. Investigate the low-level visual perception in vision-language based image quality assessment. *arXiv preprint arXiv:2512.09573*, 2025d.

Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *CVPRW*, 2017.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

Zhiqiu Lin, Xinyue Chen, Deepak Pathak, Pengchuan Zhang, and Deva Ramanan. Revisiting the role of language priors in vision-language models. In *ICML*, 2024.

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

OpenAI. GPT-5.6 System Card, 2026. URL `https://deploymentsafety.openai.com/gpt-5-6`.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *ICLR*, 2024.

Qwen Team. Qwen3.5: Towards native multimodalagents, February 2026.

Arijit Ray, Karan Sikka, Ajay Divakaran, Stefan Lee, and Giedrius Burachas. Sunny and dark outside?! improving answer consistency in VQA through entailed question generation. In *EMNLP-IJCNLP*, 2019.

Jaesung Rim, Haeyun Lee, Jucheol Won, and Sunghyun Cho. Real-world blur dataset for learning and benchmarking deblurring algorithms. In *ECCV*, 2020.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

Meet Shah, Xinlei Chen, Marcus Rohrbach, and Devi Parikh. Cycle-consistency for robust visual question answering. In *CVPR*, 2019.

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

Jiaqi Tang, Jianmin Chen, Wei Wei, Xiaogang Xu, Runtao Liu, Xiangyu Wu, Qipeng Xie, Jiafei Wu, Lei Zhang, and Qifeng Chen. Robust-r1: Degradation-aware reasoning for robust visual understanding. In *AAAI*, 2026.

Gemma Team. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026.

Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. In *CVPR*, 2022.

Chenxi Wang, Xiang Chen, Ningyu Zhang, Bozhong Tian, Haoming Xu, Shumin Deng, and Huajun Chen. MLLM can see? dynamic correction decoding for hallucination mitigation. In *ICLR*, 2025a.

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

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*, 2025b.

Haoning Wu, Zicheng Zhang, Erli Zhang, Chaofeng Chen, Liang Liao, Annan Wang, Chunyi Li, Wenxiu Sun, Qiong Yan, Guangtao Zhai, and Weisi Lin. Q-bench: A benchmark for general-purpose foundation models on low-level vision. In *ICLR*, 2024a.

Haoning Wu, Zicheng Zhang, Erli Zhang, Chaofeng Chen, Liang Liao, Annan Wang, Kaixin Xu, Chunyi Li, Jingwen Hou, Guangtao Zhai, Geng Xue, Wenxiu Sun, Qiong Yan, and Weisi Lin. Q-instruct: Improving low-level visual abilities for multi-modality foundation models. In *CVPR*, 2024b.

Haoning Wu, Zicheng Zhang, Weixia Zhang, Chaofeng Chen, Liang Liao, Chunyi Li, Yixuan Gao, Annan Wang, Erli Zhang, Wenxiu Sun, Qiong Yan, Xiongkuo Min, Guangtao Zhai, and Weisi Lin. Q-align: Teaching LMMs for visual scoring via discrete text-defined levels. In *ICML*, 2024c.

Haoning Wu, Hanwei Zhu, Zicheng Zhang, Erli Zhang, Chaofeng Chen, Liang Liao, Chunyi Li, Annan Wang, Wenxiu Sun, Qiong Yan, Xiaohong Liu, Guangtao Zhai, Shiqi Wang, and Weisi Lin. Towards open-ended visual quality comparison. In *ECCV*, 2024d.

Tianhe Wu, Kede Ma, Jie Liang, Yujiu Yang, and Lei Zhang. A comprehensive study of multimodal large language models for image quality assessment. In *ECCV*, 2024e.

Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. *arXiv preprint arXiv:2311.04257*, 2023.

Zhiyuan You, Zheyuan Li, Jinjin Gu, Zhenfei Yin, Tianfan Xue, and Chao Dong. Depicting beyond scores: Advancing image quality assessment through multi-modal language models. In *ECCV*, 2024.

Zhiyuan You, Xin Cai, Jinjin Gu, Tianfan Xue, and Chao Dong. Teaching large language models to regress accurate image quality scores using score distribution. In *CVPR*, 2025a.

Zhiyuan You, Jinjin Gu, Xin Cai, Zheyuan Li, Kaiwen Zhu, Chao Dong, and Tianfan Xue. Enhancing descriptive image quality assessment with a large-scale multi-modal dataset. *IEEE Trans. Image Process.*, 34:8201–8215, 2025b.

Yuanyuan Yuan, Shuai Wang, Mingyue Jiang, and Tsong Yueh Chen. Perception matters: Detecting perception failures of VQA models using metamorphic testing. In *CVPR*, 2021.

Yunan Zeng, Yan Huang, Jinjin Zhang, Zequn Jie, Zhenhua Chai, and Liang Wang. Investigating compositional challenges in vision-language models for visual grounding. In *CVPR*, 2024.

Yuan Zhang, Fei Xiao, Tao Huang, Chun-Kai Fan, Hongyuan Dong, Jiawen Li, Jiacong Wang, Kuan Cheng, Shanghang Zhang, and Haoyuan Guo. Unveiling the tapestry of consistency in large vision-language models. In *NeurIPS*, 2024a.

Zicheng Zhang, Haoning Wu, Erli Zhang, Guangtao Zhai, and Weisi Lin. Q-bench+: A benchmark for multi-modal foundation models on low-level vision from single images to pairs. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(12):10404–10418, 2024b.

Zicheng Zhang, Haoning Wu, Ziheng Jia, Weisi Lin, and Guangtao Zhai. Teaching lmms for image quality scoring and interpreting. *arXiv preprint arXiv:2503.09197*, 2025.

A Additional Benchmark Construction Details

All MLLM-based steps in benchmark construction, including QA generation, query normalization, edit realization, and automatic validation, use GPT-5.6 Sol, hereafter the construction model.

A.1 Anchor Construction and Normalization

Anchor construction consists of two stages. We first construct and manually review image-question pairs that provide reliable low-level visual anchors, and then normalize each accepted QA pair into the structured representation used for controlled editing and transition evaluation.

Image pool and filtering. We construct the candidate image pool from eight heterogeneous sources covering natural photographs, web imagery, text-rich scenes, and real degradations. For paired degradation datasets, we use the reference image as the anchor candidate and retain the corresponding degraded image for later variant construction. The raw pools are summarized in Table 3.

We remove images that cannot be decoded into usable RGB inputs. For datasets containing multiple closely related images of the same scene, such as RealBlur and FoundIR, we apply scene-level deduplication to avoid repeatedly sampling near-identical captures or variants. From the resulting pools, we sample 1,600 candidate images for first-stage human review. This review is performed before QA generation and retains images with sufficiently clear objects, regions, or global structure to support image-grounded low-level questions. It yields 1,186 images for subsequent QA construction.

Anchor QA generation and review. For each retained image, we generate four QA candidates with four query-specific prompt templates corresponding to complementary levels of low-level visual questioning. Each generation call receives the image and the requested query type, but not dataset identity, filenames, or other source metadata. Questions are required to depend on visible low-level evidence rather than high-level scene semantics or expected object appearance. The query types and supported answer interfaces are summarized in Table 4.

The query type and answer interface are determined independently. After the four candidates are generated, a separate call to the construction model compares them and selects one default candidate. The selection prompt favors visual correctness, image dependence, clarity, and relevance to low-level visual evidence, while discouraging redundant or potentially confounded candidates.

All 1,186 model-selected QA pairs then undergo human review. Reviewers inspect the image and generated candidates, and may retain the default choice, replace it with another candidate for the same image, or revise the question and reference answer when necessary. They verify that the answer is visually supported, the relevant evidence is sufficiently clear, the queried referent and low-level property are unambiguous, and the answer does not follow primarily from linguistic or scene priors. Questions are rejected when these conditions cannot be satisfied. This process yields 1,040 final anchors from the 1,186 reviewed records.

