WHEN ATTENTION HEADS HERD: SPECTRAL RELEASE FOR FAITHFUL MULTIMODAL REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-form multimodal reasoning improves capability but can let a fluent chain drift from visual evidence. Existing test-time remedies target output distributions, fixed heads, or attention intermediates that stock fused scaled-dot-product attention (SDPA) does not expose. We identify a distinct population-level spectral pattern in residual-stream head contributions. Under *spectral herding*, temporal innovations remain concentrated around a dominant residual-space mode even as its carrier heads change. The leading spectral share persists through carrier turnover, exceeding a row-energy-preserving permutation null across three models. Yet spectral peaks occur in both factual and hallucinated traces: factual events release them as innovation rises, whereas hallucinated events retain them, producing a *failed spectral release*. We introduce **SPLIT** (**S**pectral **P**eak **L**iberation via **I**nstantaneous **T**urnover), a training-free, single-pass controller that tracks this mode and redistributes gain toward the head bulk with exact reference-energy conservation. We characterize conditions for sketch stability and temporal tracking, proving permutation-equivariant steepest release. Across five models and eight benchmarks, one shared setting improves the underlying models and outperforms five recent baselines across diverse tasks, visual interfaces, and reasoning backbones. Operating post-SDPA and pre-W_O, SPLIT preserves stock fused attention, whereas attention-explicit baselines materialize attention tensors under eager execution.

1 INTRODUCTION

More reasoning is useful only while its premises remain grounded. Multimodal large reasoning models (MLRMs) solve difficult visual problems through extended trajectories (Yang et al., 2025; Huang et al., 2025; Wang et al., 2026b), but added deliberation can dilute visual reliance and propagate one unsupported premise through otherwise coherent logic (Dong et al., 2026; Xu et al., 2026a; Ma et al., 2026; Peng et al., 2026; Zhang et al., 2026f). The final answer may therefore look earned after the trace has invented what it needed to see.

Existing remedies reacquire or preserve visual evidence (Ghosal et al., 2026; Xi et al., 2026; Zhang et al., 2026b; Liufu et al., 2026), or alter token distributions, latent states, head gains, frequency components, and causal routes (Leng et al., 2024; Xu et al., 2026b; Lu et al., 2026a;b; Cheng et al., 2026b). Their deployment cost follows the object being controlled. Probability matrices, raw scores, or route-resolved products and gradients lie outside an unmodified fused-SDPA call (Huang et al., 2024a; Huo et al., 2024; Chen et al., 2026c; Wang et al., 2026c; Lu et al., 2026a;b; Cheng et al., 2026b). VGA shows that an algebraic output-side reformulation can instead remain FlashAttention-compatible without computing attention weights (Zhao et al., 2025). This suggests a missing design point: an online signal that is internally selective yet available after attention.

We find that signal by treating the *head population*, rather than an attention map or named head, as the unit of analysis. Like a transient, leaderless crowd, its members change while remaining aligned with one residual-space direction, a state we call *spectral herding*. Attention weights alone cannot expose it because realized contribution also depends on values, output projections, residual composition, and later layers (Abnar & Zuidema, 2020; Sun & Marasović, 2021; Chefer et al., 2021b; Elhage et al., 2021; Rai et al., 2024). We therefore form an angular Gram operator over temporally centered head contributions in shared residual coordinates. The operator preserves directional agreement across the evolving population at each decoding step while factoring out headwise scale.

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.

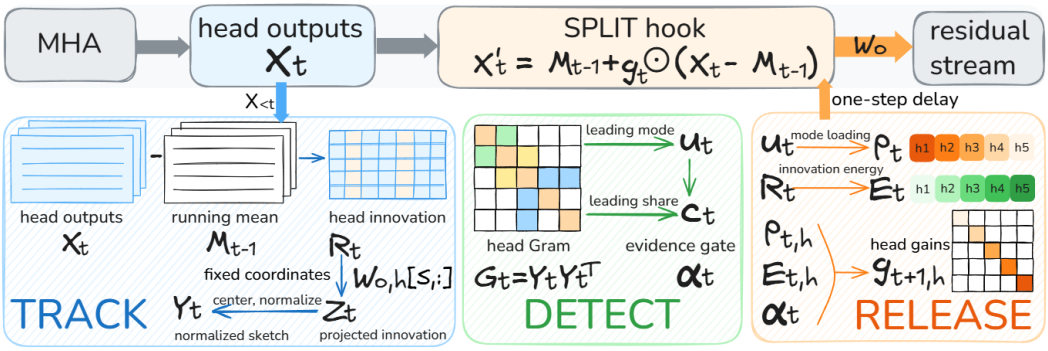


Figure 1: **SPLIT tracks and releases spectral herding outside native attention.**

Three analyses distinguish this pattern from alternative explanations. The population direction persists as its top-loading heads turn over; its leading share exceeds a row-energy-preserving permutation null across three models, by 82.1% over 2,000 R1-OneVision generations. The excess isolates coordination beyond fixed heads or unequal energy. Spectral peaks nevertheless occur in factual and hallucinated traces. Their matched-event dynamics separate only when innovation rises: factual mentions disperse the dominant share, whereas hallucinated mentions retain it. We call this response a *failed spectral release*.

This finding yields **SPLIT**: **S**pectral **P**eak **L**iberation via **I**nstantaneous **T**urnover. Figure 1 shows its track–detect–release loop. SPLIT sketches the evolving mode, gates on retained concentration under innovation surprise, and redistributes gain from current carriers toward the head bulk. A one-token lag preserves causality; spherical retraction preserves reference innovation energy. Acting after native attention and before W_O, it retains stock fused SDPA, RoPE, softmax, the KV cache, and the output projection in one decoding branch with one strength parameter. Detection and release thus act on the same instantaneous population geometry written to the residual stream.

Our contributions are:

- **A new population-level failure mode.** We identify failed spectral release and distinguish it from fixed-head effects, unequal head energy, and the mere presence of a spectral peak using carrier, permutation-null, and matched-event evidence.
- **A fused-attention-compatible controller with formal guarantees.** SPLIT combines fixed-coordinate sketching, online tracking, calibrated detection, and energy-preserving causal release. We prove sketch stability, tracking under drift, and permutation-equivariant Riemannian steepest release.
- **Broad gains across accuracy and efficiency.** Our evaluation spans five models, eight benchmarks, and five recent baselines. With one shared setting, SPLIT improves every evaluated model on all eight benchmarks and delivers broad gains across hallucination, perception, and reasoning metrics, including relative accuracy gains of up to 13.4% over vanilla decoding. SPLIT preserves stock SDPA, whereas attention-explicit controls require eager execution and materialize attention tensors throughout autoregressive decoding.

2 Related Work

Faithful reasoning requires preserving visual premises. Reasoning post-training and test-time deliberation strengthen MLRMs (Yang et al., 2025; Huang et al., 2025; Wang et al., 2026b), but can leave unfaithful traces. Analyses separate perception from reasoning-induced hallucination, expose weakened visual reliance in longer chains, and localize divergent steps (Dong et al., 2026; Xu et al., 2026a; Ma et al., 2026; Peng et al., 2026; Zhang et al., 2026f). Evidence-centric methods reacquire or reorganize inputs through zooming, region requests, refocusing, perception circuits, reflection, or a dedicated reward (Zheng et al., 2025; Jiang et al., 2026; Ghosal et al., 2026; Avogaro et al., 2026; Zhang et al., 2026d; Xiao et al., 2025). SPLIT asks the complementary question: whether a fixed image–context pair produces abnormal coordination inside the frozen decoder.

Inference-time control has a consequential execution boundary. Training-free mitigation includes contrastive or rollback decoding (Leng et al., 2024; Wang et al., 2024; Huang et al., 2024a; Huo et al., 2024), attention-score and route control (Chen et al., 2026c; Wang et al., 2026c; Lu et al., 2026a;b; Cheng et al., 2026b), and latent or residual editing (Xu et al., 2026b; Lin et al., 2026; Wu et al., 2026; Zhang et al., 2026a; Zou et al., 2025). LEAD switches between discrete and latent reasoning, MCoT revises divergent steps, RAL applies role-dependent head gains, FLASH performs spatial–frequency attention surgery, and CRG suppresses conflicting routes (Xu et al., 2026b; Ma et al., 2026; Lu et al., 2026a;b; Cheng et al., 2026b). Methods requiring explicit probabilities, raw scores, or route-resolved activations and gradients cannot use an otherwise unmodified fused-SDPA call. VGA shows that output-side reformulation can preserve FlashAttention without materializing attention weights (Zhao et al., 2025). SPLIT likewise respects this boundary, but derives adaptive headwise control from the post-SDPA tensor immediately before W_O.

SPLIT makes the changing head population the controlled object. Head contributions depend jointly on values, output projections, skip connections, and later layers (Abnar & Zuidema, 2020; Sun & Marasović, 2021; Chefer et al., 2021b; Elhage et al., 2021; Rai et al., 2024); dominant spectral modes can govern signal propagation (Noci et al., 2022; Nait Saada et al., 2025; Giorlandino & Goldt, 2026). SPLIT connects these views at the event level through an operator over head innovations in shared residual coordinates, a coordination-breaking null, and the mode's response to new information. Randomized and streaming estimators track the mode (Martinsson & Tropp, 2020; Vershynin, 2026; Greenbaum et al., 2020; Bienstock et al., 2022); a metric-aware, energy-preserving step releases its instantaneous carriers (Amari, 2016; Martens, 2020; Boumal, 2023). The resulting estimator–detector–controller chain uses spectral diagnosis as a causal control signal. Appendix B expands the literature map, and Appendix D states the assumptions and proofs.

3 METHOD

The population-level diagnosis motivates SPLIT, a training-free feedback controller for a frozen autoregressive MLRM. It identifies a population mode that outlives its carriers, tests whether that mode fails to release under innovation, and redistributes current carrier load on a reference-energy sphere. All state is derived online from headwise residual contributions, with one user-facing strength ρ. Pseudocode and proofs appear in Appendices A and D.

3.1 INTERFACE AND CAUSAL STATE

Consider a decoder with L attention layers, H heads per layer, head width d_h, and $d_{\text{model}} = Hd_h$ (Vaswani et al., 2017). At generated token t, let

$$X_{\ell,t} = \begin{bmatrix} x_{\ell,t,1}^\top & \cdots & x_{\ell,t,H}^\top \end{bmatrix}^\top \in \mathbb{R}^{H \times d_h} \tag{1}$$

denote the last-query head outputs immediately before the native output projection $W_{O,\ell}$. A pre-hook first saves this clean tensor, then applies the gain finalized from the preceding token. The attention computation and KV cache remain inside stock fused scaled-dot-product attention (SDPA), with `output_attentions` disabled.

