EVIDENCEFLOW: INFERENCE-TIME SUPPRESSION OF CONFOUNDER EVIDENCE IN VISION–LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

A vision–language model often affirms an object that is absent from an image when a related object is visible. Such errors are usually attributed to weak visual grounding, yet the visible content behind them is accurate and genuinely used: the failure is one of attribution rather than amount, so supplying more or less visual influence does not address it. We make this failure measurable per example. A visible cue is *confounder evidence* for an object query when removing it lowers the model's *Yes* margin more than every area-matched control removal, and the error it sustains is a *context-supported false affirmation*. In every evaluated model, removing co-occurring cues fixed in advance lowers the *Yes* margin more than the controls do on average, and restoration and blocking show that this support is concentrated in a narrow layer window at the prediction position. EvidenceFlow acts in that window at inference time: when the model answers *Yes*, it subtracts the positive component of the prediction state along a calibrated confounder direction, removing the cue's support for the queried claim while keeping the visual content. The direction and window come from 12–17 image pairs per model, one strength serves all models, and deployment uses only the original image and question with frozen weights. On POPEv2, EvidenceFlow improves InternVL3-8B accuracy by 8.7 percentage points, correcting 35.4% of false affirmatives while retaining 98.4% of correct ones, and accuracy improves in all nine model–benchmark cells. On Causal-HalBench with InternVL3-8B, replacing the calibrated direction with a random one at the same window and strength lowers correction from 46.2% to 1.1%.

1 INTRODUCTION

Object hallucination is commonly explained as a deficit of visual grounding: the model draws too little on what the image shows, so its answer follows language priors. Figure 1 shows an error that calls for a different account. The scene contains a cat and a keyboard but no laptop, and when asked whether a laptop is present, a vision–language model (VLM) answers *Yes*. Removing the keyboard, with the question fixed and the laptop still absent, flips the answer to *No*; removing an area-matched region elsewhere leaves the *Yes* in place. The visible content is used, but its evidential role is misassigned: the keyboard makes a laptop plausible, and the model treats that plausibility as support for the laptop's presence. The failure is one of attribution rather than amount, so making a model rely more on the image, or less, leaves unchanged what the keyboard is taken to be evidence for.

This view exposes a gap in both existing accounts and inference-time repairs. Co-occurrence and language-prior analyses identify which object claims are error-prone, as POPE's adversarial split does (Rohrbach et al., 2018; Li et al., 2023; Wang et al., 2024); they describe where errors are likely, but not what cue-derived support should be targeted. Existing remedies act on visual influence or its carriers. Methods that strengthen visual grounding treat hallucination as an imbalance between image evidence and language priors and increase or reweight visual influence at inference (Leng et al., 2024; Liu et al., 2025c;b;a); in the error of Figure 1, the misleading support already comes from image content the model uses, so amplifying visual influence can reinforce it. Subtractive methods move in the right direction but act on the carrier of the evidence, penalizing over-trusted

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

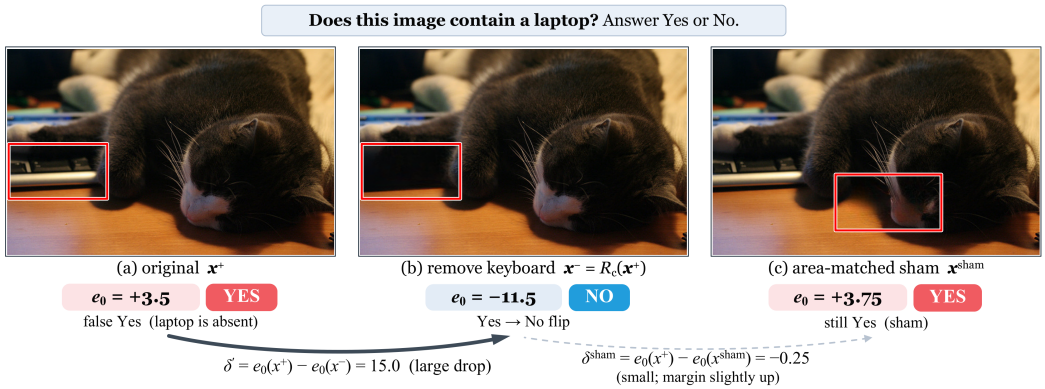


Figure 1: **A measured context-supported false affirmation.** The queried object (*laptop*) is absent in all three panels and the question is fixed; only the image changes. Removing the candidate confounder cue (keyboard) withdraws the affirmation, $\delta^* = 15.0$, whereas an area-matched sham leaves it standing, $\delta^{\text{sham}} = -0.25$. The contrast separates the cue from the act of editing. Section 4.2 reports the same comparison over the full diagnostic set.

attention patterns or removing image tokens judged hallucinatory (Huang et al., 2024; Che et al., 2025), and in these cases that carrier holds accurate visual content. The needed target is the cue's support for the queried claim, separated from the cue's visual content.

We make this target measurable. A visible cue is *confounder evidence* for an object query when it is not the queried object and it increases the model's support for answering *Yes*; when the queried object is absent and removing such a cue withdraws the *Yes*, we call the error a *context-supported false affirmation*. The definition is relative to the query and is tested on the model itself: we remove the cue and compare the drop in the *Yes* margin with that produced by area-matched control edits (Section 3.1). The analysis thus moves from co-occurrence counts to the model's measured use of a particular cue as support for the queried object's presence.

EvidenceFlow turns this measurement into an inference-time repair of a frozen model. Offline, a few image pairs with and without the cue serve two purposes: the average change they induce in the model's internal state defines a confounder direction, and restoration and blocking on the same runs locate a narrow prediction-position layer window where the cue's support reaches the answer. At test time the model first answers normally, and a *No* is returned unchanged, since object hallucination is an affirmative error. On a *Yes*, a second pass subtracts, within that window, a fixed fraction of the state's excess along the confounder direction, measured from the average cue-removed state; a state with no excess passes through unchanged. The edit removes support and keeps content: the image, the weights, and every other direction of the state are untouched. Calibration takes 12–17 image pairs per model, and deployment needs only the original image and question.

The paired runs also show that this support is concentrated: holding the selected window, three to four layers wide, at its cue-removed state withholds 60.5–93.5% of the restored query-side support across models. Because baseline *No* answers, whether correct rejections or missed detections, are never edited, accuracy can fall only when a correct *Yes* is overturned, so we report correction and retention together. On POPEv2, EvidenceFlow raises InternVL3-8B accuracy by 8.7 percentage points, correcting 35.4% of false affirmatives while retaining 98.4% of correct ones. Accuracy improves in all nine model–benchmark cells, and pooled over these cells, false affirmatives are flipped about 11 times as often as true ones. On Causal-HalBench with InternVL3-8B, the calibrated direction corrects 46.2% of false affirmatives, against 1.1% for a unit-norm random direction at the same window and strength.

Our contributions are as follows:

- **A failure mode and its measurement.** We isolate context-supported false affirmation, an error of attribution rather than amount, and define confounder evidence, a claim-relative

property of a visible cue measured per example against area-matched controls (Sections 3.1 and 4.2).

- **A repair that removes support rather than content.** EvidenceFlow subtracts one calibrated component of the prediction state inside a causally located window, using 12–17 image pairs per model, one shared strength, and neither training nor a test-time counterfactual (Section 3).
- **Direction-specific evidence.** Accuracy improves in all nine model–benchmark cells, false affirmatives flip about 11 times as often as true ones, and a random direction at the same window and strength corrects 1.1% of false affirmatives against 46.2% for the calibrated one (Section 4).

2 Related Work

Object hallucination and contextual bias. Benchmarks characterize object hallucination through model outputs: CHAIR measures it in generated captions (Rohrbach et al., 2018), POPE casts it as object-existence questioning and ties false claims to object frequency and co-occurrence (Li et al., 2023), and AMBER extends evaluation to attributes and relations in generative and discriminative settings (Wang et al., 2024). These measurements describe how often errors occur and in which settings. Counterfactual benchmarks intervene on the queried object itself: they remove or mask it to test whether absence is recognized (He et al., 2025; Li et al., 2026), or alter its attributes to conflict with knowledge priors, as in the Visual CounterFact benchmark of Pixels Versus Priors, which also steers between the two sources (Golovanevsky et al., 2025). We intervene on the context, holding the target absent and removing a co-occurring cue, which tests what an incorrect *Yes* rests on.

Where visual evidence reaches the decision. Mechanistic studies ask whether and where visual information reaches the answer. Information-flow analysis follows it from image features through question tokens to the prediction position (Zhang et al., 2025), Attention Lens locates hallucination-related object signals in middle-layer heads (Jiang et al., 2025), and TRACE identifies model-specific relay windows (Ye et al., 2026). A second line separates encoded visual evidence from the route that decides the answer: attributes can stay encoded while the answer follows a conflicting prior (Nooralahzadeh et al., 2026), circuit analyses distinguish grounding from hallucination pathways (Liu et al., 2026), and CRG suppresses prior-dominant textual routes (Cheng et al., 2026). In these accounts, a correct visual signal is available but does not control the answer. In a context-supported false affirmation, the visual signal does control the answer; the fault lies in the claim it is used to support. Localizing that support means tracing one cue's contribution to one answer, for which we adopt the restoration-and-blocking convention of causal tracing and activation patching (Meng et al., 2022; Wang et al., 2022; Zhang & Nanda, 2024).

Inference-time mitigation. Most inference-time mitigation methods either supply more visual influence (Leng et al., 2024; Liu et al., 2025c;b;a) or subtract attention patterns and image tokens (Huang et al., 2024; Che et al., 2025). Both change the strength of visual influence, either globally or by suppressing selected visual carriers, rather than the cue's support for the queried claim. The error studied here lies in that support: accurate evidence is taken to support the wrong claim.