Query normalization. Following the formulation in Section 2, we convert each accepted image-QA pair into a structured JSON representation. The normalization input consists of the image, the human-reviewed question and reference answer, and the answer interface with its options when applicable; it excludes the source dataset, filename, and other construction metadata.

The resulting JSON explicitly records the referents mentioned by the question, the queried visual facts associated with those referents, the expression that composes these facts, and the query polarity corresponding to $\mathcal{F}$, Φ, and ρ in equation 1. It also stores the resulting query result z together with its surface reference answer y. For local referents, the normalization includes a content-based description that can subsequently be used for spatial grounding during localized visual edits. For choice-based questions, the semantic meaning of each option is stored separately from its displayed label so that option reordering or answer-interface changes do not alter the underlying result. Normalization outputs are checked programmatically for schema validity, answer-type compatibility, option coverage when applicable, and internal structural consistency. The construction model also reports whether its structured interpretation is consistent with the provided reference answer. Invalid structural outputs are retried with validation feedback, while ambiguous or unsupported parses are otherwise excluded or manually resolved during finalization.

Table 3: Raw image pools used for anchor construction.

Source	CC12M	COCO	DIV2K	Flickr2K	FoundIR	LAION	RealBlur	TextVQA
Raw pool	10,000	5,000	800	2,650	3,000	10,000	8,368	5,000

Table 4: Anchor QA types and response interfaces.

Query type	Description	Response interface	Constraint
Global state	Image-wide visual condition	Yes/no	Binary judgment
Single region	Low-level state of an object or region	Single-choice	4-6 options; one correct
Spatial distribution	Location or extent of a low-level property	Multi-select	4-6 options; multiple correct
Inter-region relation	Low-level relation between identifiable regions	Short answer	Concise free-form response

Visual-state representation. In the implementation, V is represented by the image state associated with a node together with the query-local visual semantics produced during normalization. For an anchor, the root image and the rendered image are identical, and the visual edit path is empty. The normalization does not attempt to construct a dense annotation of all low-level properties in the image. Instead, it records only the visual facts required by the accepted query and the query result z implied by those facts.

This representation is updated during controlled visual editing. Each visual variant keeps the same root image, records the sequence of applied edits, and points to the rendered image obtained by replaying that edit sequence from the root. The corresponding query-local visual facts and query result z are then updated to reflect the edited image state. This lets CABLE determine whether a transition changes the query result without densely annotating unrelated visual properties.

A.2 Controlled Edit Families and Realization

Given the normalized anchor representation, we construct controlled variants by intervening on the surface realization T, structured query Q, or visual state V. The central design principle is *atomicity*: each edit targets one of these components while satisfying the component-wise constraints defined in equation 3. This separation allows the resulting parent-child pairs to test whether model responses track the aspect of the image-query state that was actually changed.

Importantly, the edit family specifies the intended intervention rather than the required transition behavior. After a candidate child is realized and validated, its query result z_v is determined under the structured representation in equation 2. For parent-child pairs whose results are semantically comparable under the evaluation protocol, the transition is classified as invariant if $z_u = z_v$ and altered otherwise, following equation 4. Thus, an edit subtype does not by itself determine whether response preservation or adaptation is required. Surface and query proposals undergo automatic semantic validation before acceptance, as described below.

Surface edits. Surface edits vary how the same visual query is expressed while preserving the structured query and visual state. Q and V, and hence the query result z, remain fixed, whereas the language, answer interface, or pragmatic framing in T may change. The surface reference answer y is regenerated when required by the modified realization. We instantiate six surface strategies:

- **Neutral paraphrase** rewrites the question in the same language and answer interface while preserving the structured query Q and query result z.
- **Answer-interface conversion** changes the format in which the answer is elicited, for example between free-form, yes/no, and multiple-choice forms, while preserving the underlying query Q and query result z.
- **Cross-lingual rewriting** translates the question and its surface reference answer into another language while preserving the underlying query Q and query result z.

Table 5: Controlled edit families and their component-wise constraints.

Family	Subtypes	Edited	Preserved
Surface	Neutral paraphrase, answer interface, cross-lingual, aligned prior, conflicting prior, rhetorical	T	Q and V (hence z)
Query	Referent, property, expression, polarity	Q; T updated	V
Visual	Queried intervention, unqueried control, recovery	V	T and Q

- **Prior-aligned** rewriting adds an attributed linguistic cue that is consistent with the query result z.
- **Prior-conflicting** rewriting adds an attributed linguistic cue that conflicts with the query result z.
- **Rhetorical** rewriting changes the question into a conversational confirmation, tag question, or indirect challenge while preserving the structured query Q, including its polarity.

Surface proposals are accepted only when they preserve the structured query Q and visual state V, and hence the query result z, while modifying only the requested aspect of the surface realization. The resulting question must remain answerable from the image and admit a valid surface reference answer y under its answer interface. For edits that introduce linguistic priors, the added cue must not by itself determine the answer.

Query edits. Query edits keep the visual state V fixed while changing one designated component of the structured query Q. The natural-language realization T is updated to express the modified query, and the resulting query result z is recomputed from the updated Q and unchanged V. The surface reference answer y is then regenerated as needed to express z under the new question and answer interface. We instantiate four query-edit families:

- **Referent edits** replace the referent of one queried referent-property fact while preserving its low-level property and, where possible, the surrounding query structure.
- **Property edits** replace the low-level property associated with a queried referent while preserving the referent and, where possible, the surrounding query structure.
- **Expression edits** modify the composition operator Φ, changing how the queried facts are read, compared, selected, tested, or composed while preserving the underlying referent-property facts where applicable.
- **Polarity edits** modify the polarity term ρ by switching between identity and negation for a Boolean query while preserving the underlying positive visual condition.

A query proposal is retained only when the unchanged visual state supports an unambiguous result for the updated query, the written question T faithfully realizes the proposed Q, and the designated query component is the only intended semantic change. We then recompute the query result z from (Q, V) and verify that the surface reference answer y correctly expresses that result under the resulting answer interface. This validation is important because a syntactically well-formed query edit may still produce a question that is unsupported or ambiguous for the current image.

Not every valid query edit yields a parent-child pair that is directly comparable for transition-level evaluation. Some query edits may change the result type or semantic space and therefore yield parent-child pairs that are not directly comparable for transition-level evaluation. Such children remain valid benchmark nodes, but parent-child transitions are scored only when both endpoints share a compatible result space under the evaluation protocol in Section 2.

Visual edits. Visual edits preserve the surface question T and structured query Q while modifying the visual state V. At the semantic level, visual edits are organized into three families. A **queried intervention** modifies a referent-property fact that is directly used by the current query. An **unqueried control** modifies a region or low-level property outside the evidence required by the current query. A **recovery** edit applies a recovery operation intended to attenuate or remove a degradation affecting the queried visual fact, including both reversing a previously introduced synthetic degradation and applying an image-restoration (IR) method.

Table 6: Synthetic visual operators and candidate parameter settings.

Operator	Candidate settings
Blur	Gaussian ($r = 1.5, 3.5, 7.0$); defocus ($r = 2, 4, 7$); motion ($\ell = 7, 13, 21$)
Noise	Gaussian ($\sigma = 5, 12, 25$); Poisson (peak $= 60, 20, 8$); impulse (amount $= 0.005, 0.02, 0.05$)
Compression	JPEG (quality $= 65, 35, 12$); JPEG2000 (compression ratio $= 10, 30, 70$)
Resolution	Bicubic down–up (scale $= 0.65, 0.40, 0.20$); pixelation (scale $= 0.35, 0.18, 0.08$)
Underexposure	Brightness factor $= 0.72, 0.48, 0.28$
Overexposure	Brightness factor $= 1.3, 1.7, 2.3$
Low contrast	Contrast factor $= 0.72, 0.48, 0.25$
Saturation shift	Saturation factor $= 0.45, 0.15, 1.55, 2.1$
Color cast	Cast color chosen from the six RGB primary and secondary colors. $\alpha = 0.18$
Color quantization	Palette size $\in \{32, 16, 8\}$ colors

Queried interventions are more likely to change the resulting query result, whereas unqueried controls more often leave it unchanged. Recovery edits may either change or preserve the query result depending on whether the recovered visual evidence is outcome-determining for the current query. In all cases, the final transition relation is determined only after the rendered image is validated and the query result z is recomputed from the updated (Q, V) via equation 2.