SPLIT compares each clean observation with a streaming reference. With anchor $M_{\ell,t-1}$ and valid-visit count $n_{\ell,t}$, define

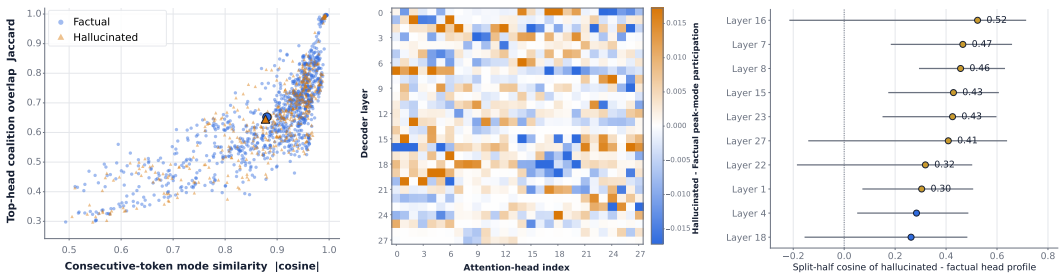
$$R_{\ell,t} = X_{\ell,t} - M_{\ell,t-1}. \tag{2}$$

$$M_{\ell,t} = M_{\ell,t-1} + \frac{X_{\ell,t} - M_{\ell,t-1}}{n_{\ell,t}}. \tag{3}$$

Here $R_{\ell,t}$ is the current clean deviation from the pre-update anchor, and $M_{\ell,t}$ is its exact online mean (Welford, 1962) over $n_{\ell,t}$ valid visits. Saving $X_{\ell,t}$ before applying a gain prevents the intervention from entering its own reference. Statistics from token t first affect token $t+1$; Appendix D.2 proves this strict causality and clean-anchor identity.

3.2 RESIDUAL-SPACE HERDING GEOMETRY

Attention weights describe routing, not the value- and output-dependent contribution written to the residual stream (Abnar & Zuidema, 2020; Sun & Marasović, 2021; Chefer et al., 2021b;a). Viewing



(a) Mode–carrier mobility (b) Outcome profile (c) Split-half recurrence

Figure 2: **The leading mode persists as its carrier heads turn over.**

heads as additive residual writers (Elhage et al., 2021; Rai et al., 2024), partition

$$W_{O,\ell} = \begin{bmatrix} W_{O,\ell}^{(1)} & \cdots & W_{O,\ell}^{(H)} \end{bmatrix} \tag{4}$$

and compare innovations after their respective W_O blocks. To avoid repeating the full projection, SPLIT fixes a layerwise selector $S_\ell \in \{0,1\}^{s \times d_{\text{model}}}$ with

$$s = \min\{d_{\text{model}}, \max(8, 4H)\}, \tag{5}$$

and sketches

$$z_{\ell,t,h} = S_\ell W_{O,\ell}^{(h)} r_{\ell,t,h} \in \mathbb{R}^s. \tag{6}$$

The selector is sampled once and cached, placing every token in fixed residual coordinates. Reusing these coordinates prevents sketch resampling from masquerading as temporal mode drift. Stacking rows into $Z_{\ell,t}$, let $C_H = I_H - \mathbf{1}\mathbf{1}^\top/H$ and $\mathcal{N}(B)_h = B_h / \max(\|B_h\|_2, \epsilon)$. SPLIT forms

$$\widetilde{Z}_{\ell,t} = C_H Z_{\ell,t}, \qquad Y_{\ell,t} = \mathcal{N}(\widetilde{Z}_{\ell,t}), \qquad G_{\ell,t} = Y_{\ell,t} Y_{\ell,t}^\top \succeq 0. \tag{7}$$

Head centering removes the instantaneous population mean, and row normalization separates directional coordination from head magnitude. Thus $G_{\ell,t}$ acts on headwise innovations rather than tokens.

SPLIT tracks the unit leading mode with three power steps at the first valid observation and one warm step thereafter:

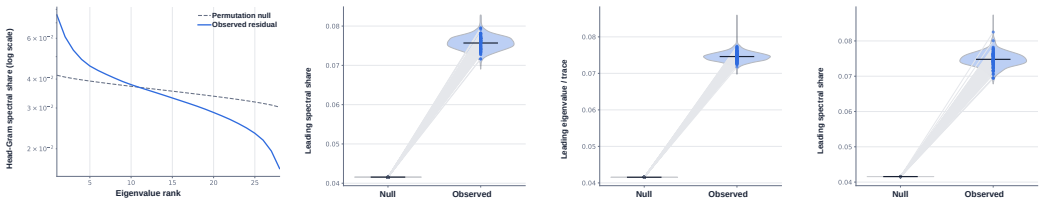
$$\bar{u}_{\ell,t} = G_{\ell,t} u_{\ell,t-1}, \qquad u_{\ell,t} = \frac{\bar{u}_{\ell,t}}{\max(\|\bar{u}_{\ell,t}\|_2, \epsilon)}, \tag{8}$$

$$c_{\ell,t} = \frac{u_{\ell,t}^\top G_{\ell,t} u_{\ell,t}}{\max(\operatorname{tr}(G_{\ell,t}), \epsilon)}, \qquad p_{\ell,t} = \frac{u_{\ell,t} \odot u_{\ell,t}}{\|u_{\ell,t}\|_2^2}. \tag{9}$$

Here $c_{\ell,t}$ is the fraction of coordination energy explained by the tracked direction, expressed as its Rayleigh quotient normalized by operator trace. The sign-invariant participation $p_{\ell,t}$ sums to one and locates the current carriers without assigning persistent roles to head identities. Randomized subspace estimation and streaming PCA motivate the fixed sketch and warm update (Martinsson & Tropp, 2020; Bienstock et al., 2022); Appendices D.3 and D.4 bound mode error and tracking under drift. Zero-energy states return the native tensor.

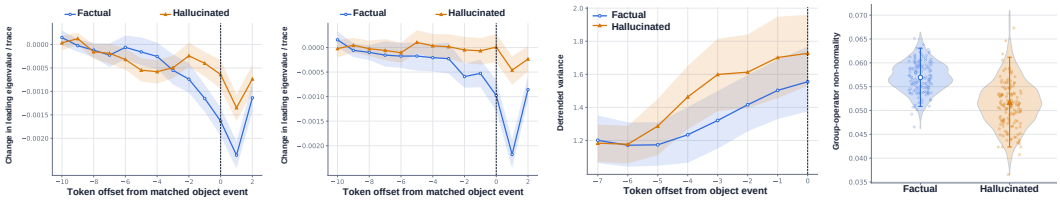
The population mode persists despite carrier turnover. Figure 2 tests the fixed-head explanation. Consecutive population directions remain similar although their top-loading heads only partly overlap; outcome-conditioned profiles have mixed signs and modest split-half recurrence. The stable object is the collective direction that the heads jointly occupy in shared residual coordinates, not a persistent list of harmful heads, motivating dynamic carrier tracking.

Coordination exceeds the matched null, but is not itself error. We compare $G_{\ell,t}$ with a row-energy-preserving permutation null that breaks cross-head alignment without changing any row norm. Figure 3 shows that the observed leading share exceeds this null in all three settings; across 2,000 R1-OneVision CHAIR generations, its mean rises from 0.04156 to 0.07569, an 82.1% uplift. The mode also appears in factual traces, so the null establishes genuine coordination while ruling out peak presence alone as a hallucination criterion.



(a) CHAIR scree (b) R1-OneVision (c) Vision-R1 (d) ThinkLite-VL

Figure 3: **Observed concentration exceeds an energy-matched null in all three models.**



(a) R1-OneVision (b) Vision-R1 (c) Variance (d) Non-normality

Figure 4: **Matched-event dynamics of failed spectral release.**

3.3 Detecting Failed Spectral Release

The matched-event comparison reveals a dynamical signature associated with hallucinated trajectories. A *spectral release* is a transient decrease of $c_{\ell,t}$ during energetic innovation; a *failed release* occurs when innovation rises but the peak remains above its recent layer-local target. Position-matched factual events release more strongly than hallucinated events in R1-OneVision and Vision-R1 (Figure 4a–b). Hallucinated trajectories also show higher detrended variance before the event (panel c), whereas factual traces show greater group-operator non-normality (panel d), consistent with richer directional redistribution. The joint pattern is active but insufficient population reorganization: event arrival with retained concentration, the condition used by the detector.

For each layer, bounded histories $\mathcal{H}_{c,\ell,t}$ and $\mathcal{H}_{v,\ell,t}$ retain the preceding H valid leading shares and innovation energies, a fixed window that follows the local regime. Write $\mathrm{MAD}(\mathcal{H}) = \mathrm{median}_{x\in\mathcal{H}}\,|x - \mathrm{median}(\mathcal{H})|$, $\theta_H = 1/H$ for the isotropic leading share, and $a \vee b = \max\{a, b\}$. Suppressing layer and token subscripts, SPLIT calibrates

$$b_c = \mathrm{median}(\mathcal{H}_c), \qquad s_c = \mathrm{MAD}(\mathcal{H}_c) \vee \frac{(b_c - \theta_H)_+}{H} \vee \epsilon, \qquad q_c = \theta_H \vee (b_c - s_c). \tag{10}$$

Here b_c, s_c, and q_c are the recent center, robust local scale, and isotropy-floored release target. Two scores separate persistence from magnitude:

$$r_t = \mathrm{clip}_{[0,1]}\left(\frac{c_t - q_c}{s_c}\right), \qquad q_t = \mathrm{clip}_{[0,1]}\left(\frac{(c_t - \theta_H)_+}{\max\{b_c - \theta_H, \epsilon\}}\right). \tag{11}$$

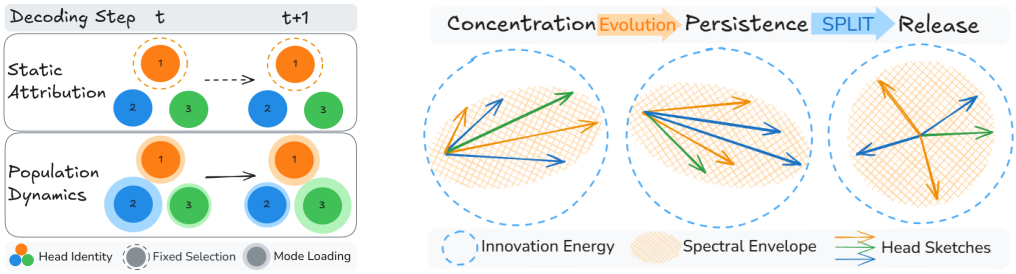
The release deficit r_t measures persistence above q_c, whereas q_t requires absolute separation from isotropy, preventing diffuse slow variation from triggering. With innovation energy $v_t = \|\widetilde{Z}_t\|_F^2/(Hs)$, evidence arrival is scored by

$$b_v = \mathrm{median}(\mathcal{H}_v), \qquad s_v = \mathrm{MAD}(\mathcal{H}_v) \vee |b_v|/H \vee \epsilon, \qquad e_t = \mathrm{clip}_{[0,1]}\left(\frac{v_t - b_v - s_v}{s_v}\right). \tag{12}$$

Here b_v and s_v set the normal energy scale; subtracting s_v activates e_t only beyond one robust deviation. Thus concentration alone cannot trigger SPLIT. The gain amplitude finalized for the next token is

$$\alpha_{\ell,t+1} = \mathrm{clip}_{[0,1]}\left(\frac{\rho}{\sqrt{L}} r_{\ell,t} q_{\ell,t} e_{\ell,t}\right), \tag{13}$$

where $\rho \geq 0$ is the only tunable strength. The product is a soft AND gate: absent a non-isotropic peak, a failed release, and unusual innovation simultaneously, the intervention vanishes or remains weak.



(a) Dynamic coordinates

(b) Peak-to-bulk release

Figure 5: **Population-control geometry.**

Calibration by the median and median absolute deviation (MAD) is dimensionless and layer-local; Appendix D.5 proves scale equivariance and resistance to a minority of extreme history values.