Directional edits in representation space. Directional edits identify a direction from a contrast and add or remove it at inference. The contrast usually varies the behavior or concept to be controlled: truthful and untruthful activations (Li et al., 2024), paired prompts or examples that differ in a target behavior (Panickssery et al., 2024; Turner et al., 2024), harmful and harmless requests for refusal-direction ablation (Arditi et al., 2024), or concept labels for linear erasure (Ravfogel et al., 2020; Belrose et al., 2025). RepE develops such contrasts into a general framework for representation reading and control (Zou et al., 2025). Closest to our setting, REVIS decouples visual information from language priors by orthogonal projection at a fitted depth (Wu et al., 2026), Nullu projects MLP weights into the null space of a HalluSpace estimated from truthful and hallucinated prompts (Yang et al., 2025), and concurrent work removes a recurrent prior-aligned direction (Gong et al., 2026). EvidenceFlow shares the removal form and differs first in what its contrast varies: paired image counterfactuals hold the question and the target's absence fixed and change one visible

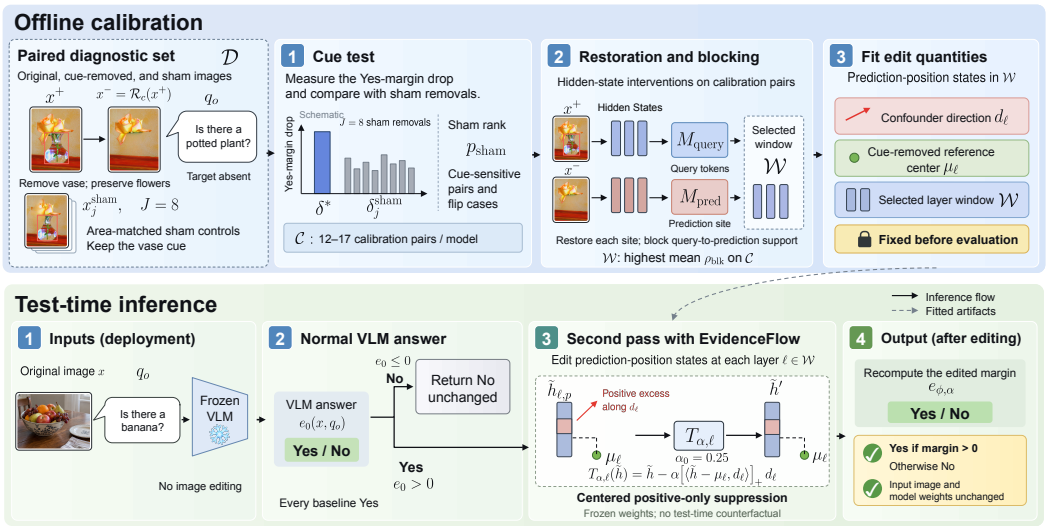


Figure 2: **EvidenceFlow pipeline.** Offline, cue-present and cue-removed calibration pairs are used to test cue-specific support against sham removals, locate the prediction-position window by restoration and blocking, and fit the frozen edit quantities $(d_\ell, \mu_\ell, \mathcal{W})$. At deployment, the model first answers normally; a *No* is returned unchanged, while a baseline *Yes* triggers a second pass that subtracts the positive excess along d_ℓ within $\mathcal{W}$. The edit uses only the original image and question, with frozen weights and no test-time counterfactual.

cue, so the direction isolates that cue's support for the queried claim. Its edit site is fixed by restoration and blocking, and the edit is conditional: it runs only on baseline affirmatives and removes a fraction of the state's excess over a cue-removed reference center.

3 Formulation and Method

The method has three steps: establish whether a visible cue supplies confounder evidence for the query, locate where that cue's support reaches the decision, and remove it there at inference time. Figure 2 summarizes the pipeline. The first two steps run once, offline, on paired images. For each model, a diagnostic set $\mathcal{D}$ holds 153 baseline false affirmatives, each paired with a cue-removed image and area-matched control edits, and Step 1 is computed on every pair in $\mathcal{D}$. A calibration set $\mathcal{C} \subset \mathcal{D}$ of 12–17 pairs, fixed before evaluation, determines the action: Step 2 locates the edit window on $\mathcal{C}$, and Step 3 fits the edit direction on $\mathcal{C}$. Both sets are disjoint by image from every evaluation benchmark. The calibration clean images come from COCO2017, with no image overlap with the test datasets. Only Step 3 touches a test input, every edit is to a hidden state, and the model's weights are frozen throughout.

Let $\mathcal{M}$ be a frozen vision–language model, x an image, and q_o an object-existence query for category o. We write $e_0(x, q_o) \in \mathbb{R}$ for the model's *Yes* margin at the first answer position: the logit of the affirmative answer token minus that of the negative one. The baseline answer is $\widehat{y}_0(x, q_o) = \mathbf{1}\{e_0(x, q_o) > 0\}$, so ties are decoded as *No*. The task label is $y(x, o) \in \{0, 1\}$, where $y = 1$ means that the queried object is present. Within the baseline affirmative population $\{\widehat{y}_0 = 1\}$, a false affirmative has $y = 0$ and a true affirmative has $y = 1$. EvidenceFlow is evaluated on the full distribution and acts only within $\{\widehat{y}_0 = 1\}$, so correction on false affirmatives and retention on true affirmatives determine the change in population accuracy.

3.1 The Method in Three Steps

Step 1: identifying confounder evidence. Fix a candidate confounder cue $c \neq o$ before any intervention, drawn from annotation-backed or pre-specified object lists. Let $\mathcal{R}_c$ be an image edit that removes c from a scene while leaving the query and the presence or absence of o unchanged,

and write x^+ for the original image and $x^- = \mathcal{R}_c(x^+)$ for its cue-removed counterpart. To calibrate what editing costs in itself, we draw $J = 8$ sham edits $\mathcal{R}_{\text{sham}_1}, \ldots, \mathcal{R}_{\text{sham}_J}$, each an area-matched control removal (Appendix C), and set $x_j^{\text{sham}} = \mathcal{R}_{\text{sham}_j}(x^+)$. The image-level margin effects are

$$\delta^*(x^+, o, c) = e_0(x^+, q_o) - e_0(x^-, q_o), \qquad \delta_j^{\text{sham}}(x^+, o) = e_0(x^+, q_o) - e_0(x_j^{\text{sham}}, q_o),$$

and the sham draws give both a size and a reference distribution for the cue's effect,

$$\bar{\delta}^{\text{sham}} = \frac{1}{J}\sum_{j=1}^{J} \delta_j^{\text{sham}}, \qquad \delta^{\text{adj}} = \delta^* - \bar{\delta}^{\text{sham}}, \qquad p_{\text{sham}} = \frac{1 + \left|\{j : \delta_j^{\text{sham}} \geq \delta^*\}\right|}{J+1}.$$

Because any edit can change the model's margin, we compare the cue removal with the sham removals. The adjusted effect δ^{adj} is the cue-specific drop in the *Yes* margin beyond the average sham drop, and p_{sham} tests whether the cue removal is larger than the area-matched controls (Appendix A).

These quantities define two objects. The first is a cue property. We call c confounder evidence for q_o at x^+ if removing it lowers the *Yes* margin more than every one of the J controls. With $J = 8$, this all-sham criterion gives $p_{\text{sham}} = 1/9$, the smallest attainable one-sided p-value. The second is an error property: the answer is a *context-supported false affirmation* sustained by c when the target is absent, the model answers *Yes*, and removing c changes that answer to *No*.

$$\underbrace{\delta^* > \max\big(0,\ \max_j \delta_j^{\text{sham}}\big)}_{c \text{ is confounder evidence for } q_o}, \qquad \underbrace{y(x^+, o) = 0, \quad e_0(x^+, q_o) > 0, \quad e_0(x^-, q_o) \leq 0}_{\text{context-supported false affirmation}}.$$

The first condition is graded: a cue can supply support without flipping the answer by itself. This makes cue status depend on the cue's own contribution, not on the original affirmative margin or on whether other cues also support the answer. The flip condition adds necessity for this particular answer, making it an error type rather than a cue property. Pairs meeting the first condition are the cue-sensitive pairs, and those meeting both are the flip cases. Section 4.2 reports these statistics over $\mathcal{D}$; in $\mathcal{C}$, 44 of the 45 pairs exceed all eight controls (Appendix C).

Step 2: locating where the support reaches the decision. Step 1 returns a cue whose removal costs the model affirmative margin. Step 2 locates where in the computation that support travels, working inside the two runs of Step 1 and replacing hidden states while the images stay fixed. Let $\mathcal{T}_q$ be the query-token positions and p the first answer position. For a window of contiguous layers $\mathcal{W}$, the tested sites are $M_{\text{query}} = \mathcal{W} \times \mathcal{T}_q$ and $M_{\text{pred}} = \mathcal{W} \times \{p\}$. The query site set M_{query} sits upstream on the question tokens, whose representation changes when the cue is removed, and the prediction site set M_{pred} is the readout at which the answer is produced. Two hidden-state interventions separate these roles, both starting from the cue-removed pass on x^-. In *restoration*, the states at a site set M are replaced by their values from the original pass on x^+, and the computation then continues, giving a first-answer margin $e^{\text{res}}(M)$. We apply restoration at both M_{query} and M_{pred}. In *blocking*, M_{query} is restored as before, while M_{pred} is held at its cue-removed values. This restores upstream support while freezing the readout site, and the resulting first-answer margin is e^{blk}. Because the query states feed the prediction position, restoration can recover the margin at both sites, and choosing where to edit calls for blocking.

One normalization scores all three margins: $r(e)$ is the fraction of the cue's margin gap that a run reproduces, equal to 0 at $e_0(x^-, q_o)$ and to 1 at $e_0(x^+, q_o)$,

$$r(e) = \frac{e - e_0(x^-, q_o)}{e_0(x^+, q_o) - e_0(x^-, q_o)}, \qquad \rho_{\text{res}}(M) = r\big(e^{\text{res}}(M)\big), \qquad \rho_{\text{blk}} = 1 - \frac{r\big(e^{\text{blk}}\big)}{\rho_{\text{res}}(M_{\text{query}})}.$$

Each is averaged over $\mathcal{C}$, and Figure 3 reports both recoveries and ρ_{blk}, the share of the query-side recovery that blocking removes. Recovery shows that the support is available at the restored site; a high ρ_{blk} shows that the restored support is mediated through M_{pred}, since only the residual fraction $1 - \rho_{\text{blk}}$ reaches the answer by other routes (Appendix A). Restoration is a sufficiency test and blocking a necessity test. An edit that removes support needs the second, because a site can carry the cue's signal while the answer is reached around it (Nooralahzadeh et al., 2026). We therefore select the window by blocking: among contiguous windows of at most four layers, $\mathcal{W}$ is the one with the highest average ρ_{blk} on $\mathcal{C}$, with ties broken toward fewer and then earlier layers; the selection uses only these paired runs (Appendix C).