Deterministic visual operators. To realize ordinary synthetic visual edits, we apply deterministic image operators using a small discrete set of candidate parameter settings, as summarized in Table 6. For additive or increasing interventions, the renderer materializes the candidate settings in the corresponding parameter grid, and the rendered candidates are subsequently validated jointly to select a valid realization. Recovery or decreasing operations instead construct a small set of candidates that weaken or reverse the relevant operation.

JPEG and JPEG2000 variants are generated through in-memory encode-decode round trips, while the resulting benchmark images are stored in PNG format. For localized edits, the transformed image is composited with the original image using the target mask, with slight boundary feathering to avoid visible seams.

Spatial grounding and visual-state replay. Localized visual edits require the edited region to correspond to the target referent specified by the structured query Q. Each local visual proposal therefore specifies the target referent, a grounding description, a SAM3 query, and the intended visual change. The SAM3 query generates candidate masks for the target object or region.

Before rendering, the candidate grounding is checked automatically against the original image, numbered mask overlays, masked crops, and the proposal's referent description. Only masks judged to correspond to the intended referent are retained; proposals with incorrect or ambiguous grounding are rejected before visual rendering. The validated mask is then used to restrict the visual operation to the target region. This prevents a nominally local edit from affecting an unrelated object or region. Full-image operations such as image restoration and rerendering bypass this localization step.

Post-render visual validation. The specified visual operation and its parameters describe an intended intervention, but do not by themselves guarantee that a rendered image realizes the proposed post-edit state. For each ordinary visual proposal, we therefore render a set of candidate images and pass them to an automatic validator. The validator is given the parent state, the unchanged structured query Q, the proposed pre- and post-edit visual facts, and the expected query result and surface reference answer. It jointly inspects the rendered candidates and selects a candidate only if the image realizes the proposed visual change, preserves scene identity and unrelated evidence, keeps the unchanged question answerable, and supports the proposed result. If no rendered candidate satisfies these conditions, the proposal is rejected.

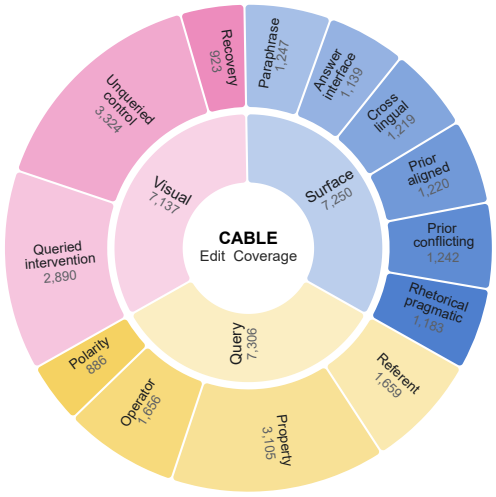
The validation criterion is specialized to the visual-edit family. For a queried intervention, the rendered change must affect evidence used by the current query. For an unqueried control, the rendered change must remain outside the queried evidence. For an ordinary recovery edit, the rendered im-

Source	#	Source	#
COCO	327	CC12M	102
TextVQA	173	LAION	69
Flickr2K	135	FoundIR	57
DIV2K	128	RealBlur	49
		Total	1,040

(a) Image-source composition

Property	Nodes	Property	Nodes
Blur	12,874	Low Contrast	1,001
Noise	5,170	Saturation Shift	422
Compression	3,680	Color Cast	1,665
Resolution	3,479	Color Quant.	357
Underexposure	4,106	Restoration Art.	1,065
Overexposure	5,254	AIGC Artifacts	1,336

(b) Low-level property coverage



(c) Edit coverage

Figure 6: Composition and coverage of CABLE. **(a)** Distribution of the 1,040 anchor bundles across eight image sources. **(b)** Coverage of low-level visual properties across all 22,733 benchmark nodes. **(c)** Hierarchical coverage of surface, query, and visual edits and their fine-grained subtypes.

age must sufficiently weaken or reverse the relevant prior degradation. The selected rendering is then materialized as the child visual state V_v, and the validated proposal provides the corresponding query result z_v and surface reference answer y_v. The transition is classified as invariant or altered by comparing z_v with the parent result z_u, rather than from the edit-family name.

Artifact-oriented extensions. Restoration and AI-rerender artifacts require realization procedures that differ from the ordinary deterministic image operators. We therefore add these cases after ordinary bundle construction while retaining the same query- and visual-edit formulation, rather than introducing a separate edit family. The extension considers two artifact-presence properties: visible restoration artifacts and visible AI-rerendering artifacts.

Artifact-oriented query edits are constrained property edits. They preserve the referents, expression structure, polarity, fact count, and fact order while replacing exactly one queried property with an artifact-presence property. The resulting question and surface reference answer are validated against the current image, so artifact presence is determined from visible evidence rather than inferred from how the question is phrased.

Artifact-oriented visual edits preserve the current question T and structured query Q while applying a full-image transformation. Image-restoration edits are realized with AdaIR (Cui et al., 2025) or AutoDIR (Jiang et al., 2024). For AI rerendering, we use SDXL Base 1.0 (Podell et al., 2024) and FLUX.1-dev in image-to-image mode without a semantic text prompt, reducing bias introduced by prompt design and keeping the transformation primarily conditioned on the input image. Because either transformation may alter query-relevant evidence in ways that cannot be prescribed in advance, the construction model independently answers the unchanged question from the transformed image. The resulting child query result is then compared with the parent result to determine whether the transition is invariant or altered.

Candidate validation. Candidate generation and validation are kept separate throughout construction. Each proposal is first materialized according to its edit family and then subjected to the corresponding family-specific validation described above. Invalid, ambiguous, or unrealized proposals are discarded. Only validated variants enter the candidate pool for recursive bundle expansion.

A.3 Diversity-Aware Bundle Expansion

Candidate pool formation. Bundle expansion proceeds layer by layer. At depth d, all retained nodes at that depth form the current frontier and serve as parents for the next expansion step. For

each parent, we consider the applicable surface, query, and visual edit families, with each parent-family pair contributing at most one proposal.

The validated candidates produced from the scheduled parent-family actions at the current frontier are merged into a single bundle-level candidate pool. The diversity-aware selector then retains at most six candidates from this pool, which form the frontier for the next depth. At depth zero, the anchor is the sole parent; if fewer than six valid candidates exist at a later depth, all are retained.

Diversity-aware scoring. For each candidate c, we assign a diversity reward

$$H(c \mid \mathcal{B}) = \frac{1}{1 + N_{\text{content}}(c; \mathcal{B})} + \frac{1}{1 + N_{\text{transition}}(c; \mathcal{B})},$$

where both counts are computed within the current bundle. N_{content} counts previously retained nodes with the same content signature as c. For surface edits, the signature is the surface strategy. For query edits, it is the atomic query family, namely referent, property, expression, or polarity. For visual edits, it is the combination of visual role and operator.

$N_{\text{transition}}$ counts previously retained edges with the same parent-modality to child-modality transition. The parent modality is anchor, surface, query, or visual, while the child modality is surface, query, or visual. For example, surface-to-query and visual-to-visual transitions have distinct signatures. Both terms have equal weight, so candidates associated with less frequent content or transition signatures receive larger diversity rewards.