3.4 Energy-Constrained Peak-to-Bulk Release

Detection decides *when* to intervene; instantaneous participation decides *where*. Let $E_h = \|r_h\|_2^2$, $w_h = E_h / \sum_j E_j$, and $\bar{p}_w = \sum_h w_h p_h$. Equip the sphere $\mathcal{M} = \{g : \sum_h w_h g_h^2 = 1\}$ with $\langle a, b \rangle_w = \sum_h w_h a_h b_h$, and consider the local dominant-mode exposure $J(g) = \sum_h w_h p_h g_h$. Its steepest tangent descent at identity is $\bar{p}_w \mathbf{1} - p$. Normalizing this direction gives

$$d_h = \frac{\bar{p}_w - p_h}{\max_j |\bar{p}_w - p_j|}, \qquad \widetilde{g}_h = 1 + \alpha d_h. \tag{14}$$

Thus heads carrying more than the energy-weighted mean participation are attenuated, while the contemporaneous bulk is amplified. The rule depends on current squared loadings, not head identity, and satisfies $\sum_h w_h d_h = 0$.

A scalar retraction prevents the direction change from becoming an uncontrolled scale change:

$$\beta = \sqrt{\frac{\sum_h E_h}{\sum_h E_h \widetilde{g}_h^2}}, \qquad g_h = \beta \widetilde{g}_h. \tag{15}$$

The gate inferred at token t acts on the next clean innovation:

$$X'_{\ell,t+1,h} = M_{\ell,t,h} + g_{\ell,t+1,h}\big(X_{\ell,t+1,h} - M_{\ell,t,h}\big). \tag{16}$$

Equation 15 exactly preserves the *reference* energy $\sum_h E_{\ell,t,h}$ used to construct the lagged gate. The next innovation is unseen when the gate is set, so its response depends on adjacent-token drift. Appendix D.7 derives the Riemannian steepest-descent direction and its energy-preserving retraction. Under bounded drift, that lagged direction remains a descent direction at $t+1$ (Proposition 9). Appendix D.6 proves sign and head-permutation invariance, while Appendices D.10 and D.11 establish exact identity cases, a depth-independent budget $\sum_{\ell=1}^{L} \alpha_{\ell,t}^2 \le \rho^2$, and explicit local and downstream perturbation bounds.

Cost and backend compatibility. Per layer and token, the selected projection costs $O(Hsd_h) = O(sd_{\text{model}})$, the Gram operator $O(H^2 s)$, and one warm power step $O(H^2)$. Tracking uses matrix–vector updates in the small head space, and the controller runs after the native fused SDPA call while preserving that kernel. Persistent state consists of the clean anchor, two bounded scalar histories, the warm mode, gains, and cached coordinate indices. State initialization, batching, numerical safeguards, and full pseudocode appear in Appendix A.

4 Experiments

4.1 Experimental Setup

We evaluate five frozen open-weight MLRMs spanning three model families with distinct visual interfaces and reasoning styles: R1-OneVision-7B (Yang et al., 2025), Vision-R1-7B (Huang et al.,

Table 1: **SPLIT leads across five backbones and eight benchmarks.** COCO/A-OKVQA are POPE means; VH, MV, and MIR denote VHTest, MathVista$_{\text{mini}}$, and MIRAGE.

	General Understanding					Robust Perception			Visual Reasoning		Hallucination Mitigation					
	COCO		A-OKVQA		MME	POPEv2		VH	MV	MIR	CHAIR			AMBER		
Method	Acc↑	F1↑	Acc↑	F1↑	All↑	Acc↑	F1↑	All↑	Acc↑	Acc↑	C_S ↓	C_I ↓	F1↑	CH↓	F1↑	Sc↑
R1-OneVision																
Vanilla	81.46	79.95	81.66	81.86	2166.36	80.6	82.00	62.58	63.4	23.02	38.4	10.0	74.5	7.2	86.4	89.6
LEAD[CVPR'26]	83.28	81.82	83.17	83.09	2196.86	82.1	83.32	64.17	65.5	24.60	40.4	11.3	73.8	7.7	85.9	89.1
MCoT[CVPR'26]	82.79	81.42	81.93	82.17	2236.03	81.2	82.56	63.67	64.3	23.10	35.4	9.1	74.0	6.7	86.4	89.8
RAL[CVPR'26]	80.73	79.71	81.21	81.44	2075.89	79.9	81.51	62.33	64.1	22.65	34.4	9.3	73.8	6.9	85.9	89.5
CRG[ICML'26]	81.80	80.60	81.78	82.19	2188.24	79.1	80.04	63.58	62.7	22.72	33.0	8.6	74.0	6.5	86.7	90.1
FLASH[ICML'26]	81.82	80.34	81.77	82.18	2124.35	79.4	80.89	62.33	62.2	22.57	35.8	9.0	74.2	6.6	86.5	90.0
SPLIT (ours)	**84.58**	**83.03**	**84.10**	**83.86**	**2254.26**	**83.2**	**84.27**	**65.50**	**66.5**	**26.11**	**29.8**	**7.4**	**74.9**	**6.1**	**87.5**	**90.7**
Vision-R1																
Vanilla	88.77	88.41	85.15	86.45	2330.01	81.4	83.51	65.75	72.7	27.69	29.8	6.0	76.3	4.5	88.8	92.2
LEAD[CVPR'26]	88.79	88.43	85.19	86.47	2364.72	82.4	84.31	66.67	74.2	28.29	28.6	5.6	76.6	4.2	88.8	92.3
MCoT[CVPR'26]	88.82	88.44	85.29	86.53	2347.98	82.2	84.19	66.00	73.4	29.42	28.0	5.8	76.5	4.3	88.9	92.3
RAL[CVPR'26]	88.12	88.07	84.27	85.77	2201.96	81.3	83.35	63.83	73.6	27.84	27.8	5.7	76.0	4.3	85.7	90.7
CRG[ICML'26]	88.94	88.65	85.41	86.72	2363.85	79.8	82.47	64.67	71.1	26.56	26.6	5.3	76.4	4.1	88.9	92.4
FLASH[ICML'26]	89.01	88.51	85.38	86.64	2326.57	77.7	80.96	63.25	71.8	26.94	27.4	6.1	76.5	4.2	89.0	92.4
SPLIT (ours)	**89.75**	**89.39**	**86.27**	**87.42**	**2386.78**	**82.9**	**85.04**	**67.00**	**75.3**	**30.40**	**24.0**	**4.5**	**77.1**	**3.7**	**89.5**	**92.9**
ThinkLite-VL																
Vanilla	86.50	84.89	86.91	86.85	2369.53	89.4	89.79	70.08	75.2	30.17	35.8	8.2	77.4	4.9	90.4	92.8
LEAD[CVPR'26]	86.73	85.15	87.03	86.99	2388.46	89.8	90.17	70.75	75.9	31.53	34.8	8.3	76.0	4.9	90.5	92.8
MCoT[CVPR'26]	86.69	85.07	87.23	87.21	2389.41	89.8	90.19	71.08	75.6	32.05	33.8	7.9	76.9	4.8	90.5	92.8
RAL[CVPR'26]	86.01	84.30	86.38	86.36	2259.51	87.0	87.85	69.83	75.4	31.00	34.6	8.1	77.1	4.7	89.5	92.4
CRG[ICML'26]	86.58	84.94	87.26	87.14	2379.57	86.2	86.96	69.50	74.1	29.27	34.0	8.0	77.2	4.5	90.8	93.2
FLASH[ICML'26]	86.57	84.89	87.18	87.20	2375.85	85.4	86.43	68.42	73.7	28.74	34.2	7.7	77.3	4.6	90.7	93.0
SPLIT (ours)	**87.43**	**86.03**	**87.91**	**87.89**	**2405.49**	**91.1**	**91.21**	**71.25**	**76.4**	**33.26**	**29.8**	**7.1**	**77.8**	**4.1**	**91.4**	**93.7**
Qwen3-VL-Thinking																
Vanilla	87.87	86.95	87.10	87.66	2373.23	86.2	87.13	74.58	81.0	32.05	44.8	9.3	75.5	7.2	90.3	91.5
LEAD[CVPR'26]	88.13	87.27	87.29	87.84	2405.34	86.9	87.93	75.42	82.6	33.56	45.6	8.9	**76.5**	6.9	90.4	91.8
MCoT[CVPR'26]	88.07	87.16	87.28	87.79	2394.78	86.5	87.39	74.58	81.8	33.78	46.8	9.2	76.1	6.8	90.4	91.8
RAL[CVPR'26]	87.40	86.35	86.86	87.40	2316.65	86.7	87.63	75.25	82.0	33.41	45.8	8.8	75.7	7.0	89.7	91.3
CRG[ICML'26]	88.09	87.24	87.44	87.99	2391.85	87.0	87.71	75.75	80.5	31.00	41.0	8.6	76.2	6.6	90.5	92.0
FLASH[ICML'26]	88.07	87.09	87.49	87.95	2423.32	87.5	88.22	75.17	79.7	30.78	42.2	8.9	**76.5**	6.8	90.4	91.8
SPLIT (ours)	**88.79**	**87.77**	**88.18**	**88.67**	**2471.25**	**88.8**	**89.04**	**76.83**	**83.4**	**34.08**	**35.4**	**8.1**	76.1	**6.3**	**91.1**	**92.4**
InternVL3.5																
Vanilla	87.53	87.57	85.44	87.01	2315.84	81.4	83.63	67.08	78.4	31.08	46.4	9.6	76.2	7.7	86.9	89.6
LEAD[CVPR'26]	87.80	87.76	85.33	86.91	2310.83	78.4	81.35	66.58	79.7	32.51	46.6	10.1	76.3	7.5	86.6	89.5
MCoT[CVPR'26]	87.83	87.86	85.60	87.20	2328.36	82.8	84.78	69.42	80.7	31.98	45.2	9.2	**76.6**	7.6	87.2	89.8
RAL[CVPR'26]	87.07	86.90	84.97	86.40	2295.11	82.9	84.80	66.42	79.0	32.13	45.6	9.8	75.6	7.4	86.5	89.5
CRG[ICML'26]	87.85	87.87	85.74	87.09	2333.57	81.9	84.08	67.83	77.8	30.20	42.4	8.5	75.7	7.0	87.5	90.3
FLASH[ICML'26]	87.80	87.88	85.72	87.24	2339.27	81.8	84.17	68.25	77.1	29.80	43.8	8.9	75.8	7.2	87.2	90.0
SPLIT (ours)	**88.48**	**88.59**	**86.51**	**87.95**	**2384.15**	**84.5**	**85.71**	**70.58**	**81.6**	**33.18**	**38.6**	**7.7**	76.5	**6.7**	**88.1**	**90.7**

2025), ThinkLite-VL-7B (Wang et al., 2026b), Qwen3-VL-8B-Thinking (Bai et al., 2025), and InternVL3.5-8B-HF (Wang et al., 2025b). Eight benchmarks cover hallucination, perception, and visual reasoning: CHAIR (Rohrbach et al., 2018; Lin et al., 2014), POPE on COCO and A-OKVQA (Li et al., 2023; Schwenk et al., 2022), POPEv2 (Li et al., 2026c), AMBER (Wang et al., 2023), MME (Fu et al., 2023), VHTest (Huang et al., 2024b), MIRAGE (Dong et al., 2026), and MathVista$_{\text{mini}}$ (Lu et al., 2024). We compare vanilla decoding, SPLIT, and five training-free baselines: LEAD (Xu et al., 2026b), MCoT (Ma et al., 2026), RAL (Lu et al., 2026a), FLASH (Lu et al., 2026b), and CRG (Cheng et al., 2026b). All methods share model–record inputs and evaluation protocols, enabling a direct assessment of each intervention on identical cases; SPLIT uses $\rho = 1.0$ throughout. Each method uses its compatible backend: Vanilla, LEAD, MCoT, and SPLIT use stock SDPA; RAL, FLASH, and CRG require eager access to attention internals unavailable to SDPA. Appendices A and C provide implementation, runtime, sample-construction, and metric details.