Step 3: suppressing the support at inference time. Steps 1 and 2 fix the cue, the window $\mathcal{W}$, and the position p through which its support reaches the answer. Write $h_{\ell,p}(x, q_o)$ for the residual state at position p in layer ℓ; this is the state the method edits, after which the remaining layers run unchanged. Deployment sees only the original image, so every quantity that defines the action is fixed beforehand on $\mathcal{C}$. An ideal correction would remove the cue's own contribution to the prediction state, the displacement that erasing the cue would undo,

$$b_\ell(x, q_o, c) = h_{\ell,p}(x, q_o) - h_{\ell,p}\big(\mathcal{R}_c(x), q_o\big).$$

Subtracting b_ℓ would leave exactly the state the cue-removed image produces. Computing b_ℓ at deployment would require identifying c in the input and constructing $\mathcal{R}_c(x)$, so Step 3 uses the small calibration set $\mathcal{C}$ to estimate a reusable direction instead. Averaging b_ℓ over the pairs in $\mathcal{C}$ and normalizing gives the direction,

$$\overline{v}_\ell = \frac{1}{|\mathcal{C}|} \sum_{(x_i^+, x_i^-, q_i) \in \mathcal{C}} \left[h_{\ell,p}(x_i^+, q_i) - h_{\ell,p}(x_i^-, q_i) \right], \qquad d_\ell = \frac{\overline{v}_\ell}{\|\overline{v}_\ell\|_2}.$$

The unit vector d_ℓ is the calibrated confounder direction at layer ℓ. It averages the per-pair displacements whose restoration at M_{pred} recovered affirmative margin in Step 2, so suppressing along it is the counterpart of that restoration. The average runs over all pairs in $\mathcal{C}$, whether or not either run flipped. The reference center $\mu_\ell = |\mathcal{C}|^{-1} \sum_i h_{\ell,p}(x_i^-, q_i)$ is the mean prediction-position state of the cue-removed runs in $\mathcal{C}$, and a positive component along d_ℓ places a state on the cue-present side of the cue-removed center. With d_ℓ and μ_ℓ fixed, the estimate of b_ℓ at a state $\widetilde{h}_{\ell,p}$ reached at test time is

$$\widehat{b}_\ell(\widetilde{h}_{\ell,p}) = \left[\left\langle \widetilde{h}_{\ell,p} - \mu_\ell,\ d_\ell \right\rangle \right]_+ d_\ell, \qquad [a]_+ = \max(a, 0).$$

Only the scalar magnitude comes from the input, and the direction comes from $\mathcal{C}$: EvidenceFlow models the displacement as lying along one direction shared across inputs, which removes the need for a per-example counterfactual. The estimate is one-sided, so a state on the cue-removed side of μ_ℓ along d_ℓ is assigned no confounder contribution. The window, directions, and centers form the frozen action metadata $\phi = (\mathcal{W}, \{d_\ell, \mu_\ell\}_{\ell \in \mathcal{W}})$.

At inference the model first answers normally, and nothing is edited unless that answer is *Yes*. On a baseline affirmative, EvidenceFlow runs a second pass and edits the prediction-position state at every layer in $\mathcal{W}$ by

$$T_{\alpha,\ell}(\widetilde{h}_{\ell,p}) = \widetilde{h}_{\ell,p} - \alpha\, \widehat{b}_\ell(\widetilde{h}_{\ell,p}), \qquad 0 \le \alpha \le 1,$$

where $\widetilde{h}_{\ell,p}$ is the state reached in that pass just before the edit at layer ℓ, so later edits act on states that already reflect earlier ones. The strength α is the fraction of the estimated confounder contribution that is removed, making the action a bias correction with a shrinkage factor; $\alpha = 1$ would move the state to where the estimate places its cue-removed counterpart. One global value, $\alpha_0 = 0.25$, fixed on a development split disjoint from the main benchmarks, serves all models. Two features of this action protect correct affirmatives. It is confined to the single direction d_ℓ, so evidence for the object's actual presence survives unless it lies along that direction, and the truncation makes it one-sided, so a state with non-positive displacement is returned unchanged. Their effect is measured on the decoded answer after the forward pass completes.

Write $e_{\phi,\alpha}(x, q_o)$ for the first-answer margin after the intervention pass. EvidenceFlow edits every baseline affirmative, so its decision rule and the flip event observed on $\{\widehat{y}_0 = 1\}$ are

$$\widehat{y}_{\phi,\alpha}(x, q_o) = \begin{cases} \mathbf{1}\{e_{\phi,\alpha}(x, q_o) > 0\}, & e_0(x, q_o) > 0, \\ \widehat{y}_0(x, q_o), & \text{otherwise}, \end{cases} \qquad K_{\phi,\alpha}(x, q_o) = \mathbf{1}\{e_{\phi,\alpha}(x, q_o) \le 0\},$$

with K_α abbreviating $K_{\phi,\alpha}$ when ϕ is fixed. Baseline negatives are retained by construction, and since correction and retention take the baseline false and true affirmatives as denominators, the one event K_α determines both. Algorithm 1 in Appendix A summarizes the procedure.

4 Experiments

4.1 Experimental Setup

We evaluate InternVL3-8B (Zhu et al., 2025), DeepSeek-VL2-small (Wu et al., 2024), and Qwen3-VL-8B (Bai et al., 2025) with frozen weights on the first-token object-existence decision of Section 3: the *Yes–No* margin is computed from the logits of the first generated token, and every

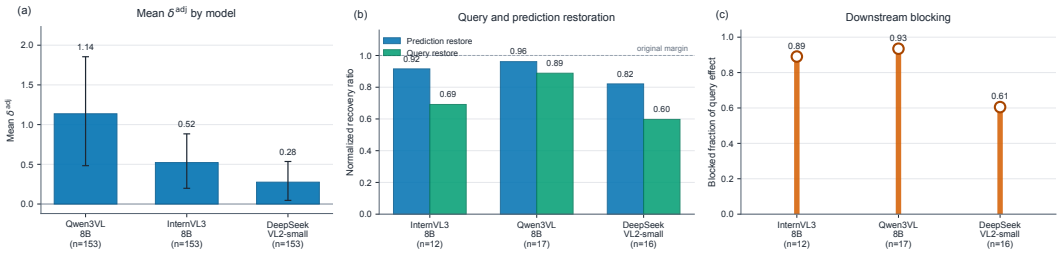


Figure 3: **Counterfactual path evidence on false affirmatives.** (a) On the diagnostic set $\mathcal{D}$, removing a candidate co-occurring cue lowers the *Yes* margin more than area-matched control removals; the sham-adjusted effect δ^{adj} is positive for every evaluated model (95% bootstrap interval excluding zero). (b) On the calibration pairs $\mathcal{C}$, restoring prediction-position states or query-side states recovers much of the lost affirmative margin. (c) At the selected window, clamping the prediction position to its cue-removed state withholds most of the query-restored support. Appendix C reports the numerical summaries and sample sizes behind each panel.

compared inference-time method is rerun under this protocol. The full-distribution evaluation uses POPEv2 (Li et al., 2026), RePOPE (Neuhaus & Hein, 2025), and Causal-HalBench (Xu et al., 2026), which instantiate object-existence hallucination under different construction principles. The full POPE split (Li et al., 2023) hosts the strength sweep, and the appendix adds further controls, baselines, and MME results (Fu et al., 2025).

EvidenceFlow applies the calibrated intervention to every baseline affirmative, with directions fitted and windows selected on the calibration set $\mathcal{C}$ of 12–17 pairs per model and one strength, $\alpha_0 = 0.25$, for all models and benchmarks (Section 3.1). Because α is a fraction of each state's own estimated confounder contribution, the same value carries over across models with different activation scales. At test time EvidenceFlow uses only the original image and question, at a cost of $1 + p_+$ forward passes per example for a baseline affirmative rate p_+ (Appendix A).

Correction and retention are scored on the baseline affirmative population: a false affirmative (FP) is corrected when the answer changes from *Yes* to *No*, and a true affirmative (TP) is retained when it remains *Yes*. With $g \in [0, 1]$ the probability that a baseline affirmative is edited ($g \equiv 1$ in the main evaluation, while a Gate sets g from a predeclared property of the example), the correction rate and the retention loss are

$$R = \mathbb{E}[g\, K_\alpha \mid \text{FP}], \qquad D = \mathbb{E}[g\, K_\alpha \mid \text{TP}].$$

Let $\pi = P(y = 0 \mid \widehat{y}_0 = 1)$ be the FP share among baseline affirmatives and $p_+ = P(\widehat{y}_0 = 1)$ the baseline affirmative rate. Because baseline negatives are never edited, the population accuracy change is

$$\Delta\text{Acc} = p_+ \big[\pi R - (1 - \pi) D\big],$$

so the intervention improves accuracy exactly when $\pi R > (1 - \pi) D$. When $\pi < 1/2$, the condition requires $R/D > (1 - \pi)/\pi > 1$: the action must flip false affirmatives more often than true ones, by a margin that grows as false affirmatives become rarer. Section 4.3 tests this requirement, and Appendix B analyzes how selection rules trade R against D.

4.2 Counterfactual Path Evidence

We first verify that the pre-specified co-occurring cues supply measurable affirmative support on false affirmatives. Step 1 of Section 3.1 is run on each model's diagnostic set $\mathcal{D}$, comparing cue removals with $J = 8$ area-matched controls. Figure 3a shows that removing the cue lowers the *Yes* margin more than the controls do: the sham-adjusted effect δ^{adj} ranges from 0.28 to 1.14 across models, each with a 95% bootstrap interval excluding zero. Per example, about a third of these false affirmatives pass the strict test of Section 3.1, with the cue effect exceeding all eight controls (34–35% per model), three times the rate of $1/9$ expected under the null. Thus part of the affirmative support behind these errors comes from co-occurring content.

We then localize where the cue-derived support reaches the answer. On the calibration pairs $\mathcal{C}$, Figure 3b,c shows that restoring prediction-position states recovers 82–96% of the lost margin and

restoring query-side states recovers 60–89% (96.1% and 88.8% for Qwen3-VL-8B), so both sites carry the support and restoration alone leaves the choice open. Blocking settles it: at the window selected in Step 2, clamping the prediction position to its cue-removed state while query-side restoration is active withholds most of the restored support in every model. A window of three to four layers (Appendix C) thus carries most of the support that reaches the answer, and an edit there intercepts it while leaving the question tokens and the image untouched. The window is fixed at this stage, before any accuracy is measured.

4.3 Full-Distribution Results

The full-distribution evaluation tests whether the localized intervention improves object-existence accuracy on the three benchmark distributions. Table 1 compares the main configuration with inference-time baselines. EvidenceFlow edits every baseline affirmative, so any separation between false and true affirmatives must come from the action itself.

Table 1: **Main full-distribution comparison.** Each method cell reports Acc/F1-Yes, with ΔAcc in parentheses. EvidenceFlow (EF) uses the frozen intervention with $\alpha_0 = 0.25$ and edits every baseline affirmative. EF corr./ret. reports false-affirmative correction and true-affirmative retention (%). PAI, VCD, and MFCD (Liu et al., 2025c; Leng et al., 2024; Liu et al., 2025a) are rerun under the same first-token Yes/No protocol; the appendix reports additional baselines and same-denominator POPE-full results.