Greedy beam selection. The diversity reward is accumulated along each generation path. For a candidate c with parent p, its beam score is

$$S(c) = S(p) + H(c \mid \mathcal{B}),$$

with the anchor initialized at $S = 0$. Candidates therefore inherit the cumulative score of their parent while receiving an additional reward for the current edit.

Selection is performed greedily within each bundle. At each step, we select the highest-scoring remaining candidate and add it to the retained bundle. The content and transition counts are then updated, and the scores of the remaining candidates are recomputed before the next selection. This process continues until six children have been retained or the candidate pool is exhausted. The selected children form the frontier for the next expansion depth.

A.4 Benchmark Statistics

CABLE contains 1,040 anchor bundles and 22,733 benchmark nodes. Figure 6 summarizes the benchmark composition across image sources, low-level visual properties, and controlled edit families. The resulting dataset provides broad coverage along all three dimensions.

A.5 Human Validation of Transition Labels

Transition relations and child references are produced by the automatic pipeline in Appendix A.2, so labeling errors would directly bias RA and RP. Visual transitions are the most exposed case: an edge labeled altered whose rendered change does not in fact change the queried result would record a correct answer as response inertia, which is precisely the direction of the query-visual adaptation gap in Section 3.3. We therefore had three experts with research experience in low-level vision audit a stratified sample of labels; one is an author and the other two were not otherwise involved.

The audit covers 412 scored, same-space transitions drawn by stratified random sampling over edit family and transition relation, with at most one transition per bundle (Table 7). Because mislabeled edges would concentrate where models almost uniformly retain the parent result, we additionally oversampled a consensus-RI stratum of visual altered transitions with at least 10 eligible models and an RI rate of at least 85%. To stay within the experts' language proficiency, we restrict the audit to English transitions, excluding any transition whose generation path contains a cross-lingual rewrite; the Frame column of Table 7 counts the remaining scored transitions per stratum.

For each transition, the experts saw the parent and child images, questions, and reference answers, with a synchronized overlay for comparing the two images, and were blind to the sampling stratum

Table 7: **Human validation of transition labels.** Error rates (%) with Wilson 95% intervals.

Stratum	Frame	n	Error
Visual, altered (random)	2,040	100	2.0 [0.6, 7.0]
Visual, altered (consensus RI)	350	70	2.9 [0.8, 9.8]
Population-weighted	2,390	170	2.1 [$\leq$7.4]
Visual, invariant	4,262	40	0.0 [0.0, 8.8]
Query, altered	2,311	80	1.2 [0.2, 6.7]
Query, invariant	2,556	50	0.0 [0.0, 7.1]
Surface	5,576	72	0.0 [0.0, 5.1]

and to all model responses. They judged whether each reference is correct, and additionally whether a visual change is perceptible, whether a query edit modifies only the designated component, and whether a surface rewrite preserves the queried fact and presents any added prior as an attributed cue. Parent references were not re-judged for anchor parents, which had been reviewed during anchor construction. A transition counts as an error if a displayed reference is judged incorrect or the edit ill-formed. A subset of 90 transitions was labeled independently by two experts, and such a transition is counted as an error if either expert flags it.

Label errors are rare across all strata. Reweighted to the sampling frame, the error rate of visual altered labels is 2.1%; weighting the Wilson upper bounds of the two visual altered strata by their frames gives a conservative 95% upper bound of 7.4%. Query labels are similarly reliable, with 1 error in 130 transitions. The error rate in the consensus-RI stratum (2.9%) is close to that of the random visual stratum (2.0%) and far below the RI rates of at least 85% that define the stratum; near-unanimous inertia on these transitions therefore reflects model behavior rather than mislabeled edits. The remaining errors concentrate on comparative or multi-condition questions and on visual edits whose change is subtle.

This upper bound is well below the error rate that would be needed to explain the query-visual gap. If every mislabeled visual transition were a correct response scored as a failure, the corrected visual RA would be at most $\mathrm{RA}_v/(1-e)$ for a label error rate e, so closing the gap requires $e \geq 1-\mathrm{RA}_v/\mathrm{RA}_q$. Among the 28 models whose visual RA is below their query RA, this threshold is smallest for LLaVA-OV-7B at 20.4%, so none of the 28 gaps can be attributed to label noise. Query labels are similarly reliable, so the gap does not reflect asymmetric label quality across edit families.

The two experts agreed on 96.7% of doubly labeled transitions, with a Gwet's AC1 (Gwet, 2008) of 0.96; Cohen's κ is lower (0.39) because nearly all labels are valid, a known behavior of κ at extreme prevalence (Feinstein & Cicchetti, 1990). The experts differed in strictness, and the conclusion holds under the strictest one alone: using only that expert's judgments raises the conservative upper bound to 16.9%, which still lies below the threshold for all 28 models.

B Model and Inference Details

B.1 Models and Execution Backends

We evaluate the models listed in Table 1 using the execution backends summarized in Table 8. Proprietary models are accessed through their corresponding provider APIs, while the Qwen3.5 series and DeepSeek-V4-FV-Exp are evaluated through hosted model APIs. The remaining open-weight models are deployed locally on a workstation equipped with two NVIDIA RTX 4090 GPUs with 24 GB memory each. We use vLLM (Kwon et al., 2023) for locally deployed models whenever supported; the others use their released inference implementations.

B.2 Unified Inference Protocol

Each CABLE node is evaluated independently in a zero-shot setting. The model receives only the image associated with the current node and its surface-form question, together with answer options when specified by the benchmark. No bundle structure, parent response, edit metadata, normalized query, visual state, or reference answer is exposed to the evaluated model.

Table 8: Model execution backends and inference settings.

Backend	Models	Inference
Provider API	GPT-5.6 Sol/Luna/Terra; Gemini 3.7 Flash, Gemini 3.1 Pro; Claude Opus 5, Sonnet 5, Haiku 4.5	Provider-hosted
Hosted API	DeepSeek-V4-Flash-Vision-Exp; Qwen3.5-397B-A17B, 122B-A10B, 35B-A3B	Hosted inference
Local vLLM	Gemma 4 12B IT; Molmo2-8B; InternVL3.5-8B; LLaVA-OneVision-7B/0.5B; Q-SiT/Q-SiT-Mini; Qwen2.5-VL-7B/3B; Q-Insight; Robust-R1-SFT/RL; LLaVA-v1.5-7B	2× RTX 4090, Temperature = 0
Local native	mPLUG-Owl2-7B; Q-Instruct-mPLUG, Q-Instruct-LLaVA; DepictQA-v2	2× RTX 4090, released code

To reduce backend-specific variation in output formatting, we use a lightweight common response format. Each model is instructed to answer the provided question based on the provided image and to return a single JSON object. The requested schema follows the response mode of the benchmark item: yes/no, single choice, multiple selection, or short answer. For option-based items, the benchmark-provided answer options are included in the model input. The question itself and its answer interface are otherwise preserved.

We introduce no benchmark-specific image preprocessing. For vLLM-based evaluation, images are converted to RGB and passed directly to the model interface, with subsequent processing handled by the model's standard visual processor. Models evaluated through their released implementations similarly use their native visual processing pipelines.

For vLLM-based inference, we set the sampling temperature to zero. For other local or API-based backends, we match this setting as closely as permitted by the corresponding interface. When a decoding parameter is not exposed by an API, we retain the provider-defined behavior.

B.3 Response Parsing and Projection

Reference result contracts. Before projecting model responses, we construct an aligned semantic result space for each group of compatible nodes. Each group is associated with a result contract that specifies the result type, output structure, and semantic values used for comparison. When surface interfaces encode the same result in different forms, the contract also specifies the required mapping from option choices or Boolean answers to these semantic values. The benchmark reference result is aligned once to this contract to obtain $\bar{z}_i$, which serves as the reference value for scoring. Transition-level comparisons are restricted to nodes with compatible result spaces. Reference alignment is independent of evaluated model outputs and does not use image pixels or transition labels.