4.2 Main Results

Across five models, eight benchmarks, and five recent baselines, Table 1 shows a consistent advantage across models and tasks. Including the four AMBER diagnostics in Appendix Table 6, SPLIT sets the strongest result across virtually the entire model–metric matrix while improving every model on every benchmark. The gains pair lower hallucination with retained or improved content coverage. Averaged

Table 2: **Both conjunctive detection and carrier-aligned dynamic control matter.** Rel., Dyn., Map, P2, and MV denote release, dynamic tracking, the unshuffled carrier map, POPEv2, and MathVista$_{\text{mini}}$.

Variant	Trigger Rel.	$r_t q_t$	e_t	$C_S \downarrow$	$C_I \downarrow$	P2↑	MV↑
R1-OneVision							
Vanilla	×	×	×	38.4	10.0	80.6	63.4
Variance only	✓	×	✓	31.8	7.8	82.9	64.6
Spectrum only	✓	✓	×	35.8	8.4	82.6	65.6
Always on	✓	×	×	35.0	7.9	82.1	65.8
SPLIT (full)	✓	✓	✓	**29.8**	**7.4**	**83.2**	**66.5**
InternVL3.5							
Vanilla	×	×	×	46.4	9.6	81.4	78.4
Variance only	✓	×	✓	40.0	8.2	81.5	81.1
Spectrum only	✓	✓	×	39.8	8.4	83.1	78.9
Always on	✓	×	×	38.8	7.7	82.9	80.8
SPLIT (full)	✓	✓	✓	**38.6**	**7.7**	**84.5**	**81.6**

Variant	Control state R_t	Dyn.	Map	$C_S \downarrow$	$C_I \downarrow$	P2↑	MV↑
R1-OneVision							
Vanilla	×	×	×	38.4	10.0	80.6	63.4
Static mode	✓	×	✓	32.6	8.1	80.2	65.9
Raw state	×	✓	✓	34.6	8.4	81.2	65.8
Shuffled mode	✓	✓	×	33.8	7.9	82.2	65.3
SPLIT (full)	✓	✓	✓	**29.8**	**7.4**	**83.2**	**66.5**
InternVL3.5							
Vanilla	×	×	×	46.4	9.6	81.4	78.4
Static mode	✓	×	✓	40.8	8.1	82.3	78.9
Raw state	×	✓	✓	41.4	8.2	82.7	80.7
Shuffled mode	✓	✓	×	42.4	8.3	82.6	81.2
SPLIT (full)	✓	✓	✓	**38.6**	**7.7**	**84.5**	**81.6**

over the five backbones, SPLIT reduces CHAIR C_S and C_I by 19.3% and 19.4%, respectively, while improving object F1. On AMBER, it lowers CH, Hal, and Cog and increases coverage for every backbone, jointly improving fidelity and informativeness.

Object-presence evaluation reveals the same pattern under increasingly difficult negatives. SPLIT ranks first in both accuracy and F1 for every backbone on the COCO and A-OKVQA variants of POPE, improving mean accuracy over Vanilla by 1.38 and 1.34 points. Its advantage grows on counterfactual POPEv2, where the average gains reach 2.30 points in accuracy and 1.84 points in F1. The same ordering holds across random, popular, adversarial, and counterfactual negatives; complete split-wise and class-balance results appear in Appendix Tables 5 and 6.

These gains also hold in tests of both direct visual perception and extended reasoning: at the same $\rho = 1.0$ on stock SDPA, SPLIT raises MME by 69.4 points on average and MIRAGE by 9.0%, while all five models improve on VHTest and MathVista$_{\text{mini}}$.

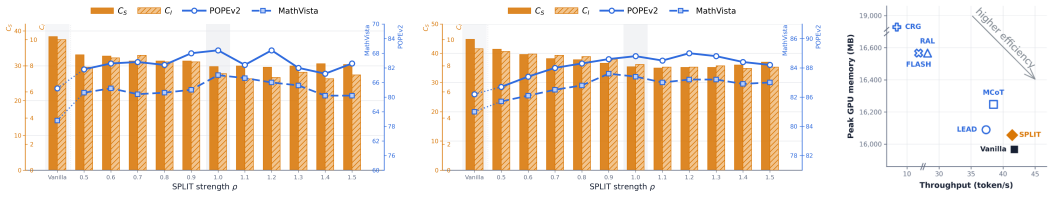
4.3 Component Analysis

Table 2 isolates the trigger and control state on R1-OneVision and InternVL3.5 across CHAIR, POPEv2, and MathVista$_{\text{mini}}$. With all other operations fixed, the left panel asks whether detection needs both unusual innovation and retained spectral concentration. Full SPLIT is best or tied best in all eight backbone–metric comparisons, whereas variance-only and spectrum-only detection trade strength across models and tasks. Always-on release remains competitive in isolated cells but trails the full trigger in seven of eight comparisons. The conjunction $r_t q_t e_t$ therefore supplies the selectivity needed to act on failed release rather than on either novelty or concentration alone.

The right panel asks what SPLIT must track once an event is detected. Estimating from the raw state consistently weakens performance because persistent content is no longer separated from new writes by temporal centering. Freezing the first valid mode is especially costly on R1-OneVision POPEv2 and InternVL3.5 MathVista, where the relevant population direction changes as the trace unfolds. Shuffling participation also degrades every metric: it preserves the global concentration signal but assigns release to the wrong heads. SPLIT therefore needs both a centered, time-varying mode and its corresponding instantaneous carriers.

4.4 Hyperparameter Sensitivity

SPLIT exposes one global strength ρ. Figure 6 sweeps ρ from 0.5 to 1.5 on R1-OneVision-7B and Qwen3-VL-8B-Thinking while holding all other settings fixed. Every tested value improves all four metrics over Vanilla on both backbones, with a broad high-performing region over $\rho \in [0.8, 1.3]$; the shared default $\rho = 1.0$ lies near its center and is used unchanged in the main evaluation. This stability matches the controller's division of labor: ρ scales the bounded amplitude $\alpha_{\ell,t} = (\rho/\sqrt{L}) r_{\ell,t} q_{\ell,t} e_{\ell,t}$ after the detector has selected an event and its carriers, so moderate changes adjust release strength without changing where or when SPLIT acts during the reasoning trajectory.



(a) R1-OneVision-7B (b) Qwen3-VL-8B-Thinking (c) Efficiency

Figure 6: **Strength sensitivity and decoding efficiency.** Gray bands mark Vanilla and the default $\rho = 1.0$.

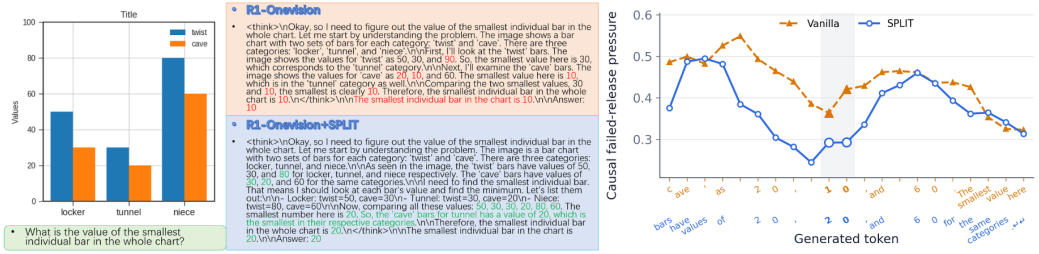


Figure 7: **SPLIT corrects an early visual readout before it propagates through reasoning.**

4.5 Efficiency Analysis

Figure 6c compares matched throughput and peak memory on R1-OneVision-7B. Each method uses its fastest compatible backend: stock SDPA where possible, and eager only when materialized attention is required. The resulting clusters expose the deployment cost of attention-explicit control. Within the SDPA cluster, SPLIT is 7.6–10.9% faster than LEAD and MCoT and has the lowest peak memory while retaining native attention kernels and KV caching.

4.6 Qualitative Analysis

In Figure 7, the right panel reports causal failed-release pressure, $P_t = \frac{1}{5}\sum_{j=t-4}^{t} \mathrm{RMS}_\ell(r_{\ell,j} q_{\ell,j} e_{\ell,j})$, the five-token mean of the gate signal setting $\alpha_{\ell,t+1}$ in Equation 13. Vanilla retains high pressure around its unsupported "10"; SPLIT releases it before the correct "20," then both traces reconverge under the shared comparison. Thus, carrier redistribution lowers failed-release pressure and changes the visual premise while leaving the subsequent comparison intact; further cases appear in Appendix E.

5 Conclusion

We identify *failed spectral release*, a population-level signature associated with reasoning drift: during high-innovation steps, a dominant residual-space mode remains concentrated even as its carrier heads turn over. Carrier-turnover analysis, an energy-matched permutation null, and matched-event comparisons distinguish this pattern from fixed-head effects, unequal head energy, and spectral concentration alone. SPLIT turns the diagnosis into a training-free, single-pass controller that tracks the evolving mode, detects retained concentration under innovation, and redistributes instantaneous carrier load while conserving reference energy. Across five models and eight benchmarks, SPLIT improves over vanilla decoding in every evaluated setting and outperforms five recent baselines on most reported metrics. Its post-SDPA, pre-W_O interface preserves fused attention without materializing attention probabilities or adding a second model pass, leaving the native efficient-decoding path intact. More broadly, SPLIT shifts inference-time intervention from static head identities to the time-varying geometry of a coordinated population whose carriers reorganize continually as evidence enters the residual stream. Taken together, these findings identify failed spectral release as a measurable signature of reasoning drift and motivate population-level spectral control as a practical intervention for reliable long-form multimodal reasoning under deployment constraints.

AI USE STATEMENT

In this work, generative AI tools were used solely for language polishing, grammar correction, and improving manuscript readability. Generative AI was not used for conceptualizing the research, developing the theoretical framework, deriving proofs, designing experiments, implementing code, or generating data. All scientific ideas, experimental findings, and claims were developed entirely by the authors. The authors have reviewed all AI-assisted text modifications to ensure technical precision and preserve the original meaning, and take full responsibility for the final content of this work.

ETHICS STATEMENT

This work studies multimodal-model reliability using publicly available models and standard research benchmarks. It involves no human subjects, private data, or collection of personal information. SPLIT is intended to reduce unsupported model outputs, and its use remains governed by the conditions and safeguards of the underlying models and application setting.

REPRODUCIBILITY STATEMENT

Appendix A specifies the model interfaces, unified evaluation protocol, baseline configurations, SPLIT settings, complete pseudocode, hardware environment, and run-level checks. Appendix C defines the evaluation metrics and aggregation procedures, Appendix D states the theoretical assumptions and provides complete proofs, and Appendix E supplies additional qualitative cases. Together, these materials document the implementation and evaluation choices needed to reproduce the reported results.

REFERENCES

Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pp. 4190–4197, 2020.

Shun-ichi Amari. *Information geometry and its applications*. Springer, 2016.

Niccolo Avogaro, Nayanika Debnath, Li Mi, Thomas Frick, Junling Wang, Zexue He, Hang Hua, Konrad Schindler, and Mattia Rigotti. Sparc: Separating perception and reasoning circuits for test-time scaling of vlms. *arXiv preprint arXiv:2602.06566*, 2026.