Model	Benchmark	Baseline	EF	EF corr./ret.	PAI	VCD	MFCD
InternVL3-8B	POPEv2	72.8/78.5	**81.5/84.1** (+8.7)	35.4/98.4	73.0/78.6 (+0.2)	74.3/79.3 (+1.5)	78.7/82.0 (+5.9)
InternVL3-8B	RePOPE	89.3/88.8	**93.4/92.6** (+4.1)	51.6/97.8	89.5/89.0 (+0.2)	88.6/88.1 (-0.7)	89.3/88.6 (+0.0)
InternVL3-8B	Causal-HalBench	82.6/81.6	**86.2/84.0** (+3.6)	46.2/93.5	83.1/82.1 (+0.5)	82.6/81.8 (+0.1)	83.9/82.4 (+1.3)
DeepSeek-VL2-small	POPEv2	80.5/83.3	**82.9/84.9** (+2.4)	16.6/98.8	79.0/82.2 (-1.5)	81.6/83.8 (+1.1)	79.8/82.4 (-0.7)
DeepSeek-VL2-small	RePOPE	93.0/92.1	93.3/92.3 (+0.3)	20.8/98.4	92.5/91.7 (-0.5)	93.2/92.3 (+0.2)	**93.5/92.6** (+0.5)
DeepSeek-VL2-small	Causal-HalBench	88.9/87.4	**89.8/88.1** (+0.9)	26.2/97.7	88.3/87.0 (-0.5)	89.0/87.4 (+0.1)	89.0/87.3 (+0.1)
Qwen3-VL-8B	POPEv2	86.2/87.5	**91.2/91.5** (+5.0)	48.3/98.3	85.3/86.9 (-0.9)	85.5/86.9 (-0.7)	84.7/86.4 (-1.5)
Qwen3-VL-8B	RePOPE	93.5/92.8	**94.4/93.6** (+0.9)	45.6/96.9	92.7/92.0 (-0.8)	93.1/92.3 (-0.4)	92.9/92.1 (-0.6)
Qwen3-VL-8B	Causal-HalBench	86.6/84.9	**87.7/85.2** (+1.1)	38.8/94.4	85.7/84.2 (-0.9)	86.6/85.0 (+0.1)	85.8/84.3 (-0.7)

The baselines adjust how much visual influence reaches the answer, PAI through visual attention and VCD and MFCD through contrastive decoding. Their effect changes sign across models and benchmarks, and each lowers accuracy in at least three of the nine cells. This is consistent with Section 1: when the misleading content is already used, amplifying visual influence can strengthen the co-occurring cue along with the object evidence, and the net effect varies with the benchmark. EvidenceFlow, which removes only the support along the confounder direction, improves accuracy in all nine cells.

The source of that gain is the asymmetry required in Section 4.1. False affirmatives are a minority of the answers the main configuration edits in every cell, 10–35% in each cell and 17.7% overall, so an edit that flipped both groups at the same rate would lower accuracy everywhere. Accuracy rises in all nine cells because the action flips false affirmatives 7–29 times as often as true ones, 39.5% against 3.5% overall, or about 11 times (Table 12). The two features of Section 3.1 predict this asymmetry: answers held up by the confounder component lose their margin once it is removed, while answers supported by the object itself keep it, as expected if presence evidence lies largely off the confounder direction. Since baseline negatives are never edited, no true negative flips.

The operating point also tracks the mechanism measured in Section 4.2. DeepSeek-VL2-small, which shows the smallest mean cue effect and the weakest blocking in Figure 3, has the lowest correction and the highest retention of the three models: the action removes less where less confounder support is measured. The same strength serves all three models without retuning.

4.4 Direction and Strength Controls

We also verify that correction is specific to the calibrated confounder direction. Holding the edited set, window, centers, and strength fixed, Figure 4a shows that the calibrated direction corrects 598 of 1294 false affirmatives (46.2%), against 14 (1.1%) for one unit-norm random direction per edited

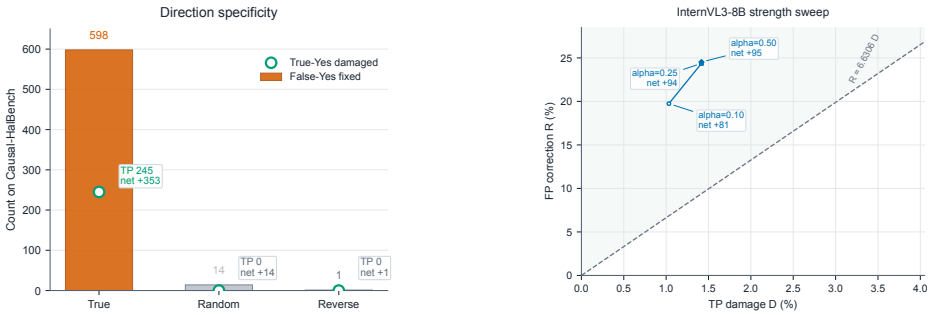


Figure 4: **Direction and strength controls on InternVL3-8B.** Left: Causal-HalBench controls (1294 false and 3761 true affirmatives) keep the edited set, window, centers, and strength fixed; the calibrated direction corrects 598 false affirmatives, a unit-norm random direction 14, and the reversed action 1. Right: strength sweep on the POPE-full adversarial-only Gate in the correction–retention plane; the dashed line is the net-positive boundary $\pi R = (1-\pi)D$ for this population, and $\alpha = 0.25$ is the strength of the main configuration. Appendix C reports all counts and denominators.

layer, and its net gain remains far larger once retention losses are counted ($+353$ versus $+14$). Reversing the sign of the edit corrects almost no false affirmatives (1 of 1294, or 0.1%). The correction is therefore specific to the calibrated direction, and the same advantage holds on selected POPE-full rows for all three models (Appendix C).

The strength sweep evaluates the operating point selected by the shared strength $\alpha_0 = 0.25$. Figure 4b sweeps α on a fixed population, the POPE-full adversarial-only Gate, which edits baseline affirmatives only on questions from the adversarial split and otherwise matches the main configuration. Raising α from 0.10 to 0.25 increases correction from 19.8% to 24.4% while retention loss moves from 1.0% to 1.4%, and raising it to 0.50 leaves the response nearly unchanged. All three points lie on the net-positive side of the boundary given by the accuracy identity of Section 4.1. The shared strength thus sits on a plateau of the response for this model, and Table 1 shows that the same value improves accuracy for all three.

5 Discussion and Conclusion

We isolate context-supported false affirmation, an error of attribution rather than amount in which accurate and genuinely used visual content supports the wrong claim. Confounder evidence makes this support testable per example, restoration and blocking locate the prediction-position window through which the cue's support reaches the answer, and EvidenceFlow removes that support there while keeping the content. One construct thus serves detection, localization, and correction, and the repair stays light: 12–17 image pairs per model, one shared strength, frozen weights, and only the original image and question at inference. Across all nine model–benchmark cells, the repair improves accuracy while flipping false affirmatives about 11 times as often as true ones, and the gain follows the calibrated direction: on Causal-HalBench with InternVL3-8B, a random direction at the same window and strength lowers correction from 46.2% to 1.1%. These results extend the observation that the decision point shapes how encoded visual evidence is used (Nooralahzadeh et al., 2026), from attribute conflicts to co-occurrence-driven existence errors and from diagnosis to per-example repair.

Binary object existence isolates the claim–evidence link most cleanly, and the construct applies to other hallucination types, each of which can be read as support for the wrong claim. For attribute and relation hallucinations, the counterfactual removes the cue that makes a false attribute or relation plausible while the queried objects and their true properties or relations stay in place; in free-form generation, the same idea suggests testing individual generated claims and editing the positions that produce them. These extensions share the question behind this work: how visible evidence is attributed to specific claims inside a multimodal decision.

AI USE STATEMENT

We used generative AI tools to assist with writing, editing, LaTeX formatting, consistency checks, and preparation of the manuscript. These tools were used to improve clarity, revise wording, inspect cross-references and numerical consistency, and help organize disclosure materials. We did not use generative AI tools to generate synthetic datasets, fabricate results, run experiments autonomously, or replace author verification of claims. All experimental results, tables, figures, definitions, and final claims were checked by the authors. We take responsibility for the final content of this work, including any text or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies object hallucination in vision–language models and proposes an inference-time mitigation method. The experiments use public datasets, public or standard model checkpoints, and object-existence evaluation protocols; no human-subject data, private user data, or personally identifying information is collected. The intended use of EvidenceFlow is diagnostic and corrective: it reduces a specific class of false affirmative object claims. A potential risk is that improved hallucination mitigation could be mistaken for a general safety guarantee. EvidenceFlow should therefore be viewed as a targeted reliability intervention rather than a complete solution for trustworthy multimodal deployment, especially in high-stakes settings where additional validation, monitoring, and human oversight remain necessary.

REPRODUCIBILITY STATEMENT

We describe the evaluation protocol, margin definition, correction and retention metrics, model choices, benchmarks, calibration sets, sham controls, intervention windows, and fixed strength in Sections 3 and 4. Appendix A gives the inference procedure and deployment cost, Appendix C reports the calibration metadata, sham-test summaries, mechanism-panel statistics, direction controls, and strength sweep, and Appendix D provides full flip accounting and complete method comparisons under the unified first-token *Yes*/*No* protocol. These details specify the data roles, fitted quantities, intervention sites, and evaluation denominators needed to reproduce the reported results. We will release the implementation and configuration files after documentation and cleanup, including the scripts needed to fit the calibration directions, apply the inference-time intervention, and reproduce the reported evaluations.

REFERENCES

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in Language Models Is Mediated by a Single Direction, October 2024. URL http://arxiv.org/abs/2406.11717. arXiv:2406.11717 [cs.LG].

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-VL Technical Report, November 2025. URL http://arxiv.org/abs/2511.21631. arXiv:2511.21631 [cs.CV].

Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. LEACE: Perfect linear concept erasure in closed form, April 2025. URL http://arxiv.org/abs/2306.03819. arXiv:2306.03819 [cs.LG].

Liwei Che, Tony Qingze Liu, Jing Jia, Weiyi Qin, Ruixiang Tang, and Vladimir Pavlovic. Hallucinatory Image Tokens: A Training-free EAZY Approach to Detecting and Mitigating Object Hallucinations in LVLMs. In *Proceedings of the IEEE/CVF International Conference on Computer*

Vision, pp. 21635–21644, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Che_Hallucinatory_Image_Tokens_A_Training-free_EAZY_Approach_to_Detecting_and_ICCV_2025_paper.html.

Zhe Cheng, Wenyu Chen, Fode Zhang, and Dehuan Shen. Mitigating Hallucinations in Large Vision-Language Models via Causal Route Gating, May 2026. URL http://arxiv.org/abs/2605.24024. arXiv:2605.24024 [cs.CV].

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, Rongrong Ji, Caifeng Shan, and Ran He. MME: A Comprehensive Evaluation Benchmark for Multimodal Large Language Models, October 2025. URL http://arxiv.org/abs/2306.13394. arXiv:2306.13394 [cs.CV].

Michal Golovanevsky, William Rudman, Michael A. Lepori, Amir Bar, Ritambhara Singh, and Carsten Eickhoff. Pixels Versus Priors: Controlling Knowledge Priors in Vision-Language Models through Visual Counterfacts. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 24837–24852. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.1262. URL https://aclanthology.org/2025.emnlp-main.1262/.

Weile Gong, Zijian Lu, Mingcai Chen, Yiping Zuo, Xin He, and Weibei Fan. Prior Directions: Why GUI Grounding Gets Locked in the Past, July 2026. URL http://arxiv.org/abs/2607.26913. arXiv:2607.26913 [cs.CV].