Deterministic response projection. We first apply a conservative local parser that extracts only explicit answers from the model response. When the parsed answer can be resolved unambiguously under the node's answer interface and result contract, it is projected deterministically into the corresponding semantic value. Choice responses are mapped from selected options to their contract-defined values, while yes/no responses are mapped through the Boolean contract when the surface answer does not directly represent a Boolean query result. Scalar and set-valued responses can also be resolved by direct matching against the contract vocabulary, with set outputs normalized to a duplicate-free ordered representation. Responses that cannot be resolved by these rules are passed to the LLM-assisted projection stage.

LLM-assisted projection. Responses that cannot be resolved deterministically are projected with GPT-5.4-mini. The projector receives the surface question, structured query, answer interface, and the result contract for the corresponding semantic space. It preferentially operates on the locally parsed answer and consults the raw model response only when parsing fails. Guided by the contract's result type, structure, and projection instruction, it extracts the answer actually expressed by the evaluated model and maps it to the required typed semantic value, including scalar, set, relation, or composite outputs when applicable.

Table 9: **Sensitivity of pairwise comparisons to shared eligibility.** Each row summarizes 406 model pairs. Rev. counts pairs whose Δ_{orig} and Δ_{shared} have strictly opposite signs, reported in total, for $|\Delta_{\text{orig}}| < 5$ pp, and for pairs whose original interval includes zero. Weakened counts pairs whose original interval excludes zero but whose shared interval includes zero, with or without reversal; Supp. rev. counts pairs whose intervals both exclude zero with opposite signs. Shift CI≠0 counts pairs whose shift interval excludes zero, and Toward larger reports the percentage of pairs in which the shared comparison moves in favor of the model with the larger eligible set (one Surface RP pair with equal set sizes is excluded). All intervals are pointwise bundle-bootstrap 95% intervals.

	Rev.					|Shift| (pp)			
Metric	All	<5 pp	CI∋0	Weakened	Supp. rev.	Median	P90	Shift CI≠0	Toward larger (%)
Surface RP	33	32	21	27	2	1.84	3.76	246	79
Query RA	16	14	11	18	0	2.38	5.43	217	65
Query RP	48	48	42	31	0	1.74	3.57	183	79
Visual RA	21	21	19	17	0	2.47	7.50	222	60
Visual RP	60	60	49	88	0	1.16	3.24	223	74

The projector is instructed to preserve incorrect, contrary, or incomplete answers rather than correcting them. It is not given the image, reference result, parent response, edit history, or expected transition relation. Accordingly, this stage performs answer extraction and semantic normalization, not visual question answering or correctness assessment.

Projection validity. Projected values are accepted only if they satisfy the result contract associated with the node. Scalar outputs must match the required type and value space, set-valued outputs must contain valid normalized items, and composite outputs must conform to the declared slot structure and ordering. Projections that satisfy these constraints are retained as the model result a_i. When the response cannot be mapped reliably to a contract-compatible value, it is marked as ambiguous or unparseable rather than assigned a semantic result. Only projections with a usable semantic value are included in subsequent node- and transition-level scoring.

B.4 Human Validation of Response Projections

LLM-assisted projection handles 25% of the scored responses across the 29 models, while the remaining responses are resolved by deterministic rules. Because an unfaithful projection would misclassify a response as correct or incorrect, the same three experts as in Appendix A.5 audited 200 LLM-assisted projections that enter scoring.

Of the 200 audited projections, 150 were drawn at random and 50 were drawn from child responses on altered transitions scored as response inertia, where a projection that wrongly maps the child response onto the parent result would create spurious inertia. The experts saw the question, the result contract, the model's response, and the projected value, and judged only whether the projection faithfully expresses the response, irrespective of whether the response itself is correct. A subset of 42 projections was labeled independently by two experts.

Only two of the 200 projections are unfaithful, one in the random stratum (0.7%) and one in the inertia stratum (2.0%). They are a doubly negated yes/no question whose answer was projected with inverted polarity and a composite answer in which one slot was left unspecified. Spurious inertia introduced by projection is therefore at most on the order of 2%, small relative to the RI rates in Table 1. The two experts agreed on 97.6% of the doubly labeled projections (Gwet's AC1 $= 0.97$).

C Additional Evaluation Analyses

C.1 Cross-Model Eligibility

Conditioning on a correct parent in equation 7 separates evidence tracking from initial task correctness, but it also makes the evaluated transition set model-dependent. Each model's RA and RP are computed over its own eligible set E_M^+, which contains only transitions whose parent the model

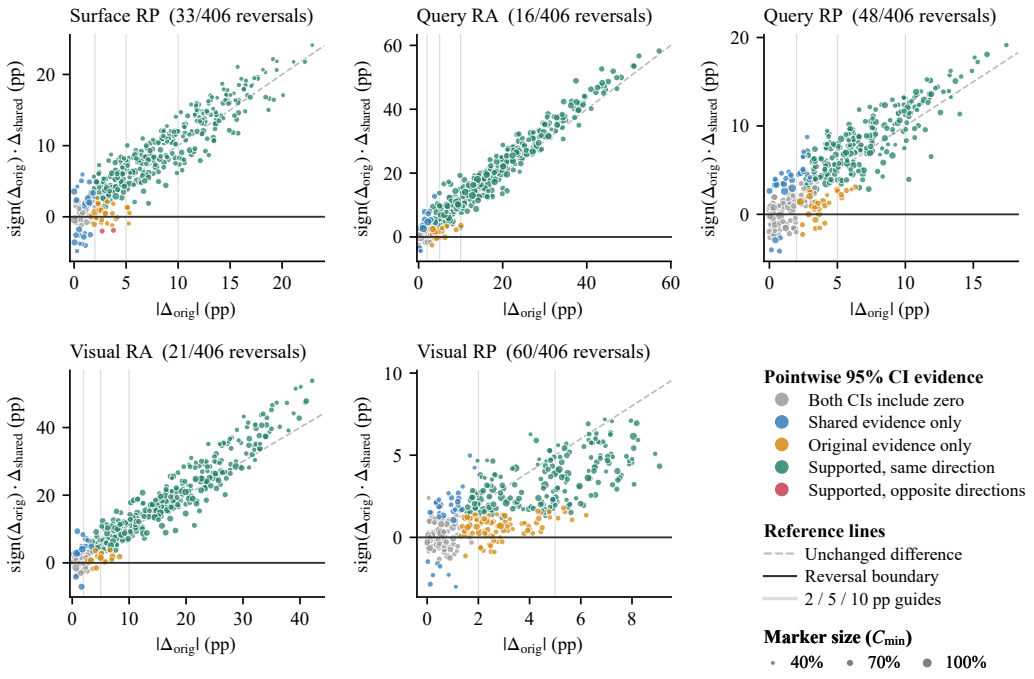


Figure 7: **Original versus shared-eligibility differences for all 406 model pairs.** The horizontal axis is the magnitude of the original difference, and the vertical axis is the shared difference oriented by the sign of the original one, so points below zero are point-estimate reversals and points on the dashed diagonal are unchanged. Colors indicate whether the original and shared pointwise 95% intervals exclude zero, and marker area increases with $C_{\min}$. Vertical guides mark 2, 5, and 10 pp.

answers correctly. A difference in RA or RP between two models may therefore reflect not only how they handle transitions, but also which transitions enter their denominators: a model that solves harder parents is also evaluated on the transitions that follow from them. We assess the sensitivity to this selection by recomputing every pairwise comparison on a common set of transitions.