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

Abhinav Bhardwaj and Van Vu. Matrix perturbation: Davis–kahan in the infinity norm. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 880–934, 2024. doi: 10.1137/1.9781611977912.34.

Daniel Bienstock, Minchan Jeong, Apurv Shukla, and Se-Young Yun. Robust streaming pca. *Advances in Neural Information Processing Systems*, 35:4231–4243, 2022.

Nicolas Boumal. *An Introduction to Optimization on Smooth Manifolds*. Cambridge University Press, 2023. doi: 10.1017/9781009166164.

Zongsheng Cao, Yangfan He, Anran Liu, Jun Xie, Zhepeng Wang, and Feng Chen. Cofi-dec: Hallucination-resistant decoding via coarse-to-fine generative feedback in large vision-language models. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 10709–10718, 2025.

Hila Chefer, Shir Gur, and Lior Wolf. Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 397–406, 2021a.

Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 782–791, 2021b.

Guoqing Chen, Fu Zhang, Bingqian Liu, Chenglong Lu, and Jingwei Cheng. Vcgd: Visual clue guided decoding with caption model for mitigating hallucination in multimodal large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 20041–20049, 2026a.

Xinrong Chen, Xu Chu, Yingmin Qiu, Hengyuan Zhang, Jing Xiong, Shiyu Tang, Shuai Liu, Shaokang Yang, Cheng Yang, Hayden Kwok-Hay So, et al. Residual decoding: Mitigating hallucinations in large vision-language models via history-aware residual guidance. *arXiv preprint arXiv:2602.01047*, 2026b.

Yujia Chen, Rui Sun, Huayu Mai, Wangkai Li, Zhangyu He, Bingzhou Wang, Aibing Li, Wenzhang SUN, and Tianzhu Zhang. Beyond logits: Coherent hallucination mitigation via attention contrastive decoding. In *Forty-third International Conference on Machine Learning*, 2026c.

Yujia Chen, Rui Sun, Bingzhou Wang, Huayu Mai, Wangkai Li, Zhaoyang Li, Aibing Li, and Wenzhang SUN. Beyond blind noising: Disentangled visual rectification for hallucination mitigation in mllms. In *Forty-third International Conference on Machine Learning*, 2026d.

Junyu Cheng, Zhibiao Liang, Yidong Chen, and Shuangyin Li. Rfi: Rectified flow intervention for mitigating object hallucination in large vision-language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 3255–3263, 2026a.

Zhe Cheng, Wenyu Chen, Fode Zhang, and Dehuan Shen. Mitigating hallucinations in large vision-language models via causal route gating. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026b.

TianTian Dang, Chao Bi, Shufan Shen, Jinzhe Liu, Qingming Huang, and Shuhui Wang. Locate-then-sparsify: Attribution guided sparse strategy for visual hallucination mitigation. *arXiv preprint arXiv:2603.16284*, 2026.

Hamidreza Dastmalchi, Aijun An, Ali Cheraghian, and Hamed Barzamini. Fighting hallucinations with counterfactuals: Diffusion-guided perturbations for lvlm hallucination suppression. *arXiv preprint arXiv:2603.10470*, 2026.

Bowen Dong, Minheng Ni, Zitong Huang, Guanglei Yang, Wangmeng Zuo, and Lei Zhang. Mirage: Assessing hallucination in multimodal reasoning chains of mllm. *Advances in Neural Information Processing Systems*, 38:122910–122955, 2026.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 1(1):12, 2021.

Zhan Fa, Yue Duan, Jian Zhang, Lei Qi, and Yinghuan Shi. One token, two fates: A unified framework via vision token manipulation against mllms hallucination. *arXiv preprint arXiv:2603.10360*, 2026.

Hao Fang, Jinyu Li, Jiawei Kong, Tianqu Zhuang, Kuofeng Gao, Bin Chen, and Shu-Tao Xia. Seeing through the chain: Mitigate hallucination in multimodal reasoning models via cot compression and contrastive preference optimization. *arXiv preprint arXiv:2602.03380*, 2026.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

Soumya Suvra Ghosal, Youngeun Kim, Zhuowei Li, Ritwick Chaudhry, Linghan Xu, Hongjing Zhang, Jakub Zablocki, Yifan Xing, and Qin Zhang. Visref: Visual refocusing while thinking improves test-time scaling in multi-modal large reasoning models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 33404–33414, 2026.

Alessio Giorlandino and Sebastian Goldt. Two failure modes of deep transformers and how to avoid them: a unified theory of signal propagation at initialisation. In *International Conference on Learning Representations*, 2026.

Anne Greenbaum, Ren-cang Li, and Michael L Overton. First-order perturbation theory for eigenvalues and eigenvectors. *SIAM review*, 62(2):463–482, 2020.

Nhat Hoang-Xuan, Minh Vu, My T Thai, and Manish Bhattarai. Pas: Prelim attention score for detecting object hallucinations in large vision–language models. *arXiv preprint arXiv:2511.11502*, 2025.

Steven R Howard, Aaditya Ramdas, Jon McAuliffe, and Jasjeet Sekhon. Time-uniform, nonparametric, nonasymptotic confidence sequences. *The Annals of Statistics*, 49(2):1055–1080, 2021.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13418–13427, 2024a.

Wen Huang, Hongbin Liu, Minxin Guo, and Neil Gong. Visual hallucinations of multi-modal large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 9614–9631, 2024b.

Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Xu Tang, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models. *arXiv preprint arXiv:2503.06749*, 2025.

Fushuo Huo, Wenchao Xu, Zhong Zhang, Haozhao Wang, Zhicheng Chen, and Peilin Zhao. Self-introspective decoding: Alleviating hallucinations for large vision-language models. *arXiv preprint arXiv:2408.02032*, 2024.

Hongrui Jia, Chaoya Jiang, Shikun Zhang, and Wei Ye. Mitigating visual context degradation in large multimodal models: A training-free decoupled agentic framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9574–9585, 2026.

Pu Jian, Junhong Wu, Wei Sun, Chen Wang, Shuo Ren, and Jiajun Zhang. Look again, think slowly: Enhancing visual reflection in vision-language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 9262–9281, 2025.

Chaoya Jiang, Yongrui Heng, Wei Ye, Haiyang Xu, Ming Yan, Ji Zhang, Fei Huang, and Shikun Zhang. Vlm-r^3: Region recognition, reasoning, and refinement for enhanced multimodal chain-of-thought. *Advances in Neural Information Processing Systems*, 38:63841–63869, 2026.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13872–13882, 2024.

Qiming Li, Zekai Ye, Xiaocheng Feng, Weihong Zhong, Weitao Ma, and Xiachong Feng. Causal tracing of object representations in large vision language models: Mechanistic interpretability and hallucination mitigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 31645–31653, 2026a.

Quanjiang Li, Zhiming Liu, Wei Luo, Tingjin Luo, and Chenping Hou. Correcting visual blur induced by attention distraction to reduce hallucinations: Algorithm and theory. *arXiv preprint arXiv:2605.24602*, 2026b.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne 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, 2023.

Yifan Li, Kun Zhou, Xin Zhao, Lei Fang, and Jirong Wen. Analyzing and mitigating object hallucination: A training bias perspective. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 6636–6643, 2026c.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Yangguang Lin, Quan Fang, Yufei Li, Jiachen Sun, Junyu Gao, and Jitao Sang. Hulluedit: Single-pass evidence-consistent subspace editing for mitigating hallucinations in large vision-language models. *arXiv preprint arXiv:2602.22727*, 2026.

Jia Liufu, Qiangyu Yan, Zhehan Kan, Wenming Yang, Hailin Hu, Xinghao Chen, and Borui Jiang. Entropy-based visual re-perception inference for multimodal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9770–9779, 2026.

Haolang Lu, Bolun Chu, WeiYe Fu, Guoshun Nan, Junning Liu, Minghui Pan, Qiankun Li, Yi Yu, Hua Wang, and Kun Wang. Reallocating attention across layers to reduce multimodal hallucination. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4157–4167, 2026a.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations*, volume 2024, pp. 23439–23554, 2024.

Siqi Lu, Wei Suo, Yongbin Zheng, Jianhang Yao, Wanying Xu, and Peng Wang. Beyond attention imbalance: Mitigating hallucinations via spectral surgery. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026b.

Zhiwen Luo, Siyu Jiang, Weilong Jiang, and Kun He. Latent attention denoising: A training-free energy-based framework for mitigating hallucinations in vision-language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 28748–28777, 2026.

Ji Ma, Wei Suo, Peng Wang, and Yanning Zhang. Understanding and mitigating hallucinations in multimodal chain-of-thought models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 40224–40234, 2026.

James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020.

Per-Gunnar Martinsson and Joel A Tropp. Randomized numerical linear algebra: Foundations and algorithms. *Acta Numerica*, 29:403–572, 2020.

Thiziri Nait Saada, Alireza Naderi, and Jared Tanner. Mind the gap: A spectral analysis of rank collapse and signal propagation in attention layers. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 45561–45587, 2025.

Tuan Dung Nguyen, Minh Khoi Ho, Qi Chen, Yutong Xie, Nguyen Cam-Tu, Minh Khoi Nguyen, Dang Huy Pham Nguyen, Anton van den Hengel, Johan W Verjans, Phi Le Nguyen, et al. Beyond the global scores: Fine-grained token grounding as a robust detector of lvlm hallucinations. *arXiv preprint arXiv:2604.04863*, 2026.

Lorenzo Noci, Sotiris Anagnostidis, Luca Biggio, Antonio Orvieto, Sidak Pal Singh, and Aurelien Lucchi. Signal propagation in transformers: Theoretical perspectives and the role of rank collapse. *Advances in Neural Information Processing Systems*, 35:27198–27211, 2022.

Valentin Noël. Geometry of reason: Spectral signatures of valid mathematical reasoning. *arXiv preprint arXiv:2601.00791*, 2026.

Shuyi Ouyang, Hongyi Wang, Gongfan Fang, Xinyin Ma, Lanfen Lin, and Xinchao Wang. Taming the phantom: Token-asymmetric filtering for hallucination mitigation in large vision-language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 8206–8214, 2026.

Ruiying Peng, Xueyu Wu, Jing Lei, Lu Hou, Yuanzheng Ma, and Xiao-Hui Li. Deeper thought, weaker aim: Understanding and mitigating perceptual impairment during reasoning in multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12064–12073, 2026.

Zheng Qi, Chao Shang, Evangelia Spiliopoulou, and Nikolaos Pappas. Capturing gaze shifts for guidance: Cross-modal fusion enhancement for vlm hallucination mitigation. *arXiv preprint arXiv:2510.22067*, 2025.

Zhe Qian, Yanbiao Ma, Zhuohan Ouyang, Zhonghua Wang, Zhongxing Xu, Fei Luo, Xinyu Liu, Zongyuan Ge, Yike Guo, and Jungong Han. Cognitive pivot points and visual anchoring: Unveiling and rectifying hallucinations in multimodal reasoning models. *arXiv preprint arXiv:2604.10219*, 2026.

Daking Rai, Yilun Zhou, Shi Feng, Abulhair Saparov, and Ziyu Yao. A practical review of mechanistic interpretability for transformer-based language models. *arXiv preprint arXiv:2407.02646*, 2024.

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, 2018.