Yixiao He, Haifeng Sun, Pengfei Ren, Jingyu Wang, Huazheng Wang, Qi Qi, Zirui Zhuang, and Jing Wang. Evaluating and Mitigating Object Hallucination in Large Vision-Language Models: Can They Still See Removed Objects? In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 6841–6858. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.naacl-long.349. URL https://aclanthology.org/2025.naacl-long.349/.

Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. OPERA: Alleviating Hallucination in Multi-Modal Large Language Models via Over-Trust Penalty and Retrospection-Allocation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13418–13427, Seattle, WA, USA, June 2024. IEEE. ISBN 979-8-3503-5300-6. doi: 10.1109/CVPR52733.2024.01274. URL https://ieeexplore.ieee.org/document/10655465/.

Zhangqi Jiang, Junkai Chen, Beier Zhu, Tingjin Luo, Yankun Shen, and Xu Yang. Devils in Middle Layers of Large Vision-Language Models: Interpreting, Detecting and Mitigating Object Hallucinations via Attention Lens. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 25004–25014, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Jiang_Devils_in_Middle_Layers_of_Large_Vision-Language_Models_Interpreting_Detecting_CVPR_2025_paper.html.

E. L. Lehmann and Joseph P. Romano. *Testing statistical hypotheses*. Springer texts in statistics. Springer, New York, 3rd ed edition, 2005. ISBN 978-0-387-98864-1.

Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. Mitigating Object Hallucinations in Large Vision-Language Models through Visual Contrastive Decoding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13872–13882, Seattle, WA, USA, June 2024. IEEE. ISBN 979-8-3503-5300-6. doi: 10.1109/CVPR52733.2024.01316. URL https://ieeexplore.ieee.org/document/10657718/.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-Time Intervention: Eliciting Truthful Answers from a Language Model, June 2024. URL http://arxiv.org/abs/2306.03341. arXiv:2306.03341 [cs.LG].

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating Object Hallucination in Large Vision-Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 292–305, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.20. URL https://aclanthology.org/2023.emnlp-main.20.

Yifan Li, Kun Zhou, Xin Zhao, Lei Fang, and Jirong Wen. Analyzing and Mitigating Object Hallucination: A Training Bias Perspective. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(8):6636–6643, 2026. doi: 10.1609/aaai.v40i8.37594. URL https://ojs.aaai.org/index.php/AAAI/article/view/37594.

Bingqian Liu, Fu Zhang, Guoqing Chen, and Jingwei Cheng. Multi-Frequency Contrastive Decoding: Alleviating Hallucinations for Large Vision-Language Models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2025a. doi: 10.18653/v1/2025.emnlp-main.1452. URL https://aclanthology.org/2025.emnlp-main.1452/.

Jiaxin Liu, Ding Zhong, Yue Wang, Zhidong Yang, Zhaolu Kang, Guangyuan Dong, Qishi Zhan, Pengcheng Fang, and Aofan Liu. Dual-Pathway Circuits of Object Hallucination in Vision-Language Models, May 2026. URL http://arxiv.org/abs/2605.13156. arXiv:2605.13156 [cs.CV].

Sheng Liu, Haotian Ye, and James Zou. Reducing Hallucinations in Large Vision-Language Models via Latent Space Steering. In *International Conference on Learning Representations*, 2025b. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/b4008025c2182bfe16fcc8566ee14d64-Abstract-Conference.html.

Shi Liu, Kecheng Zheng, and Wei Chen. Paying More Attention to Image: A Training-Free Method for Alleviating Hallucination in LVLMs. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, volume 15141, pp. 125–140. Springer Nature Switzerland, Cham, 2025c. ISBN 978-3-031-73009-2 978-3-031-73010-8. doi: 10.1007/978-3-031-73010-8_8. URL https://link.springer.com/10.1007/978-3-031-73010-8_8. Series Title: Lecture Notes in Computer Science.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and Editing Factual Associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, 2022. doi: 10.52202/068431-1262. URL https://papers.nips.cc/paper_files/paper/2022/hash/6f1d43d5a82a37e89b0665b33bf3a182-Abstract-Conference.html.

Yannic Neuhaus and Matthias Hein. RePOPE: Impact of Annotation Errors on the POPE Benchmark, April 2025. URL http://arxiv.org/abs/2504.15707. arXiv:2504.15707 [cs.CV].

Farhad Nooralahzadeh, Omid Rohanian, Yi Zhang, Jonathan Fürst, and Kurt Stockinger. Arbitration Failure, Not Perceptual Blindness: How Vision-Language Models Resolve Visual-Linguistic Conflicts, April 2026. URL http://arxiv.org/abs/2604.09364. arXiv:2604.09364 [cs.CV].

Nina Panickssery, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering Llama 2 via Contrastive Activation Addition, July 2024. URL http://arxiv.org/abs/2312.06681. arXiv:2312.06681 [cs.CL].

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. Null It Out: Guarding Protected Attributes by Iterative Nullspace Projection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7237–7256, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.647. URL https://www.aclweb.org/anthology/2020.acl-main.647.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object Hallucination in Image Captioning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 4035–4045, Brussels, Belgium, 2018. Association

for Computational Linguistics. doi: 10.18653/v1/D18-1437. URL http://aclweb.org/anthology/D18-1437.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering Language Models With Activation Engineering, October 2024. URL http://arxiv.org/abs/2308.10248. arXiv:2308.10248 [cs.CL].

Junyang Wang, Yuhang Wang, Guohai Xu, Jing Zhang, Yukai Gu, Haitao Jia, Jiaqi Wang, Haiyang Xu, Ming Yan, Ji Zhang, and Jitao Sang. AMBER: An LLM-free Multi-dimensional Benchmark for MLLMs Hallucination Evaluation, February 2024. URL http://arxiv.org/abs/2311.07397. arXiv:2311.07397 [cs.CL].

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the Wild: a Circuit for Indirect Object Identification in GPT-2 small, November 2022. URL http://arxiv.org/abs/2211.00593. arXiv:2211.00593 [cs.LG].

Jialin Wu, Wei Shi, Han Shen, Peigui Qi, Kunsheng Tang, Zhicong Huang, Binghao Wang, and Zhou Yang. Revis: Sparse Latent Steering to Mitigate Object Hallucination in Large Vision-Language Models, May 2026. URL http://arxiv.org/abs/2602.11824. arXiv:2602.11824 [cs.AI].

Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang Ma, Chengyue Wu, Bingxuan Wang, Zhenda Xie, Yu Wu, Kai Hu, Jiawei Wang, Yaofeng Sun, Yukun Li, Yishi Piao, Kang Guan, Aixin Liu, Xin Xie, Yuxiang You, Kai Dong, Xingkai Yu, Haowei Zhang, Liang Zhao, Yisong Wang, and Chong Ruan. DeepSeek-VL2: Mixture-of-Experts Vision-Language Models for Advanced Multimodal Understanding, December 2024. URL http://arxiv.org/abs/2412.10302. arXiv:2412.10302 [cs.CV].

Zhe Xu, Zhicai Wang, Junkang Wu, Jinda Lu, and Xiang Wang. Causal-HalBench: Uncovering LVLMs Object Hallucinations Through Causal Intervention. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(40):34169–34177, 2026. doi: 10.1609/AAAI.V40I40.40712. URL https://doi.org/10.1609/AAAI.V40I40.40712.

Le Yang, Ziwei Zheng, Boxu Chen, Zhengyu Zhao, Chenhao Lin, and Chao Shen. Nullu: Mitigating Object Hallucinations in Large Vision-Language Models via HalluSpace Projection, March 2025. URL http://arxiv.org/abs/2412.13817. arXiv:2412.13817 [cs.CV].

Wencheng Ye, Yi Bin, Yujuan Ding, Hongye Fang, Zheng Wang, Xing Xu, Jingkuan Song, Yun Zhang, Sirui Da, and Heng Tao Shen. The Ebb and Flow of Multimodal Focus: Scheduling Visual Relay Windows for Grounded VLM Reasoning, July 2026. URL http://arxiv.org/abs/2607.11436. arXiv:2607.11436 [cs.AI].

Fred Zhang and Neel Nanda. Towards Best Practices of Activation Patching in Language Models: Metrics and Methods. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/06a52a54c8ee03cd86771136bc91eb1f-Abstract-Conference.html.

Zhi Zhang, Srishti Yadav, Fengze Han, and Ekaterina Shutova. Cross-modal Information Flow in Multimodal Large Language Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19781–19791, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Zhang_Cross-modal_Information_Flow_in_Multimodal_Large_Language_Models_CVPR_2025_paper.html.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, Zhangwei Gao, Erfei Cui, Xuehui Wang, Yue Cao, Yangzhou Liu, Xingguang Wei, Hongjie Zhang, Haomin Wang, Weiye Xu, Hao Li, Jiahao Wang, Nianchen Deng, Songze Li, Yinan He, Tan Jiang, Jiapeng Luo, Yi Wang, Conghui He, Botian Shi, Xingcheng Zhang, Wenqi Shao, Junjun He, Yingtong Xiong, Wenwen Qu, Peng Sun, Penglong Jiao, Han Lv, Lijun Wu, Kaipeng Zhang, Huipeng Deng, Jiaye Ge, Kai Chen, Limin Wang, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. InternVL3: Exploring Advanced Training and Test-Time Recipes for Open-Source Multimodal Models, April 2025. URL http://arxiv.org/abs/2504.10479. arXiv:2504.10479 [cs.CV].

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, J. Zico Kolter, and Dan Hendrycks. Representation Engineering: A Top-Down Approach to AI Transparency, March 2025. URL http://arxiv.org/abs/2310.01405. arXiv:2310.01405 [cs.LG].

A METHOD DETAILS

Throughout the appendix, EvidenceFlow denotes the main configuration, which edits every baseline affirmative. Gate and Select variants apply the same action to a subset of baseline affirmatives (Appendix B).

Validity of the sham test. Under the null that c is no more influential than an area-matched control removal, δ^* is exchangeable with the J sham values, so p_{sham} is a valid one-sided p-value. Its smallest value is $1/(J+1)$, reached when δ^* exceeds every sham value, which under the null happens with probability $1/(J+1)$; with $J = 8$ this is $1/9$. Taking $J = 1$ recovers the single-sham comparison $\delta^* > \delta_1^{\text{sham}}$.

Why the cue condition is graded. A cue can carry support without being the whole of it. Requiring a reversal would make cue status depend on quantities unrelated to c: the same cue contributing the same amount would count in an image with a small affirmative margin and fail to count in one with a large margin. A reversal requirement would also miss a scene holding two such cues, where removing either alone leaves the answer standing although together they produce it. The graded condition treats both cases consistently, and the flip condition serves separately as the definition of the error type.

Blocking as a mediation measurement. Blocking holds M_{pred} at its cue-removed values while query states in layers past $\mathcal{W}$ still evolve from restored values, so ρ_{blk} measures mediation through M_{pred}. A small residual $1 - \rho_{\text{blk}}$ identifies M_{pred} as the bottleneck through which the cue's support reaches the answer, and separates it from a site where the cue's signal is visible but unused by the answer.