For models A and B and a given edit family and transition relation (e.g., visual altered transitions), let $E^+_{AB} = E^+_A \cap E^+_B$ denote the transitions eligible for both models, i.e., both models answer the parent correctly and all endpoint projections are usable. Writing $S_M(E)$ for the RA or RP of model M over transition set E, computed as in equation 7, we compare

$$\Delta_{\text{orig}} = S_A(E^+_A) - S_B(E^+_B), \qquad \Delta_{\text{shared}} = S_A(E^+_{AB}) - S_B(E^+_{AB}), \tag{9}$$

and define the shift as $\Delta_{\text{orig}} - \Delta_{\text{shared}}$. We compute these quantities for all $\binom{29}{2} = 406$ model pairs and for the five family-level metrics in Table 1 (Surface RP, Query RA/RP, and Visual RA/RP), reusing the existing responses and projections without re-inference. Uncertainty is estimated with a 2,000-replicate cluster bootstrap over the 1,040 bundles: each replicate resamples bundles with replacement, retaining all of their transitions, and recomputes both differences from the same draw, thereby preserving within-bundle dependence. We report pointwise 95% percentile intervals without multiple-comparison correction. Because pairs share models and bundles, the counts below are descriptive rather than independent tests. The shared sets remain sizable: the smallest $|E^+_{AB}|$ across all pairs and metrics is 519, and the median of $C_{\min} = \min(|E^+_{AB}|/|E^+_A|, |E^+_{AB}|/|E^+_B|)$ ranges from 56% to 64% across metrics.

Table 9 and Figure 7 show that the direction of cross-model comparisons is largely insensitive to shared eligibility. Point estimates reverse in 16 and 21 of 406 pairs for Query RA and Visual RA, respectively, and in 33 to 60 pairs for the RP metrics. These reversals concentrate among pairs that were already close: across the five metrics, 175 of the 178 reversals occur at $|\Delta_{\text{orig}}| < 5$ pp, none occurs at $|\Delta_{\text{orig}}| \geq 10$ pp, and the three reversals at $|\Delta_{\text{orig}}| \geq 5$ pp all have shared intervals that include zero. For RA, no comparison is reversed with both intervals excluding zero. Reversals are also not explained by small intersections, as the median $C_{\min}$ is similar for reversed and non-reversed

pairs (62.0% vs. 64.6% for Query RA and 65.6% vs. 61.4% for Visual RA). The comparatively large number of weakened Visual RP comparisons reflects the narrow range of Visual RP differences, all of which are below 10 pp, so that modest shifts suffice to bring an interval across zero.

Although reversals are rare, selection does affect the magnitude of cross-model differences: the shift interval excludes zero in 183 to 246 of the 406 pairs per metric. This effect has a consistent direction. Restricting both models to jointly solved parents removes transitions that only the model with the larger eligible set can reach. Both models typically score higher on the intersection, but the model with the larger eligible set gains more (median within-model gains of 2.1 to 3.8 pp vs. 1.0 to 2.3 pp across metrics), so that the shared comparison moves in its favor in 60% to 79% of pairs. Among shifts whose intervals exclude zero, 139 to 216 per metric favor the model with the larger eligible set, compared with 12 to 78 against it. Parent conditioning is therefore conservative toward models with higher parent accuracy.

For RA, the model with the higher score usually also has the larger eligible set (287 and 305 of 406 pairs for Query and Visual RA), so this effect widens rather than reverses large differences. Among pairs with $|\Delta_{\text{orig}}| \geq 10$ pp, the shared difference is larger than the original one in 216 of 270 pairs for Query RA and 179 of 241 pairs for Visual RA; for example, the Visual RA difference between Gemini 3.1 Pro and LLaVA-v1.5-7B increases from 39.24 to 51.51 pp. When the model with the higher RP has the smaller eligible set, the same effect instead narrows the difference. The two supported reversals in Table 9 are of this type: GPT-5.6 Sol and Claude Opus 5 have lower Surface RP than Qwen3.5-397B-A17B under their own eligible sets (-2.69 and -3.78 pp), but higher Surface RP on the shared sets ($+2.01$ and $+1.90$ pp). In both cases, over 90% of the Qwen model's eligible set is retained in the intersection, compared with only 56% to 57% for the proprietary model. RP comparisons between models whose eligible sets differ substantially in size should therefore be interpreted with this selection effect in mind.

Shared eligible sets are biased toward parents that both models solve, and they differ from pair to pair. They therefore cannot replace the model-specific RA and RP in Table 1, nor can the pairwise results be combined into a single ranking. The shift quantifies the sensitivity of a comparison to the choice of transition set rather than providing a causal decomposition of difficulty. Overall, parent conditioning can change the ordering of closely matched models and the magnitude of their differences, but it produces no supported reversal of RA comparisons, and its dominant bias is conservative toward models that solve more parents.

C.2 Bundle-Level Aggregation

The metrics in Table 1 are pooled over transitions, so each eligible transition carries equal weight. Because the controlled variants of an anchor form a bundle, a bundle that contributes more eligible transitions also carries more weight in the pooled estimate. Under parent conditioning, this weight is model-dependent: a bundle contributes more transitions to a model's metrics when the model answers more of its parents correctly. We therefore recompute all metrics with equal weight per bundle to assess whether the main results depend on this choice of aggregation.

For a given model and metric, let n_b denote the number of eligible transitions in bundle b and h_b the number of those transitions that satisfy the metric, as defined in equation 7 and equation 8. The transition-level (TL) metric in Table 1 is $\sum_b h_b / \sum_b n_b$. The bundle-level (BL) metric first computes the per-bundle rate $r_b = h_b/n_b$ and then averages it over the set $\mathcal{B}^+$ of bundles with $n_b \geq 1$, i.e., $\frac{1}{|\mathcal{B}^+|} \sum_{b \in \mathcal{B}^+} r_b$. All-node accuracy is aggregated in the same way over nodes with usable projections, whereas anchor accuracy is unaffected because each bundle contains a single anchor. Eligibility, projections, and correctness are identical to those in the main evaluation. Because the TL metric is the n_b-weighted mean of r_b, the two aggregations are related by

$$\text{TL} - \text{BL} = \frac{\text{Cov}_{b \in \mathcal{B}^+}(n_b,\, r_b)}{\text{Mean}_{b \in \mathcal{B}^+}(n_b)}, \tag{10}$$

so they differ only when bundles that contribute more eligible transitions systematically achieve higher or lower rates. We estimate the uncertainty of $\Delta = \text{BL} - \text{TL}$ with the bundle bootstrap of Appendix C.1, recomputing both aggregations from each draw.

Table 10 compares the two aggregations. Bundle-level aggregation lowers RA and RP by 1.4 to 3.7 pp on average and raises RI by a comparable amount, whereas MU and all-node accuracy change

Table 10: **Agreement between bundle-level and transition-level metrics across 29 models.** $\Delta = \mathrm{BL} - \mathrm{TL}$ in percentage points. ρ and τ denote Spearman and Kendall rank correlations across models. Rev. counts the 406 model pairs whose BL and TL differences have strictly opposite signs, in total and among pairs with a TL difference of at least 5 pp.

	Δ (pp)			Rank corr.		Rev.	
Metric	Mean	Mean $\lvert\cdot\rvert$	Max $\lvert\cdot\rvert$	ρ	τ	All	$\geq$5 pp
All acc.	-0.24	0.32	2.18	0.995	0.975	5	0
Surface RP	-3.69	3.69	5.41	0.984	0.916	17	0
Query RA	-3.34	3.47	6.66	0.991	0.936	13	0
Query RI	$+2.65$	2.85	5.74	0.990	0.941	12	1
Query MU	$+0.69$	0.70	1.55	0.979	0.901	20	0
Query RP	-2.26	2.26	3.66	0.979	0.887	23	0
Visual RA	-2.59	2.78	5.09	0.973	0.897	21	1
Visual RI	$+2.68$	2.85	5.44	0.985	0.926	15	1
Visual MU	-0.09	0.31	0.80	0.991	0.936	13	0
Visual RP	-1.40	1.40	3.01	0.973	0.892	22	0

by less than 1 pp. Despite this shift in absolute level, the ordering of models is nearly unchanged, with $\rho \geq 0.973$ and $\tau \geq 0.887$ for every metric. Across the ten metrics, 143 of the 161 pairwise reversals occur between models whose TL difference is below 2 pp, only three occur at a TL difference of 5 pp or more, and none occurs at 10 pp or more.