Jiacheng Ruan, Zongyun Zhang, Jingsheng Gao, Wenzhen Yuan, Ting Liu, and Yuzhuo Fu. Mpi-cd: Multi-path information contrastive decoding for mitigating hallucinations in large vision-language models. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 4251–4260, 2025.

Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. In *European conference on computer vision*, pp. 146–162. Springer, 2022.

Hyein Seo, Yuna Jeong, Mingyu Kang, Junhyeong Park, and Yong Suk Choi. Degap: Dynamic entropy-guided attention perturbation for contrastive decoding in large vision-language models. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026.

Kaiser Sun and Ana Marasović. Effective attention sheds light on interpretability. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 4126–4135, 2021.

Lexiang Tang, Xianwei Zhuang, Bang Yang, Zhiyuan Hu, Hongxiang Li, Lu Ma, Jinghan Ru, and Yuexian Zou. Not all tokens and heads are equally important: Dual-level attention intervention for hallucination mitigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 9439–9447, 2026.

Chongjun Tu, Peng Ye, Dongzhan Zhou, Tao Chen, and Wanli Ouyang. Mitigating low-quality reasoning in mllms: Self-driven refined multimodal cot with selective thinking and step-wise visual enhancement. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 9576–9584, 2026.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge University Press, 2 edition, 2026. doi: 10.1017/9781009490672.

Junyang Wang, Yuhang Wang, Guohai Xu, Jing Zhang, Yukai Gu, Haitao Jia, Jiaqi Wang, Haiyang Xu, Ming Yan, Ji Zhang, et al. Amber: An llm-free multi-dimensional benchmark for mllms hallucination evaluation. *arXiv preprint arXiv:2311.07397*, 2023.

Shan Wang, Maying Shen, Nadine Chang, Chuong Nguyen, Hongdong Li, and Jose M Alvarez. Mitigating multimodal hallucinations via gradient-based self-reflection. *arXiv preprint arXiv:2509.03113*, 2025a.

Tianbo Wang, Yuqing Ma, Kewei Liao, Zhange Zhang, Simin Li, Jinyang Guo, and Xianglong Liu. After: Mitigating the object hallucination of lvlm via adaptive factual-guided activation editing. In *International Conference on Learning Representations*, 2026a.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, Zhaokai Wang, Zhe Chen, Hongjie Zhang, Ganlin Yang, Haomin Wang, Qi Wei, Jinhui Yin, Wenhao Li, Erfei Cui, Guanzhou Chen, Zichen Ding, Changyao Tian, Zhenyu Wu, Jingjing Xie, Zehao Li, Bowen Yang, Yuchen Duan, Xuehui Wang, Zhi Hou, Haoran Hao, Tianyi Zhang, Songze Li, Xiangyu Zhao, Haodong Duan, Nianchen Deng, Bin Fu, Yinan He, Yi Wang, Conghui He, Botian Shi, Junjun He, Yingtong Xiong, Han Lv, Lijun Wu, Wenqi Shao, Kaipeng Zhang, Huipeng Deng, Biqing Qi, Jiaye Ge, Qipeng Guo, Wenwei Zhang, Songyang Zhang, Maosong Cao, Junyao Lin, Kexian Tang, Jianfei Gao, Haian Huang, Yuzhe Gu, Chengqi Lyu, Huanze Tang, Rui Wang, Haijun Lv, Wanli Ouyang, Limin Wang, Min Dou, Xizhou Zhu, Tong Lu, Dahua Lin, Jifeng Dai, Weijie Su, Bowen Zhou, Kai Chen, Yu Qiao, Wenhai Wang, and Gen Luo. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency, 2025b. URL https://arxiv.org/abs/2508.18265.

Xintong Wang, Jingheng Pan, Liang Ding, and Chris Biemann. Mitigating hallucinations in large vision-language models with instruction contrastive decoding. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 15840–15853, 2024.

Xiyao Wang, Zhengyuan Yang, Chao Feng, Hongjin Lu, Linjie Li, Chung-Ching Lin, Kevin Lin, Furong Huang, and Lijuan Wang. Sota with less: Mcts-guided sample selection for data-efficient visual reasoning self-improvement. *Advances in Neural Information Processing Systems*, 38: 118818–118850, 2026b.

Yujun Wang, Aniri, Jinhe Bi, Soren Pirk, and Yunpu Ma. Ascd: Attention-steerable contrastive decoding for reducing hallucination in mllm. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 10306–10314, 2026c.

Barry Payne Welford. Note on a method for calculating corrected sums of squares and products. *Technometrics*, 4(3):419–420, 1962.

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. *arXiv preprint arXiv:2602.11824*, 2026.

Gongli Xi, Kun Wang, Zeming Gao, Huahui Yi, Haolang Lu, Ye Tian, and Wendong Wang. Cluetracer: Question-to-vision clue tracing for training-free hallucination suppression in multimodal reasoning. *arXiv preprint arXiv:2602.02004*, 2026.

Tong Xiao, Xin Xu, Zhenya Huang, Hongyu Gao, Quan Liu, Qi Liu, and Enhong Chen. Perception-r1: Advancing multimodal reasoning capabilities of mllms via visual perception reward. *arXiv preprint arXiv:2506.07218*, 2025.

Yutong Xie, Zhenglin Hua, Ran Wang, Wing WY Ng, Xizhao Wang, and Yuheng Jia. Finding the correct visual evidence without forgetting: Mitigating hallucination in lvlms via inter-layer visual attention discrepancy. *arXiv preprint arXiv:2605.20965*, 2026.

Zhongxing Xu, Chengzhi Liu, Qingyue Wei, Juncheng Wu, James Zou, Xin Wang, Yuyin Zhou, and Sheng Liu. More thinking, less seeing? assessing amplified hallucination in multimodal reasoning models. *Advances in Neural Information Processing Systems*, 38:82878–82905, 2026a.

Zhongxing Xu, Zhonghua Wang, Zhe Qian, Dachuan Shi, Feilong Tang, Ming Hu, Shiyan Su, Xiaocheng Zou, Wei Feng, Dwarikanath Mahapatra, et al. Thinking in uncertainty: Mitigating hallucinations in mlrms with latent entropy-aware decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11064–11075, 2026b.

Shuo Yang, Yuwei Niu, Yuyang Liu, Yang Ye, Bin Lin, and Li Yuan. Look-back: Implicit visual re-focusing in mllm reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 11694–11702, 2026.

Yi Yang, Xiaoxuan He, Hongkun Pan, Xiyan Jiang, Yan Deng, Xingtao Yang, Haoyu Lu, Dacheng Yin, Fengyun Rao, Minfeng Zhu, et al. R1-onevision: Advancing generalized multimodal reasoning through cross-modal formalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2376–2385, 2025.

Liu Yu, Can Chen, Ping Kuang, Zhikun Feng, Fan Zhou, and Gillian Dobbie. Dismantling pathological shortcuts: A causal framework for faithful lvlm decoding. *arXiv preprint arXiv:2606.27596*, 2026a.

Liu Yu, Zhonghao Chen, Ping Kuang, Zhikun Feng, Fan Zhou, Lan Wang, and Gillian Dobbie. Causally-grounded dual-path attention intervention for object hallucination mitigation in lvlms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 36021–36029, 2026b.

Chengsheng Zhang, Chenghao Sun, Xinyan Jiang, Wei Li, and Xinmei Tian. Prefill-time intervention for mitigating hallucination in large vision-language models. *arXiv preprint arXiv:2604.25642*, 2026a.

Dongxu Zhang, Yiding Sun, Cheng Tan, Wenbiao Yan, Ning Yang, Jihua Zhu, and Haijun Zhang. Chain-of-thought compression should not be blind: V-skip for efficient multimodal reasoning via dual-path anchoring. *arXiv preprint arXiv:2601.13879*, 2026b.

Feiran Zhang, Yixin Wu, Zhenghua Wang, Xiaohua Wang, Changze Lv, Xuanjing Huang, and Xiaoqing Zheng. Vib-probe: Detecting and mitigating hallucinations in vision-language models via variational information bottleneck. *arXiv preprint arXiv:2601.05547*, 2026c.

Haoyu Zhang, Yuwei Wu, Pengxiang Li, Xintong Zhang, Zhi Gao, Rui Gao, Mingyang Gao, Che Sun, and Yunde Jia. Bridging modality disconnect in self-reflection via closed-loop visually grounded verification. *arXiv preprint arXiv:2602.18746*, 2026d.

Ruipeng Zhang, Zhihao Li, CL Chen, and Tong Zhang. Steer where it matters: Token-level visual-sensitivity steering for lvlms hallucination mitigation. *arXiv preprint arXiv:2606.07647*, 2026e.

Yongchang Zhang, Oliver Ma, Tianyi Liu, Guangquan Zhou, and Yang Chen. See it, say it, sorted: An iterative training-free framework for visually-grounded multimodal reasoning in lvlms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11933–11942, 2026f.

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

Ziwei Zheng, Michael Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, Chao Shen, and Xing Yu. Deepeyes: Incentivizing" thinking with images" via reinforcement learning. *arXiv preprint arXiv:2505.14362*, 2025.

Li'an Zhong, Ziqiang He, Jibin Zheng, Jin Li, Z Jane Wang, and Xiangui Kang. Adaiat: Adaptively increasing attention to generated text to alleviate hallucinations in lvlm. *arXiv preprint arXiv:2603.04908*, 2026.

Yingxuan Zhuang, Jingxiao Yang, Miao Pan, Cheng Tan, Yuxiang Cai, Siwei Tan, Chen Zhi, Xuhong Zhang, Jianwei Yin, and Jintao Chen. Mitigating manifold departure: Uncertainty-aware subspace rectification for trustworthy mllm decoding. *arXiv preprint arXiv:2606.09859*, 2026.

Zhengtao Zou, Ya Gao, Jiarui Guan, Bin Li, and Pekka Marttinen. Adaptive residual-update steering for low-overhead hallucination mitigation in large vision language models. *arXiv preprint arXiv:2511.10292*, 2025.

A Experimental and Implementation Details

This appendix specifies the evaluation protocol, model interfaces, compared methods, SPLIT configuration, and reproducibility checks used in Section 4.1. Sections A.1–A.3 define the common evaluation contract, Section A.4 fixes every competing intervention, Sections A.5–A.7 describe the deployed controller, and Section A.8 states the statistical and systems checks. Table 3 collects the notation shared by the method, algorithm, and proofs. SPLIT derives its controller state solely from the frozen backbone's online activations.

Table 3: **Summary of key notations and operations.** Shapes omit layer and token subscripts where unambiguous; Δ^{H-1} is the probability simplex and $\mathbb{S}_+^H$ the positive-semidefinite cone.