Inference procedure. Algorithm 1 gives the inference procedure of EvidenceFlow for one query. Gate and Select variants insert a test on $g(S(x, q_o))$ before the intervention pass (line 3) and leave the action unchanged (Appendix B.1).

Algorithm 1 EvidenceFlow inference for one query

Require: Image x, object query q_o, frozen model $\mathcal{M}$, action metadata $\phi = (\mathcal{W}, \{d_\ell, \mu_\ell\}_{\ell \in \mathcal{W}})$, strength α

1: Run $\mathcal{M}$ on (x, q_o) and read the first-token *Yes* margin e_0.
2: If $e_0 \leq 0$, **return** NO.
3: Run an intervention pass; at each layer $\ell \in \mathcal{W}$, edit the current prediction state:
$\widetilde{h}_{\ell,p} \leftarrow \widetilde{h}_{\ell,p} - \alpha[\langle \widetilde{h}_{\ell,p} - \mu_\ell, d_\ell \rangle]_+ d_\ell$.
4: Complete the forward pass, compute $e_{\phi,\alpha}$, and **return** YES iff $e_{\phi,\alpha} > 0$ (otherwise NO).

Inference cost. EvidenceFlow runs one baseline forward pass for every example and one intervention pass for each baseline affirmative, so its cost is $1 + p_+$ forward passes per example, where p_+ is the baseline affirmative rate: for example, p_+ is 52.07% for InternVL3-8B on Causal-HalBench, 60.20% for Qwen3-VL-8B on POPEv2, and 46.54% for Qwen3-VL-8B on RePOPE. Gate and proxy variants run the intervention pass on smaller subsets; on POPE full, they edit 219 to 672 of 9000 examples in replay and 108 to 342 of 4500 held-out examples. Directions, windows, and the strength are fixed during calibration, so deployment adds a forward hook to a frozen model and requires no training.

B SELECTION RULES: THEORY AND EXPERIMENTS

EvidenceFlow applies its action to every baseline affirmative. This appendix characterizes, for the fixed action of Section 3.1, which baseline affirmatives should receive it under a retention budget and which correction–retention trade-offs are attainable, and then reports the selection experiments on POPE full (Appendices B.8 and B.9). Throughout, the action metadata $\phi = (\mathcal{W}, \{d_\ell, \mu_\ell\}_{\ell \in \mathcal{W}})$ and the strength α are fixed, and K_α is the flip event of Section 3.1; changing α changes K_α and therefore the attainable frontier.

B.1 SELECTION INTERFACE AND FRONTIER

A deployable selection rule reads only a statistic fixed before evaluation,

$$S : \mathcal{X} \times \mathcal{Q} \to \mathcal{S}, \qquad S(x, q_o) = \Psi(e_0(x, q_o), \{h_{\ell,p}(x, q_o)\}_{\ell \in \mathcal{W}}, m(x, q_o)) \,,$$

where $\mathcal{X}$, $\mathcal{Q}$, and $\mathcal{S}$ are the image, query, and statistic spaces. The extractor Ψ is fixed on calibration data, and $m(x, q_o)$ is empty for deployable policies, with predeclared metadata used only in controlled analyses. A policy $g : \mathcal{S} \to \{0, 1\}$ then decides which baseline affirmatives are edited, inserting a test on $g(S(x, q_o))$ before the intervention pass (line 3 of Algorithm 1). The main EvidenceFlow configuration is the case $g \equiv 1$; the Select and Gate variants change only S and g, never the action.

Let $\mathcal{G}(S)$ be the class of S-measurable randomized rules $g : \mathcal{S} \to [0, 1]$, where $g(S)$ is the probability of applying the fixed action to a baseline affirmative instance. Deterministic gates are the special case $g \in \{0, 1\}$. With $R(g)$ and $D(g)$ as defined in Section 4.1, the selection frontier for a retention-loss budget ε is

$$\beta_{\alpha,S}(\varepsilon) = \sup_{g \in \mathcal{G}(S)} R(g) \quad \text{subject to} \quad D(g) \le \varepsilon.$$

Proposition 1 (Flip-weighted selection frontier). Let

$$k_F(s) = \mathbb{E}[K_\alpha \mid S = s, \mathrm{FP}], \qquad k_T(s) = \mathbb{E}[K_\alpha \mid S = s, \mathrm{TP}],$$

and define finite measures

$$d\mu_F(s) = k_F(s)\, dP_{\mathrm{FP}}(s), \qquad d\mu_T(s) = k_T(s)\, dP_{\mathrm{TP}}(s).$$

Let μ_F^{ac} be the part of μ_F absolutely continuous with respect to μ_T, let N support the singular part of μ_F with respect to μ_T, and define the flip-weighted likelihood ratio by the Radon–Nikodym derivative

$$\Lambda_K(s) = \frac{d\mu_F^{\mathrm{ac}}}{d\mu_T}(s).$$

Then an optimal rule for $\beta_{\alpha,S}(\varepsilon)$ selects the zero-retention-loss singular region and thresholds Λ_K on the remaining support: for some threshold τ and possible boundary randomization γ,

$$g^\star(s) = \mathbf{1}_N(s) + \mathbf{1}\{\Lambda_K(s) > \tau\} + \gamma(s)\mathbf{1}\{\Lambda_K(s) = \tau\}.$$

Appendix B.4 reduces Λ_K to the product of $k_F(s)/k_T(s)$ and the density ratio $dP_{\mathrm{FP}}/dP_{\mathrm{TP}}(s)$ wherever that ratio exists. Optimal selection therefore depends on both FP/TP enrichment and the response of the fixed action. A score that separates FPs from TPs can still be suboptimal if the action rarely flips those selected FPs, while an action-response score can be suboptimal if the same response also occurs on TPs.

B.2 SELECTION AND ACTION-RESPONSE FACTORS

For deterministic gates used in the experiments, the trade-off separates into a selection factor and an action-response factor. Define

$$a_F = P(g = 1 \mid \mathrm{FP}), \qquad a_T = P(g = 1 \mid \mathrm{TP}),$$

and, when the corresponding selected populations are nonempty,

$$r_F = P(K_\alpha = 1 \mid g = 1, \mathrm{FP}), \qquad r_T = P(K_\alpha = 1 \mid g = 1, \mathrm{TP}).$$

Then $R = a_F r_F$ and $D = a_T r_T$, so when $D > 0$,

$$\frac{R}{D} = \frac{a_F}{a_T} \cdot \frac{r_F}{r_T}.$$

Selected-FP precision and firing likelihood ratio therefore measure the selection factor a_F/a_T, and net accuracy also depends on the action-response factor r_F/r_T.

Value of baseline-available information. For a fixed action and a fixed retention-loss budget, allowing the selection rule to use more baseline-available information cannot shrink the feasible set. If $\sigma(S_1) \subseteq \sigma(S_2)$, then $\mathcal{G}(S_1) \subseteq \mathcal{G}(S_2)$, so

$$\beta_{\alpha,S_2}(\varepsilon) \geq \beta_{\alpha,S_1}(\varepsilon).$$

The size of the improvement is determined by how the added information changes the flip-weighted likelihood ratio in Proposition 1.

B.3 Proof of Proposition 1

Fix the action metadata ϕ and strength α, and write K_α for $K_{\phi,\alpha}$. Use the statistic map $S : \mathcal{X} \times \mathcal{Q} \to \mathcal{S}$ from Appendix B.1; it generates the sub-σ-algebra available before intervention. Let

$$\mathcal{G}(S) = \{g : \mathcal{S} \to [0,1] \ : \ g(S) \text{ is } \sigma(S)\text{-measurable}\}.$$

The value $g(S)$ is the conditional probability of applying the fixed action to a baseline affirmative instance. FP and TP denote the two label groups within the baseline affirmative population.

For each value s, define

$$k_F(s) = \mathbb{E}[K_\alpha \mid S = s, \mathrm{FP}], \qquad k_T(s) = \mathbb{E}[K_\alpha \mid S = s, \mathrm{TP}].$$

By the tower property,

$$\begin{aligned} R(g) &= \mathbb{E}[g(S)K_\alpha \mid \mathrm{FP}] \\ &= \mathbb{E}[g(S)\mathbb{E}(K_\alpha \mid S, \mathrm{FP}) \mid \mathrm{FP}] \\ &= \int_{\mathcal{S}} g(s)k_F(s)\, dP_{\mathrm{FP}}(s), \end{aligned}$$

and similarly

$$D(g) = \int_{\mathcal{S}} g(s)k_T(s)\, dP_{\mathrm{TP}}(s).$$

Define two finite measures on $\mathcal{S}$:

$$d\mu_F(s) = k_F(s)\, dP_{\mathrm{FP}}(s), \qquad d\mu_T(s) = k_T(s)\, dP_{\mathrm{TP}}(s).$$

The frontier problem is therefore

$$\sup_{g \in \mathcal{G}(S)} \mu_F(g) \quad \text{subject to} \quad \mu_T(g) \leq \varepsilon.$$

Decompose μ_F with respect to μ_T as

$$\mu_F = \mu_F^{\mathrm{ac}} + \mu_F^{\perp}.$$

Let N support the singular part $\mu_F^{\perp}$, so that $\mu_T(N) = 0$ and $\mu_F^{\mathrm{ac}}(N) = 0$. Setting $g = 1$ on N adds $\mu_F(N)$ correction at zero retention-loss cost. On the absolutely continuous part, define

$$\Lambda_K(s) = \frac{d\mu_F^{\mathrm{ac}}}{d\mu_T}(s).$$

The Neyman–Pearson lemma for finite measures (Lehmann & Romano, 2005, Theorem 3.2.1) gives an optimal rule that thresholds this Radon–Nikodym derivative. Thus there exist $\tau \geq 0$ and a measurable boundary randomization $\gamma : \{\Lambda_K = \tau\} \to [0,1]$ such that

$$g^\star(s) = \mathbf{1}_N(s) + \mathbf{1}\{\Lambda_K(s) > \tau\} + \gamma(s)\mathbf{1}\{\Lambda_K(s) = \tau\}$$

maximizes $\mu_F(g)$ among all rules satisfying $\mu_T(g) \leq \varepsilon$. If the budget is large enough that the constraint is not binding, the same expression holds with $\tau = 0$ and full selection on all positive-density regions. This proves the proposition. □

B.4 Regular-form expression for Λ_K

On any region where P_{FP} is absolutely continuous with respect to P_{TP} and $k_T(s) > 0$,

$$d\mu_F(s) = k_F(s)\, dP_{\mathrm{FP}}(s), \qquad d\mu_T(s) = k_T(s)\, dP_{\mathrm{TP}}(s),$$

so the Radon–Nikodym derivative becomes

$$\Lambda_K(s) = \frac{d\mu_F}{d\mu_T}(s) = \frac{k_F(s)}{k_T(s)} \frac{dP_{\mathrm{FP}}}{dP_{\mathrm{TP}}}(s).$$

Regions with $k_T(s) = 0$ and $k_F(s) > 0$ belong to the singular zero-cost correction part: the fixed action flips FPs there but produces no co-located TP retention loss.