The direction of the shift follows from equation 10. For RA and RP, the correlation between n_b and r_b is positive for 27 to 29 of the 29 models (median $+0.12$ to $+0.25$). Bundles in which a model answers more parents correctly therefore both contribute more eligible transitions and yield higher child success, so pooling assigns more weight to bundles that the model already handles well. Conversely, bundles in which the model solves only a single parent enter the pooled estimate through one transition but receive full weight under bundle-level aggregation; restricting the average to bundles with $n_b \geq 2$ roughly halves the shift. The paired bootstrap interval of Δ excludes zero for 22 to 29 models on each RA and RP metric, indicating a systematic consequence of the weighting rather than sampling noise. As in Appendix C.1, this effect stems from parent conditioning; since it is shared across models, it shifts absolute levels but largely preserves cross-model comparisons.

All qualitative conclusions of the main text hold under bundle-level aggregation. Visual RA is lower than query RA for all 29 models, with means of 32.1% and 58.2%, respectively. Mean RI increases from 35.7% under query edits to 61.7% under visual edits and rises for 28 of 29 models, whereas MU remains nearly unchanged (6.1% vs. 6.2%). RP exceeds RA and RI exceeds MU for every model under both query and visual edits. For the specialization analysis in Table 2, gains remain concentrated on polarity ($+18.27$ pp, 7/7 pairs), referent ($+6.64$ pp, 7/7), and cross-lingual rewrites ($+5.96$ pp, 6/7), and remain absent for queried interventions (-0.62 pp) and recovery (-1.69 pp, 2/7). Table 11 reports the complete bundle-level counterpart of Table 1.

Overall, weighting bundles rather than transitions lowers the absolute levels of RA and RP by a few points, because pooling favors bundles on which a model answers more parents correctly. This largely model-independent shift changes neither the model ordering nor the main conclusions.

C.3 Per-Pair Effects of Low-Level Specialization

Table 2 summarizes the seven matched base-specialist comparisons by their macro-average and the number of pairs that improve. A macro-average can be dominated by a few pairs, and the count of improved pairs reflects neither the magnitude nor the uncertainty of each difference. We therefore report each comparison separately to test whether the pattern in Section 3.5 holds across pairs.

For each pair and each column of Table 2, we compute the difference Specialist $-$ Base using the same transition-level metrics, so that the Mean row of Table 12 equals the Δ row of Table 2. Uncertainty is estimated with the 2,000-replicate bundle bootstrap of Appendix C.1, recomputing both models of a pair from the same draw. To test the central claim of Section 3.5 directly, we define

Table 11: **Bundle-level results on CABLE (%).** Counterpart of Table 1 in which each metric is averaged over bundles with equal weight rather than pooled over transitions. Anchor accuracy is identical to that in Table 1 because each bundle contains a single anchor. Comparison with Table 1 is summarized in Table 10.

	Node accuracy		Surface	Query				Visual			
Model	All↑	Anchor↑	RP↑	RA↑	RI↓	MU↓	RP↑	RA↑	RI↓	MU↓	RP↑
Proprietary General-Purpose Models											
Gemini 3.7 Flash	75.98	78.10	86.92	86.17	9.49	4.34	88.62	49.52	43.25	7.23	93.97
GPT-5.6 Sol	75.48	79.52	83.07	71.01	24.17	4.83	84.31	54.17	36.83	9.00	88.46
Gemini 3.1 Pro	73.45	75.12	84.70	81.40	13.44	5.16	85.19	56.15	35.19	8.66	91.03
Claude Opus 5	71.66	76.69	82.23	81.81	12.72	5.47	85.37	45.82	45.20	8.98	89.91
GPT-5.6 Luna	70.59	74.66	81.93	64.54	29.27	6.19	82.32	47.83	44.90	7.27	88.11
GPT-5.6 Terra	70.53	74.73	78.41	66.27	26.67	7.06	81.00	48.22	42.39	9.39	84.94
Claude Sonnet 5	66.28	71.61	79.79	75.93	18.35	5.72	83.88	38.50	54.20	7.30	87.23
Claude Haiku 4.5	52.76	58.66	71.65	50.86	38.53	10.62	72.96	23.19	66.24	10.56	85.74
General-Purpose Open-Weight Models											
DeepSeek-V4-FV-Exp	68.72	70.32	82.18	76.60	17.71	5.69	83.85	41.05	51.86	7.09	86.60
Qwen3.5-397B-A17B	71.43	77.97	87.37	67.86	27.40	4.73	85.50	38.33	58.47	3.20	95.90
Qwen3.5-122B-A10B	69.14	74.66	84.09	71.13	24.92	3.95	87.39	38.23	58.95	2.83	94.10
Qwen3.5-35B-A3B	65.60	73.39	83.26	55.72	41.01	3.28	82.70	29.29	66.72	4.00	93.69
Gemma 4 12B IT	60.85	65.23	78.50	71.48	22.01	6.51	78.47	31.50	60.50	8.00	91.21
Molmo2-8B	60.10	62.31	76.28	65.99	27.94	6.07	75.72	37.51	53.88	8.61	93.83
InternVL3.5-8B	52.08	53.81	72.23	56.51	33.13	10.36	78.08	14.80	81.26	3.94	93.62
Low-Level-Vision Specialists and Their Bases											
LLaVA-OV-7B	55.51	59.70	73.56	52.85	43.05	4.10	83.62	41.97	52.38	5.65	93.79
Q-SiT	55.61	57.84	77.23	65.36	31.18	3.46	80.93	36.22	59.46	4.33	94.23
LLaVA-OV-0.5B	44.27	46.95	68.21	30.72	64.11	5.17	72.64	30.37	63.73	5.91	92.25
Q-SiT-Mini	47.70	45.78	69.38	42.19	53.80	4.01	79.38	24.86	69.88	5.26	94.73
Qwen2.5-VL-7B	52.79	53.05	72.79	50.75	40.00	9.25	76.64	20.69	73.03	6.27	92.98
Q-Insight	51.30	50.35	73.35	52.17	38.96	8.87	78.11	21.45	71.38	7.17	93.62
Qwen2.5-VL-3B	51.10	53.41	69.76	37.31	55.25	7.44	75.54	21.45	73.91	4.64	91.81
Robust-R1-SFT	49.22	48.91	70.12	47.91	44.78	7.31	76.41	20.75	74.62	4.63	91.92
Robust-R1-RL	49.81	49.78	70.89	48.02	43.63	8.35	74.75	22.11	74.33	3.56	92.49
mPLUG-Owl2-7B	43.10	38.48	63.14	44.40	45.88	9.72	73.02	16.62	77.91	5.47	85.97
Q-Instruct-mPLUG	48.93	44.74	68.27	49.60	45.34	5.05	79.30	19.45	74.41	6.14	88.01
LLaVA-v1.5-7B	44.93	41.82	66.60	34.97	58.82	6.21	79.64	21.20	74.39	4.41	93.81
Q-Instruct-LLaVA	47.93	44.63	73.24	47.06	48.85	4.09	80.95	24.02	72.41	3.57	95.58
DepictQA-v2	43.63	37.17	67.90	40.33	55.08	4.59	75.55	16.72	76.25	7.02	86.16

for each pair the contrast

$$\mathrm{Q} - \mathrm{V} = \frac{1}{3} \sum_{s \in \mathcal{S}_{\mathrm{Q}}} \Delta_s - \frac{1}{2} \sum_{s \in \mathcal{S}_{\mathrm{V}}} \Delta_s, \tag{11}$$

where Δ_s is the specialization gain in column s, $\mathcal{S}_{\mathrm{Q}}$ contains the referent, property, and polarity columns of Query RA, and $\mathcal{S}_{\mathrm{V}}$ contains the queried-intervention and recovery columns of Visual RA. A positive value indicates that specialization improves adaptation to changed queries more than adaptation to changed visual evidence.