Symbol or operation	Shape or range	Definition and role
$\ell, t, h; L, H, d_h, d_{\text{model}}, s$	indices; positive integers	Layer, generated-token, and head indices; layer count, query-head count, head width, model width, and fixed sketch width.
$X_{\ell,t}, M_{\ell,t-1}, R_{\ell,t}$	$\mathbb{R}^{H\times d_h}$	Clean pre-W_O head output, streaming clean anchor, and temporal innovation $R = X - M$.
$W_{O,\ell}^{(h)}, S_\ell$	$\mathbb{R}^{d_{\text{model}}\times d_h}$; $\{0,1\}^{s\times d_{\text{model}}}$	Head-h block of the native output projection and the cached residual-coordinate selector.
$z_h, Z, \widetilde{Z}, Y$	$\mathbb{R}^s$; $\mathbb{R}^{H\times s}$	Projected head innovation, stacked sketch, head-centered sketch, and row-normalized population matrix.
$C_H, \mathcal{N}, G$	$H\times H$; row map; $H\times H$	Head-centering operator, ϵ-floored row normalization, and deployed Gram operator $G = YY^\top$.
$G^\star, \widehat{G}$	$\mathbb{S}_+^H$	Full and coordinate-sketched normalized operators used in Appendix D.3; the deployed G is $\widehat{G}$.
u, c, p	$\mathbb{R}^H$; $[0,1]$; Δ^{H-1}	Warm leading mode, trace-normalized Rayleigh share, and sign-invariant head participation $p = u\odot u/\|u\|_2^2$.
$\mathcal{H}_c, \mathcal{H}_v$	at most H scalars each	Bounded histories of leading shares and centered-sketch innovation energies.
$b_c, s_c, q_c; b_v, s_v$	scalars	Median baselines, robust MAD-based scales, release target q_c, and the corresponding energy reference.
v, r, q, e	$v\geq 0$; $r,q,e\in[0,1]$	Innovation energy and the release-deficit, non-isotropic-peak, and event-surprise detector factors.
$\rho, \alpha_{\ell,t+1}$	nonnegative; $[0,1]$	Global user strength and causal layerwise amplitude $\alpha = \text{clip}_{[0,1]}((\rho/\sqrt{L})rqe)$.
$E, w, \bar{p}_w$	$\mathbb{R}_+^H$; Δ^{H-1}; scalar	Head innovation energies, normalized energy weights, and energy-weighted mean participation.
$d, \widetilde{g}, \beta, g$	$\mathbb{R}^H$; $\mathbb{R}^H$; scalar; $\mathbb{R}^H$	Peak-to-bulk tangent direction, provisional gain, radial-retraction factor, and final energy-preserving gain.
$\mathcal{M}, J(g), X'_{\ell,t+1}$	manifold; scalar; $\mathbb{R}^{H\times d_h}$	Weighted energy sphere, dominant-mode exposure surrogate, and the lagged controlled head tensor.
$\eta, \gamma, \kappa_t, \delta_t$	nonnegative scalars	Operator error, leading eigengap, spectral contraction ratio λ_2/λ_1, and adjacent-token leading-mode drift.

A.1 Unified Evaluation Protocol

All compared methods are inference-time procedures over frozen backbones. For a fixed model–benchmark pair, the image processor, tokenizer, chat template, visual-token boundary convention, prompt, maximum image area, generation ceiling, stop rule, answer parser, and metric code are identical. A run activates exactly one of the following: vanilla decoding, SPLIT, LEAD, MCoT, RAL, FLASH, or CRG. Method state is reset between examples and never shared across benchmark records.

Generation is greedy with temperature 0 and batch size one for fair method comparison; this keeps prompt length, generated-token accounting, and per-sequence controller state matched across methods. Bfloat16 weights, KV caching, and seed 42 are shared by vanilla decoding, MCoT, RAL, FLASH, CRG, and SPLIT. LEAD retains the categorical draw $r_t \sim p_t$ in its discrete branch as specified by its mode-switch equation; its latent branch feeds back the probability-weighted embedding. We introduce no additional temperature, top-p, or top-k truncation into that distribution. Images are capped at 1024^2 pixels before each model's native dynamic-resolution processor creates visual tokens. The controller receives no resized image or visual-token tensor beyond what the frozen model itself consumes.

The maximum generation budget is 512 new tokens for all benchmarks. Binary evaluators parse the final decision and ignore unused budget after an end-of-sequence token; free-form and reasoning tasks retain the generated trace required by their official parser. Token throughput uses the actual generated-token count and full end-to-end generation time, and peak memory is measured over the same run.

CHAIR uses the first 500 unique COCO-val2014 image identifiers after numerical sorting, with the prompt *"Describe the image in detail."*. POPE evaluates all random, popular, and adversarial records on both COCO and A-OKVQA. POPEv2, AMBER, MME, VHTest, and MIRAGE use their complete released evaluation records; MathVista uses the 1,000-record testmini split. Each

Table 4: **Checkpoint, interface, and controller geometry.** Decoder quantities are read from the resolved checkpoint configurations. H_q/H_{kv} denotes the query-head and KV-head counts, and d is decoder width. The resulting sketch widths are architecture-derived rather than selected on a benchmark.

Backbone	Checkpoint and decoder geometry	SPLIT geometry	Visual interface	Reasoning protocol and transfer role
R1-OneVision-7B	`Fancy-MLLM/R1-Onevision-7B`; Qwen2.5-VL, L=28, H_q/H_{kv}=28/4, d_h=128, d=3584	s=112; 28 native `o_proj` pre-hooks	Native dynamic resolution; explicit vision-start/end span	Released chat template; cross-modal formalization and RL reasoning checkpoint
Vision-R1-7B	`Osilly/Vision-R1-7B`; Qwen2.5-VL, 28, 28/4, 128, 3584	s=112; same Qwen2.5-VL interface	Native dynamic resolution; explicit vision-start/end span	Released chat template; long-form R1-style multimodal reasoning
ThinkLite-VL-7B	`russwang/ThinkLite-VL-7B`; Qwen2.5-VL, 28, 28/4, 128, 3584	s=112; same Qwen2.5-VL interface	Native dynamic resolution; explicit vision-start/end span	Released `<think>` and boxed-answer instruction; data-efficient reasoning
Qwen3-VL-8B-Thinking	`Qwen/Qwen3-VL-8B-Thinking`; Qwen3-VL, L=36, H_q/H_{kv}=32/8, d_h=128, d=4096	s=128; 36 native `o_proj` pre-hooks	Interleaved M-RoPE and dynamic visual-token grid	Native thinking checkpoint; tests a newer decoder and multimodal position system
InternVL3.5-8B-HF	`OpenGVLab/InternVL3_5-8B-HF`; Qwen3 language decoder, 36, 32/8, 128, 4096	s=128; 36 native `o_proj` pre-hooks	InternViT, Dynamic High Resolution, and contiguous image-context tokens	Official thinking-mode system prompt; distinct visual encoder and image-context pipeline

model-facing instruction contains the benchmark query and, where required, its answer-format clause. Task-specific factual hints, demonstrations, and method-dependent prompts are excluded.

The rendered prompt has three transparent layers: benchmark query, checkpoint reasoning protocol, and checkpoint chat template. ThinkLite-VL appends its released `<think>...  </think>` and `\boxed{}` response instruction; InternVL3.5 uses its released thinking-mode system protocol; the other checkpoints use their native reasoning-enabled chat templates. The same rendered prompt is reused for every method on a fixed model–record pair. CHAIR disables the extra reasoning instruction because CHAIR evaluates a direct description rather than a solved-answer format.

Attention-backend contract. Vanilla decoding, LEAD, MCoT, and SPLIT retain the resolved checkpoint's SDPA implementation. SPLIT hooks the input to W_O after fused attention has returned its aggregated output, with attention probabilities left unmaterialized. RAL is executed through its released `eager_attention_forward` modification. FLASH and CRG also use eager attention because their published equations require pre-softmax score and value tensors or route-resolved attention activations and gradients. Each method therefore uses the fastest backend compatible with its published intervention; no SDPA-compatible method is moved to eager execution.

Official answer extraction. Generation and evaluation are kept separate. Each output is processed by the benchmark-specific parsing and aggregation protocol defined in Appendix C: rule-based tasks locate the expected final-answer region and apply the released task-typed normalizer, while MIRAGE uses its released LLM-based final-answer parser. Raw generations and evaluated outputs are both retained. If a rule-based benchmark cannot extract a typed answer, that record remains in the denominator with zero exact-match credit.

Paired comparison design. Method effects are computed on matched records. The independent unit follows each benchmark's released structure: individual images or records for CHAIR, POPE, AMBER, MIRAGE, and MathVista; original–masked pairs for POPEv2; paired-question image groups for MME; and records within mode for VHTest. POPE retains its corpus and negative-protocol strata, and AMBER retains its evaluation partitions. Thus every comparison preserves both sample identity and the grouping on which the official metric is defined.

A.2 Model Backbones

All models are loaded in evaluation mode with bfloat16 weights, `trust_remote_code=True`, their released tokenizer and processor, and device placement on one GPU. SPLIT discovers decoder blocks through the model interface and registers a hook when $H_q d_h = d$; other projections retain their native path. The cached selector operates on output coordinates of the language decoder, so GQA changes KV storage but not the H_q-way pre-W_O population analyzed by SPLIT.

R1-OneVision. R1-OneVision bridges visual perception and language reasoning through cross-modal formalization (Yang et al., 2025). Its pipeline converts visual content into formal textual representations, constructs multimodal reasoning annotations, and combines supervised fine-tuning with reinforcement learning. It is a central stress test because its explicit trajectories make unsupported intermediate premises visible while its Qwen2.5-VL attention stack exposes standard head-output projections.

Vision-R1. Vision-R1 builds a multimodal chain-of-thought cold-start set through modality bridging and data filtering, applies progressive thinking suppression, and then performs GRPO-style refinement (Huang et al., 2025). It tests whether SPLIT remains useful for a model already optimized to produce structured reasoning rather than short visual answers.

ThinkLite-VL. ThinkLite-VL obtains strong visual reasoning from a compact selected training set rather than knowledge distillation (Wang et al., 2026b). Monte Carlo tree search is used to identify difficult but solvable examples for reinforcement fine-tuning. It provides a distinct low-data self-improvement regime and is also used in the MIRAGE replication of Figure 3.

Qwen3-VL-8B-Thinking. Qwen3-VL supplies enhanced interleaved positional modeling, multi-level visual features, and thinking-oriented generation (Bai et al., 2025). Its dynamic-resolution visual layout and newer decoder implementation test the post-SDPA, pre-W_O interface beyond the Qwen2.5-VL checkpoints used in the motivation audit.

InternVL3.5-8B-HF. InternVL3.5 uses cascade reinforcement learning, with offline reinforcement learning for stable convergence followed by online refinement, and retains the Dynamic High Resolution strategy of the InternVL family (Wang et al., 2025b). We use the standard 8B Hugging Face checkpoint and its official thinking-mode system protocol; the separate InternVL3.5-Flash variant augments this architecture with the Visual Resolution Router. Its InternViT encoder, contiguous image-context representation, and multimodal integration pipeline test SPLIT beyond the Qwen-VL visual interfaces, while its Qwen3 language decoder keeps the pre-W_O control boundary directly comparable.

A.3 Benchmark Suite

The suite combines three evaluation forms. CHAIR and the generative partition of AMBER evaluate open descriptions; POPE, POPEv2, the discriminative partition of AMBER, MME, and VHTest test controlled decisions; MIRAGE and MathVista$_{\text{mini}}$ retain complete reasoning traces. We use 500 deterministically selected CHAIR images, every released record for the POPE, POPEv2, AMBER, MME, and VHTest protocols, all 1,329 MIRAGE questions, and all 1,000 MathVista testmini questions. No `max_samples` truncation is applied to the complete protocols.

Each benchmark uses its released query construction and the parsing and aggregation protocol specified in Appendix C. Model-facing prompts are shared record by record across methods, and each output is stored before parsing. CHAIR uses its released synonym normalizer; the binary suites use their released final-decision parsers; MIRAGE uses its released LLM-based final-answer parser; and MathVista uses its type-aware answer extractor and exact-match evaluator. Recall, balance, paired-question, and coverage measures accompany the headline scores.