B.5 Low-budget selectivity

The finite-measure formulation also explains the low-budget regime. On the absolutely continuous support, any rule with $\mu_T(g) > 0$ satisfies

$$\frac{\mu_F(g)}{\mu_T(g)} \leq \operatorname*{ess\,sup}_s \Lambda_K(s).$$

Conversely, threshold sets on which Λ_K approaches this essential supremum approach the best attainable correction-to-retention-loss ratio, subject to measurability and mass at the boundary. Singular regions with $\mu_T = 0$ and $\mu_F > 0$ yield correction at zero retention-loss cost. This statement concerns the frontier for the fixed action and fixed strength α; changing α changes K_α, hence changes μ_F, μ_T, and the frontier.

B.6 Thresholding a scalar score

If $\Lambda_K(s)$ is monotone in a scalar score $T(s)$, then the optimal rule can be implemented as a threshold on T. Otherwise the optimal rule thresholds Λ_K itself, and a threshold on a chosen scalar score is an approximation whose quality depends on how well that score orders the flip-weighted likelihood ratio. Whether it holds is an empirical property of the action and the score, separate from FP/TP separability.

B.7 Gated margin shifts and constant-shift membership

The selection result in Appendix B.1 depends only on the observed flip event K_α for a fixed internal action. A separate gated margin-shift view applies to operators that can be written at the answer margin as $e' = e - c(S)$, with $c(S)$ determined before intervention. For such an operator, the same frontier analysis applies after replacing K_α by $\mathbf{1}\{0 < e \leq c(S)\}$ on the baseline affirmative population.

A representation-space edit belongs to a constant-shift subfamily only under an additional readout condition. Under a local linear readout $e(h) = u^\top h + b$, an edit $h' = h + \Delta(h)$ changes the margin by $u^\top \Delta(h)$. This shift is constant on a selected region only when $u^\top \Delta(h)$ is constant there. For a direction-suppression edit, the induced margin change generally depends on the projection coordinate $\langle h - \mu, d \rangle$ unless $u^\top d = 0$ or that coordinate is itself fixed. Membership in a constant-shift family is therefore a property to verify for each method.

B.8 Selection ladders

Table 2 traces how the choice of selector moves the correction–retention operating point of the fixed InternVL3-8B action on POPE full. Selected-FP precision describes selection enrichment, and correction and retention are the R and D of Section 4.1.

Table 2: Selection ladders for InternVL3-8B on POPE full. Replay rows are evaluated on all 9000 examples and Held-out rows on the 4500 held-out rows.

Source	Gate / selector	Actioned	FP corr.	TP damaged	TP loss	Selected-FP precision	ΔAcc	Net
Replay	observed-effect top75 reference	672	247	171	4.11%	58.93%	+0.84	+76
Replay	SAE top75, all strategies	505	214	123	2.95%	63.76%	+1.01	+91
Replay	SAE top75, popular+adversarial	383	168	82	1.97%	68.15%	+0.96	+86
Replay	SAE top75, adversarial-only	219	106	41	0.98%	72.15%	+0.72	+65
Held-out	observed-effect top75 reference	342	123	92	4.42%	54.68%	+0.69	+31
Held-out	frozen top75, popular+adversarial	185	82	43	2.07%	64.86%	+0.87	+39
Held-out	fixed top75, adversarial-only	108	50	28	1.34%	65.74%	+0.49	+22
Held-out	post-hoc top50, all strategies	161	79	38	1.83%	65.84%	+0.91	+41

B.9 Targeting with co-occurrence proxies

Changing which affirmatives receive the fixed action sets a deployment operating point, and targeting uses co-occurrence in the role Section 1 assigns to it: the co-occurrence rate decides which baseline affirmatives receive the action, and the calibrated direction determines what the action removes. Table 3 compares selectors for InternVL3-8B on POPE full; split-based rows use benchmark metadata as a controlled condition, and online rows use annotation-derived co-occurrence scores available before intervention. Table 4 pairs the same action with a metadata gate and with the online proxy across the three main benchmarks. The proxy improves accuracy in all nine model–benchmark cells and keeps 70–94% of the gain of the metadata gate.

Table 3: Proxy selectors for InternVL3-8B on POPE full. Split-based rows use benchmark metadata as a controlled condition; online rows replace it with annotation-derived co-occurrence scores.

Selector	ΔAcc	ΔF1-Yes	FP corr.	TP damaged	TP loss	Selected-FP precision	Net
Split popular/adversarial top75	+1.00	+0.70	88	43	2.07%	69.57%	+45
Split adversarial-only top75	+0.82	+0.64	56	19	0.91%	74.78%	+37
Online annotation proxy top75	+0.89	+0.52	100	60	2.88%	63.67%	+40
Online positive log-lift	+0.84	+0.41	113	75	3.60%	60.26%	+38
Online top50	+0.64	+0.38	73	44	2.02%	61.34%	+29
Online top25	+0.11	+0.00	26	21	1.01%	52.05%	+5
Online top10	-0.18	-0.16	10	18	0.86%	35.71%	-8

Table 4: Targeting across the main benchmarks with a metadata gate and with the online proxy. Cells report percentage-point changes in accuracy/F1-Yes.

Model	Benchmark	Metadata gate	Online proxy
InternVL3-8B	POPEv2	+9.50 / +6.45	+7.40 / +4.82
	RePOPE	+3.91 / +3.71	+3.16 / +3.07
	Causal-HalBench	+1.12 / +1.05	+1.05 / +0.93
DeepSeek-VL2-small	POPEv2	+3.00 / +2.18	+2.10 / +1.43
	RePOPE	+0.41 / +0.28	+0.29 / +0.21
	Causal-HalBench	+0.57 / +0.43	+0.48 / +0.36
Qwen3-VL-8B	POPEv2	+5.80 / +4.81	+4.60 / +3.71
	RePOPE	+0.78 / +0.66	+0.61 / +0.52
	Causal-HalBench	+0.74 / +0.42	+0.65 / +0.36

C Calibration, Mechanism, and Control Details

C.1 Calibration metadata and area-matched controls

Table 5 lists, for each model, the size of the calibration set $\mathcal{C}$ and the intervention window selected on it by blocking; both are fixed before the full-distribution evaluation. Each area-matched control removes a same-image non-target object of matching area where a suitable one exists, and otherwise relocates an equal-area region. The cue-support analysis uses $J = 8$ such controls per example. Table 6 summarizes the sham tests on the calibration set $\mathcal{C}$. Table 7 reports the per-example tests on the diagnostic set $\mathcal{D}$.

Table 5: Calibration pairs used to fit the confounder direction, and intervention windows fixed by blocking.

Model	$\lvert\mathcal{C}\rvert$	Intervention window
InternVL3-8B	12	layers 21–23
Qwen3-VL-8B	17	layers 26–29
DeepSeek-VL2-small	16	layers 14–16

Table 6: Sham tests on the calibration set $\mathcal{C}$ with $J = 8$ area-matched controls per pair. The sham drop is the mean margin change across the eight controls.

Model	Pairs	Mean δ^*	Mean sham drop	Mean δ^{adj}	Median δ^{adj}	$p_{\mathrm{sham}} = 1/9$	Target flips
InternVL3-8B	12	5.120	0.018	5.102	3.246	12/12	11/12
DeepSeek-VL2-small	16	3.482	0.009	3.474	2.707	15/16	15/16
Qwen3-VL-8B	17	9.059	0.022	9.037	7.812	17/17	15/17

Table 7: Per-example sham tests on each model's diagnostic set $\mathcal{D}$ ($n = 153$, $J = 8$). Under the null, the cue effect exceeds all eight controls with probability $1/9$.

Model	Mean δ^*	Mean sham drop	Mean δ^{adj} [95% CI]	$\delta^* >$ mean sham	$\delta^* >$ all shams
InternVL3-8B	0.611	0.089	0.522 [0.198, 0.883]	58.8%	35.3%
DeepSeek-VL2-small	0.320	0.045	0.275 [0.046, 0.536]	52.9%	34.0%
Qwen3-VL-8B	1.239	0.102	1.136 [0.483, 1.855]	52.9%	35.3%

C.2 Mechanism panels

Table 8 lists the numerical summaries behind the three panels of Figure 3, which identify cue-level support and the prediction-position window before the intervention is evaluated on complete distributions. Cue support is measured on the diagnostic set $\mathcal{D}$ (153 pairs per model), and restoration and blocking on the calibration set $\mathcal{C}$.

Table 8: Numerical summaries for the mechanism panels of Figure 3. Cue-support rows report the mean sham-adjusted effect $\delta^{\mathrm{adj}} = \delta^* - \bar{\delta}^{\mathrm{sham}}$ over $J = 8$ area-matched control removals per example (Section 3.1), with 95% bootstrap intervals in brackets. Restoration rows give the recovered fraction of the lost margin at the prediction position and at the query tokens; blocking rows give ρ_{blk}.

Panel	Model / diagnostic set	n	Main statistic	Control statistic
Cue support	Qwen3-VL-8B, $\mathcal{D}$	153	mean $\delta^{\mathrm{adj}} = 1.14$ [0.49, 1.86]	J=8, $\bar{\delta}^{\mathrm{sham}}$=0.10
Cue support	InternVL3-8B, $\mathcal{D}$	153	mean $\delta^{\mathrm{adj}} = 0.52$ [0.20, 0.89]	J=8, $\bar{\delta}^{\mathrm{sham}}$=0.09
Cue support	DeepSeek-VL2-small, $\mathcal{D}$	153	mean $\delta^{\mathrm{adj}} = 0.28$ [0.04, 0.53]	J=8, $\bar{\delta}^{\mathrm{sham}}$=0.04
Restoration	InternVL3-8B, $\mathcal{C}$	12	91.5% pred. / 69.1% query	cue-removed run
Restoration	Qwen3-VL-8B, $\mathcal{C}$	17	96.1% pred. / 88.8% query	cue-removed run
Restoration	DeepSeek-VL2-small, $\mathcal{C}$	16	82.1% pred. / 59.7% query	cue-removed run
Blocking	InternVL3-8B, $\mathcal{C}$	12	89.1% of restored support blocked	cue-removed pred. window
Blocking	Qwen3-VL-8B, $\mathcal{C}$	17	93.5% of restored support blocked	cue-removed pred. window
Blocking	DeepSeek-VL2-small, $\mathcal{C}$	16	60.5% of restored support blocked	cue-removed pred. window

C.3 Direction controls

Table 9 expands the direction controls of Section 4.4. The Causal-HalBench rows keep the EvidenceFlow edited set, window, centers, and strength fixed and change only the edit: the random control uses one unit-norm random direction per edited layer, and the reversed-action control adds the same positive calibrated component, reversing the sign of the edit. The POPE-full rows apply the same convention to fixed selected rows. Table 10 repeats the comparison on model-specific selected rows for all three models, and in every model the calibrated direction exceeds both controls in net corrections.