As shown in the last column of Table 12, the query-side gain significantly exceeds the visual-side gain in five of the seven pairs, with Q−V between $+9.82$ and $+17.47$ pp, and in no pair is it significantly smaller. Computing the contrast from the family-level Query RA and Visual RA instead yields the same five pairs. The two remaining pairs depart from this pattern in different ways. For Q-Insight, no column of Table 2 changes significantly, so specialization has little measurable effect on its transition behavior. Q-Instruct-mPLUG instead improves broadly, including all-node accuracy ($+5.82$ pp) and both visual columns ($+3.71$ and $+9.79$ pp), without a significant query-side advantage; its base, mPLUG-Owl2-7B, has the lowest all-node accuracy among all 29 models.

Table 12: **Per-pair effects of low-level visual specialization (%).** Each entry is Specialist − Base in percentage points, computed as in Table 2; the Mean row equals its Δ row. Q−V is the mean gain over the three Query RA columns minus the mean gain over the two Visual RA columns. $^{\uparrow}/^{\downarrow}$ mark differences whose paired bundle-bootstrap 95% interval lies above/below zero, and the last row counts such pairs.

		Surface RP						Query RA			Visual RA		
Specialist (Base)	All acc.	Neutral para.	Rhetorical	Prior aligned	Prior conflict.	Cross-lingual	Answer interface	Referent	Property	Polarity	Queried interv.	Recovery	Q−V
Q-SiT (LLaVA-OV-7B)	+0.10	+1.43	+2.27	+1.91	$-7.56^{\downarrow}$	$+9.28^{\uparrow}$	$+8.36^{\uparrow}$	−1.77	$-4.22^{\downarrow}$	$+25.70^{\uparrow}$	$-5.15^{\downarrow}$	−5.57	**+11.93**$^{\uparrow}$
Q-SiT-Mini (LLaVA-OV-0.5B)	$+3.42^{\uparrow}$	−1.75	−0.35	$+4.97^{\uparrow}$	+4.00	−1.31	+3.27	+4.83	$+10.89^{\uparrow}$	$+18.67^{\uparrow}$	−2.51	−9.51	**+17.47**$^{\uparrow}$
Q-Insight (Qwen2.5-VL-7B)	$-1.52^{\downarrow}$	+0.51	−0.46	+1.84	+0.83	+1.76	−1.57	+3.23	+0.38	+1.40	+1.81	−4.69	+3.11
Robust-R1-SFT (Qwen2.5-VL-3B)	$-2.04^{\downarrow}$	+0.73	+1.39	$-5.43^{\downarrow}$	+1.16	+4.71	−2.09	$+10.49^{\uparrow}$	+0.93	$+18.35^{\uparrow}$	−0.35	−3.96	**+12.08**$^{\uparrow}$
Robust-R1-RL (Qwen2.5-VL-3B)	$-1.45^{\downarrow}$	−0.27	+2.21	$-5.21^{\downarrow}$	+1.17	+4.39	−2.08	$+10.26^{\uparrow}$	+0.74	$+19.20^{\uparrow}$	+1.28	−3.55	**+11.20**$^{\uparrow}$
Q-Instruct-mPLUG (mPLUG-Owl2-7B)	$+5.82^{\uparrow}$	+0.36	$+7.77^{\uparrow}$	$+7.16^{\uparrow}$	+2.90	$+6.09^{\uparrow}$	$+16.94^{\uparrow}$	+0.50	−4.39	$+8.48^{\uparrow}$	$+3.71^{\uparrow}$	$+9.79^{\uparrow}$	−5.22
Q-Instruct-LLaVA (LLaVA-v1.5-7B)	$+2.94^{\uparrow}$	+2.28	$+4.58^{\uparrow}$	$+6.20^{\uparrow}$	+1.04	$+19.83^{\uparrow}$	$+9.03^{\uparrow}$	$+12.21^{\uparrow}$	−1.26	$+31.04^{\uparrow}$	+3.04	+5.32	**+9.82**$^{\uparrow}$
Mean	+1.04	+0.47	+2.49	+1.63	+0.51	+6.39	+4.55	+5.68	+0.44	+17.55	+0.26	−1.74	+8.63
Median	+0.10	+0.51	+2.21	+1.91	+1.16	+4.71	+3.27	+4.83	+0.38	+18.67	+1.28	−3.96	+11.20
$^{\uparrow}/^{\downarrow}$ pairs	3/3	0/0	2/0	3/2	0/1	3/0	3/0	3/0	1/1	6/0	1/1	1/0	5/0

Across metrics, polarity is the most uniform gain: it is positive for all seven pairs and significant for six, with a median of $+18.67$ pp. Referent and cross-lingual gains are positive for six of seven pairs, but their magnitudes are concentrated in a subset of pairs: only three pairs are significant in each column, and the medians ($+4.83$ and $+4.71$ pp) fall below the corresponding means ($+5.68$ and $+6.39$ pp). On the visual side, the near-zero averages do not reflect uniformly null effects. Family-level Visual RA decreases significantly for Q-SiT (-5.41 pp) and increases significantly for both Q-Instruct models, and recovery has a median of -3.96 pp with only two of seven pairs improving. Consistent with Section 3.5, all-node accuracy shows no consistent direction, increasing significantly for three pairs and decreasing significantly for three, with a median of $+0.10$ pp.

These results are robust to the choice of pairs and transition sets. Robust-R1-SFT and Robust-R1-RL share the same base model, and averaging them into a single comparison leaves six base-model families and does not change the aggregate pattern: family-level Query RA improves by $+8.14$ pp, Visual RA changes by -0.02 pp, and polarity improves by $+17.34$ pp. Removing any single pair keeps the mean polarity gain at or above 15.30 pp, the mean Query RA gain between 7.33 and 9.48 pp, and the mean Visual RA change within ± 0.9 pp. On shared eligible sets (Appendix C.1), the Query RA gain remains significantly positive for six of seven pairs; the exception is again Q-Instruct-mPLUG, whose gain drops from $+2.70$ to -1.59 pp, with an interval including zero.

Overall, the imbalance reported in Table 2 reflects a pattern shared by most pairs rather than an average driven by individual comparisons. Still, specialization effects vary substantially across pairs, and seven comparisons are too few to attribute them to particular training approaches.

D Additional Related Work

Reliance on Visual Evidence. Strong benchmark performance does not necessarily imply that MLLM responses are grounded in visual evidence. At the evaluation level, models may exploit linguistic biases, memorized knowledge, or answer layouts rather than image evidence (Lin et al., 2024; Chen et al., 2024b), motivating benchmarks such as HallusionBench that use contrastive control groups to disentangle visual errors from language-induced hallucinations (Guan et al., 2024). At the model level, internal probing shows that visual concepts may be correctly encoded yet suppressed by language priors during generation (Wang et al., 2025a; Li et al., 2026b). This issue is particularly relevant to low-level vision, where MLLMs may respond to image content while overlooking distortion evidence, partly due to feature degradation during vision-language alignment (Li et al., 2025d; Chen et al., 2026). Together, these studies show that apparent task competence or internal visual encoding does not necessarily imply that generated responses follow the relevant visual evidence. CABLE complements these analyses by evaluating such reliance behaviorally through controlled response transitions.