Table 9: Direction controls on InternVL3-8B. Actioned is the edited subset, while ΔAcc and ΔF1-Yes are computed on the full evaluation denominator: 9709 for Causal-HalBench and 4500 for the POPE-full held-out rows. Net is FP corrected minus TP damaged.

Population	Direction	Actioned	FP corr.	TP damaged	ΔAcc	ΔF1-Yes	Net
Causal-HalBench EvidenceFlow	true	5055	598	245	+3.64	+2.37	+353
Causal-HalBench EvidenceFlow	random	5055	14	0	+0.14	+0.12	+14
Causal-HalBench EvidenceFlow	reversed action	5055	1	0	+0.01	+0.01	+1
POPE full, popular+adversarial top75	true	185	82	43	+0.87	+0.58	+39
POPE full, popular+adversarial top75	random	185	4	7	-0.07	-0.09	-3
POPE full, popular+adversarial top75	reversed action	185	0	0	+0.00	+0.00	0
POPE full, adversarial-only top75	true	108	50	28	+0.49	+0.30	+22
POPE full, adversarial-only top75	reversed action	108	0	0	+0.00	+0.00	0

Table 10: Selected gates on POPE full for the three models. Direction differences compare the calibrated direction with the reversed-action and random controls on the same selected rows.

Model	Baseline Acc./F1	FPR	Actioned	FP corr.	TP damaged	ΔAcc	True–reverse net	True–random net
InternVL3-8B	89.29 / 89.63	13.96%	108	50	28	+0.49	+22	+27
DeepSeek-VL2-small	89.74 / 89.70	9.78%	5	5	0	+0.11	+5	+5
Qwen3-VL-8B	88.31 / 87.37	4.24%	26	15	11	+0.09	+4	+5

C.4 Strength sweep

Table 11 gives the InternVL3-8B adversarial-only Gate sweep shown in the right panel of Figure 4. The population has 628 baseline false affirmatives and 4164 baseline true affirmatives, so the R–D plot uses one net-positive boundary, $\pi R = (1-\pi)D$ with $\pi = 628/4792$. Correction rises from $\alpha = 0.10$ to $\alpha = 0.25$ and is nearly unchanged at $\alpha = 0.50$, and every point is net-positive.

Table 11: Adversarial-only Gate strength sweep on POPE full for InternVL3-8B. The fixed population has 628 baseline false affirmatives and 4164 baseline true affirmatives; R and D are percentages on those denominators, and ΔAcc uses the 9000-example population.

Gate	α	FP corr.	TP damaged	R	D	ΔAcc	Net
adversarial-only	0.10	124	43	19.75	1.03	+0.90	+81
adversarial-only	0.25	153	59	24.36	1.42	+1.04	+94
adversarial-only	0.50	154	59	24.52	1.42	+1.06	+95

D Evaluation Protocol and Full Results

D.1 Unified evaluation protocol

All appendix tables use the first-token *Yes*/*No* margin protocol of the main text. Baseline predictions are recomputed under this protocol, and every change is measured against the baseline on the same model–benchmark denominator. EvidenceFlow edits only baseline affirmatives and carries baseline negatives forward unchanged. FP corrected counts baseline false affirmatives that change to *No*, and TP retained counts baseline true affirmatives that remain *Yes*.

D.2 Full flip accounting

Table 12 expands Table 1 with the denominators and flip events behind each cell. MME existence and count subsets are included as auxiliary checks outside the three main benchmarks.

Table 12: Full flip accounting for EvidenceFlow. Accuracy and F1-Yes are reported as baseline → EvidenceFlow, with the accuracy change in parentheses. FP corr. reports corrected false affirmatives. TP ret. reports retained true affirmatives. TN flip reports baseline true negatives changed to *Yes*.

Model	Benchmark	n	Acc.	F1-Yes	FP corr.	TP ret. / TN flip
InternVL3-8B	POPEv2	1000	72.80 → 81.50 (+8.70)	78.48 → 84.07	95/268	488/496 ; 0/232
	RePOPE	8184	89.30 → 93.43 (+4.13)	88.78 → 92.65	414/803	3389/3465 ; 0/3843
	Causal-HalBench	9709	82.55 → 86.19 (+3.64)	81.62 → 83.98	598/1294	3516/3761 ; 0/4254
	MME existence	60	98.33 → 100.00 (+1.67)	98.36 → 100.00	1/1	30/30 ; 0/29
	MME count	60	85.00 → 83.33 (-1.67)	87.00 → 85.29	0/9	29/30 ; 0/21
DeepSeek-VL2-small	POPEv2	1000	80.50 → 82.90 (+2.40)	83.29 → 84.88	30/181	480/486 ; 0/319
	RePOPE	8185	92.97 → 93.30 (+0.33)	92.09 → 92.32	80/385	3296/3349 ; 0/4261
	Causal-HalBench	9709	88.87 → 89.80 (+0.94)	87.41 → 88.11	176/672	3667/3752 ; 0/4876
	MME existence	60	98.33 → 98.33 (+0.00)	98.33 → 98.33	0/0	29/29 ; 0/30
	MME count	60	81.67 → 83.33 (+1.67)	83.64 → 84.38	2/9	27/28 ; 0/21
Qwen3-VL-8B	POPEv2	1000	86.20 → 91.20 (+5.00)	87.48 → 91.51	58/120	474/482 ; 0/380
	RePOPE	8185	93.50 → 94.44 (+0.94)	92.76 → 93.55	183/401	3302/3408 ; 0/4245
	Causal-HalBench	9709	86.58 → 87.67 (+1.09)	84.88 → 85.22	310/798	3452/3656 ; 0/4750
	MME existence	60	100.00 → 96.67 (-3.33)	100.00 → 96.55	0/0	28/30 ; 0/30
	MME count	60	90.00 → 90.00 (+0.00)	90.32 → 90.00	1/4	27/28 ; 0/26

Aggregate flip behavior. Across the three models and the three main benchmarks, EvidenceFlow improves accuracy in every cell, correcting 1944/4922 baseline false affirmatives (39.5%) with 791/22855 true-affirmative losses (3.5%) and no true-negative flips, so false affirmatives are flipped 11.4 times as often as true ones.

D.3 Same-denominator POPE-full comparison

Table 13 compares EvidenceFlow with VCD and MFCD on POPE full under a single denominator, separate from the three main benchmarks.

Table 13: Same-denominator comparison on POPE full. The baseline column reports accuracy/F1-Yes; method columns report percentage-point changes in accuracy/F1-Yes.

Model	Baseline Acc./F1	EvidenceFlow	VCD	MFCD
InternVL3-8B	89.29 / 89.63	+0.70 / +0.53	-0.90 / -0.90	-1.00 / -1.06
DeepSeek-VL2-small	89.74 / 89.70	+0.10 / +0.09	-1.88 / -2.06	-1.32 / -1.42
Qwen3-VL-8B	88.31 / 87.37	+0.21 / +0.13	+0.32 / +0.33	+0.30 / +0.32

D.4 Protocol note for adapted baselines

VCD, MFCD, PAI, VTI, and REVIS are rerun under the binary object-existence protocol used by EvidenceFlow, which fixes the answer space, margin definition, and model–benchmark denominator before any comparison. Under this protocol, the VTI and REVIS steering vectors available for these models drive them toward a near-constant answer in most cells, so those rows describe the adaptation; the original papers report each method in its native setting.

D.5 Complete method comparison

Table 14 reports standalone changes for every compared method under the unified protocol, with each baseline listed once; Table 12 gives the corresponding flip accounting for EvidenceFlow.

Table 14: Standalone method comparison under the unified object-existence protocol. The baseline column reports Acc/F1-Yes, and each method cell reports ΔAcc/ΔF1-Yes in percentage points relative to the baseline in the same row. EF denotes EvidenceFlow.

Model	Benchmark	Baseline	EF	VCD	MFCD	PAI	VTI	REVIS
InternVL3-8B	POPEv2	72.8/78.5	+8.7/+5.6	+1.5/+0.8	+5.9/+3.5	+0.2/+0.1	-22.8/-11.8	-21.8/-11.5
	RePOPE	89.3/88.8	+4.1/+3.8	-0.7/-0.7	+0.0/-0.2	+0.2/+0.2	-46.1/-28.4	-44.0/-27.7
	Causal-HalBench	82.6/81.6	+3.6/+2.4	+0.1/+0.2	+1.3/+0.8	+0.5/+0.5	-39.7/-21.6	-37.9/-21.2
	MME existence	98.3/98.4	+1.7/+1.6	-1.7/-1.6	-1.7/-1.7	+0.0/+0.0	-48.3/-31.7	-48.3/-31.7
	MME count	85.0/87.0	-1.7/-1.7	+0.0/+0.0	+1.7/+1.2	-3.3/-2.5	-35.0/-20.3	-33.3/-19.6
DeepSeek-VL2-small	POPEv2	80.5/83.3	+2.4/+1.6	+1.1/+0.5	-0.7/-0.9	-1.5/-1.1	-30.5/–	-29.7/-16.7
	RePOPE	93.0/92.1	+0.3/+0.2	+0.2/+0.2	+0.5/+0.5	-0.5/-0.4	-36.2/–	-47.9/-31.4
	Causal-HalBench	88.9/87.4	+0.9/+0.7	+0.1/+0.0	+0.1/-0.1	-0.5/-0.4	-31.7/–	-43.8/-27.0
	MME existence	98.3/98.3	+0.0/+0.0	+0.0/+0.0	+0.0/+0.0	+0.0/+0.0	-48.3/–	-48.3/-31.6
	MME count	81.7/83.6	+1.7/+0.8	-5.0/-4.8	-1.7/-1.2	-3.3/-2.4	-31.7/–	-3.3/-2.4
Qwen3-VL-8B	POPEv2	86.2/87.5	+5.0/+4.0	-0.7/-0.6	-1.5/-1.1	-0.9/-0.6	-36.2/-20.8	-32.9/-19.8
	RePOPE	93.5/92.8	+0.9/+0.8	-0.4/-0.5	-0.6/-0.7	-0.8/-0.8	-50.2/-32.4	-47.2/-31.5
	Causal-HalBench	86.6/84.9	+1.1/+0.3	+0.1/+0.1	-0.7/-0.6	-0.9/-0.7	-43.7/-24.9	-40.6/-24.2
	MME existence	100.0/100.0	-3.3/-3.4	-1.7/-1.6	-6.7/-6.7	+0.0/+0.0	-50.0/-33.3	-48.3/-32.6
	MME count	90.0/90.3	+0.0/-0.3	+0.0/+0.0	-1.7/-1.4	+0.0/+0.3	-40.0/-23.6	-38.3/-23.6