CHAIR on MSCOCO. CHAIR parses object mentions from generated captions and compares them with synonym-normalized ground-truth COCO objects (Rohrbach et al., 2018; Lin et al., 2014). We report sentence- and instance-level hallucination together with object precision, recall, and

F1, which track object coverage alongside hallucination. CHAIR also supplies the event-aligned diagnostic analyses in Section 3.

POPE on COCO and A-OKVQA. POPE converts object hallucination into balanced binary decisions (Li et al., 2023). Random negatives sample absent categories uniformly, popular negatives emphasize frequent objects, and adversarial negatives emphasize objects that co-occur with visible categories. We instantiate all three protocols on COCO and A-OKVQA (Schwenk et al., 2022); the latter adds images selected for questions involving broader knowledge.

POPEv2. POPEv2 constructs counterfactual images by masking a target object and asks presence questions about original and edited views (Li et al., 2026c). The true-negative rate measures rejection of absent targets, while prediction-balance offset detects methods that obtain apparent gains by shifting all answers toward one class.

AMBER. AMBER provides both free descriptions and binary questions covering object existence, attributes, relations, and cognition-oriented errors (Wang et al., 2023). We use the released query file and lexical object-recognition pipeline, then apply the response-level generative aggregation defined in the paper. The combination of generative hallucination and discriminative accuracy tracks both grounding and answer coverage.

MME. MME contains fourteen perception and cognition subtasks, each organized around paired binary questions for an image (Fu et al., 2023). Its Accuracy+ term requires both paired questions to be correct and therefore penalizes unconditional yes or no behavior. We report the official perception, cognition, and total scores.

VHTest. VHTest evaluates counting, shape, color, orientation, size, position, OCR, and existence hallucinations (Huang et al., 2024b). We use its YNQ protocol and average accuracy across the eight modes rather than weighting modes by their record counts.

MIRAGE. MIRAGE isolates reasoning-induced hallucination by retaining questions whose visual content is correctly perceived (Dong et al., 2026). We evaluate all 1,329 released questions and apply the official LLM-based final-answer parser and task-specific normalization. Model generations, parser prompts, and aggregation follow the released protocol and are shared across methods.

MathVista$_{\text{mini}}$. MathVista evaluates mathematical reasoning grounded in charts, diagrams, geometric figures, documents, and natural images (Lu et al., 2024). We use the official testmini questions and type-aware answer normalization. When metadata are available, we additionally retain question-, answer-, visual-skill-, and reasoning-skill subgroup scores.

A.4 Compared Methods and Compatibility

Vanilla decoding. The unmodified backbone provides the paired control for every model–benchmark configuration. Given the vocabulary logits $z_t \in \mathbb{R}^{|\mathcal{V}|}$ at step t, it applies the backbone's native output distribution and greedy decision rule,

$$p_t = \mathrm{softmax}(z_t), \qquad y_t = \arg\max_{v \in \mathcal{V}} z_t(v). \tag{17}$$

It shares all generation settings with SPLIT and uses no hook. A SPLIT run with $\rho = 0$ is separately checked for tensor-level identity but is not treated as another numerical baseline.

LEAD. Latent Entropy-Aware Decoding preserves dense semantic alternatives at high-entropy reasoning transitions rather than immediately committing to a single discrete token (Xu et al., 2026b). Let $e(v)$ denote the input embedding of token v and let $r_t \sim p_t$ be LEAD's sampled token. LEAD measures predictive uncertainty and feeds back either the sampled discrete embedding or the

probability-weighted latent embedding,

$$H_t = -\sum_{v\in\mathcal{V}} p_t(v)\log p_t(v), \quad \overline{e}_t = \sum_{v\in\mathcal{V}} p_t(v)e(v),$$
$$\widetilde{e}_t = \begin{cases} e(r_t), & H_t < \widehat{H}, \\ \overline{e}_t, & H_t \geq \widehat{H}, \end{cases} \tag{18}$$

where $\widehat{H}$ is initialized for the current mode and reset to the observed entropy after a transition. At the first latent step of a high-entropy phase, the dense state is additionally tied back to the visual evidence through the proposed convex anchor,

$$\widetilde{e}_{t^\star} = (1-\lambda)\overline{e}_{t^\star} + \lambda e_{\text{vis}}. \tag{19}$$

The persistence counter and switch budget keep the decoder from oscillating between latent and discrete modes. We use the settings supported by the paper's reported ablations: visual-anchor strength $\lambda = 0.4$, discrete-to-latent persistence window $W_{D\to L} = 128$, dynamic entropy-threshold updates, and at most five mode switches. The discrete branch samples $r_t \sim p_t$ with seed 42 as defined above; unspecified sampling controls retain their released defaults.

MCoT. MCoT estimates tokenwise visual entropy from the vocabulary projections of visual-token hidden states (Ma et al., 2026). Let $p_{v,i}(u)$ be the softmax probability of vocabulary item u obtained by applying the language head to visual-token state h_i. For N_v visual positions, the score used by MCoT is

$$p_{v,i}(u) = \frac{\exp(h_i^\top w_u)}{\sum_{r\in\mathcal{V}}\exp(h_i^\top w_r)},$$
$$E(u,v) = -\frac{1}{\log N_v}\sum_{i=1}^{N_v} p_{v,i}(u)\log p_{v,i}(u), \tag{20}$$

where w_u is the LM-head row for u. The softmax is evaluated over the vocabulary independently at each visual position, and the selected probabilities retain this normalization. Writing $u_t^\star = \arg\max_u z_t(u)$, MCoT activates the entropy penalty when the currently preferred token has high visual entropy:

$$\widetilde{z}_t(u) = \begin{cases} z_t(u) - \alpha E(u,v), & E(u_t^\star, v) > \gamma, \\ z_t(u), & E(u_t^\star, v) \leq \gamma. \end{cases} \tag{21}$$

This is equivalent to multiplying the native token probability by $\exp[-\alpha E(u,v)]$ before renormalization. We use the reported penalty $\alpha = 0.75$, threshold $\gamma = 0.5$, and final visual-token hidden states. The vocabulary calculation uses chunks of 4,096 rows only to reduce memory; chunking is algebraically identical to the full computation.

RAL. RAL separates shallow perception-oriented heads from deeper reasoning-oriented heads using visual-attention ratios (Lu et al., 2026a). For attention matrix $A_{\ell h}$, query-token set $\mathcal{T}_q$, and visual-token set $\mathcal{T}_v$, we reproduce the modality statistic and the two head sets as

$$r_{\ell h} = \frac{1}{|\mathcal{T}_q|}\sum_{i\in\mathcal{T}_q}\sum_{j\in\mathcal{T}_v} A_{\ell h}[i,j],$$
$$\mathcal{H}^\ell_{\text{perc}} = \{h : \ell \leq \ell_{\text{perc}},\ r_{\ell h} \geq \tau_{\text{perc}}\},$$
$$\mathcal{H}^\ell_{\text{reas}} = \{h : \ell \geq \ell_{\text{reas}},\ r_{\ell h} \leq \tau_{\text{reas}}\}. \tag{22}$$

RAL then rescales each selected head before the frozen output projection:

$$g_{\ell h} = 1 + (g_{\text{perc}} - 1)\mathbf{1}[h \in \mathcal{H}^\ell_{\text{perc}}] + (g_{\text{reas}} - 1)\mathbf{1}[h \in \mathcal{H}^\ell_{\text{reas}}],$$
$$Y'_\ell = \text{Concat}(g_{\ell 1}O_{\ell 1}, \ldots, g_{\ell H}O_{\ell H})\, W_{O,\ell}. \tag{23}$$

We use its published R1-OneVision profile for all five reasoning backbones: reasoning layers start at 3 with threshold 0.01 and gain 1.30; perception layers end at 7 with threshold 0.30 and gain 1.20. The same profile is fixed across models and datasets before evaluation.

FLASH. FLASH first identifies vision-specialized heads by Fourier analysis of their spatial attention maps (Lu et al., 2026b). If $L_{r,\ell h}$ is the low-frequency energy concentration, $C_{\ell h}$ is orientation coherence, and $a_{\ell h}$ is aggregate visual attention, its spectral vortex score and eligibility rule can be written as

$$\begin{aligned} M_{\ell h} &= 20 \log_{10}(|\mathcal{F}(A_{\ell h})| + \epsilon)\,, \\ \mathrm{SVS}_{\ell h} &= L_{r,\ell h} + (1 - C_{\ell h})\,, \\ \mathcal{H}^{\ell}_{\mathrm{vis}} &= \mathrm{TopK}_k\{\mathrm{SVS}_{\ell h} : a_{\ell h} \geq \tau\}. \end{aligned} \tag{24}$$

For a selected head, let $C^m = \mathcal{D}(X^m)$ be the two-dimensional DCT of stream $m \in \{V, S\}$ and $d_{pq} = \sqrt{p^2 + q^2}$. FLASH uses complementary radial masks

$$\begin{aligned} M_v(p,q) &= 1 + \lambda_v\, \sigma\Big(d_{pq} - \alpha\sqrt{N}\Big)\,, \\ M_s(p,q) &= (1 - \lambda_s) + \lambda_s\, \sigma\Big(\alpha\sqrt{N} - d_{pq}\Big)\,, \end{aligned} \tag{25}$$

with the DC coefficients reset to one. The value spectrum is shaped in the log-amplitude domain, whereas the score spectrum is attenuated linearly:

$$\begin{aligned} V^\star &= \mathcal{R}\big(\mathcal{D}^{-1}\big[\mathrm{sgn}(C^V) \odot \exp\big(\log(|C^V| + \epsilon) \odot M_v\big)\big]\,; V\big)\,, \\ S^\star &= \mathcal{R}\big(\mathcal{D}^{-1}[C^S \odot M_s]; S\big)\,, \qquad A^\star = \mathrm{softmax}(S^\star)V^\star, \\ \mathcal{R}(U;X) &= U\frac{\|X\|_F}{\|U\|_F + \epsilon}. \end{aligned} \tag{26}$$

We transfer the published 7B profile to the five MLRMs: $\lambda_v = 1.2$, cutoff 0.5, attention threshold 0.2, and top five selected heads. In zero-based indexing, score and value shaping use the released layer ranges $[15, 26]$ and $[2, 31]$, respectively. We set $\lambda_s = 0.1$ for generative tasks and 0.9 for discriminative tasks, where only the first answer token is modulated as specified by the original protocol. An absolute layer endpoint beyond the decoder depth is clipped to the final available layer.

CRG. Causal Route Gating decomposes every attention head into visual and textual key–value routes (Cheng et al., 2026b). With column selectors S_{vis} and S_{txt}, the ordinary head output and its gated intervention are

$$\begin{aligned} O_{\mathrm{vis}} &= AS_{\mathrm{vis}}V, \qquad & O_{\mathrm{txt}} &= AS_{\mathrm{txt}}V, \\ O &= O_{\mathrm{vis}} + O_{\mathrm{txt}}, \quad & \widetilde{O}(g_{\mathrm{vis}}, g_{\mathrm{txt}}) &= g_{\mathrm{vis}}O_{\mathrm{vis}} + g_{\mathrm{txt}}O_{\mathrm{txt}}. \end{aligned} \tag{27}$$

For the decision logit $\ell(g_{\mathrm{vis}}, g_{\mathrm{txt}})$, exact route effects require counterfactual ablations. CRG replaces them with the first-order estimates obtained from one local gradient query,

$$\begin{aligned} \Delta_{\mathrm{vis}} &= \ell(1,1) - \ell(0,1), \qquad & \Delta_{\mathrm{txt}} &= \ell(1,1) - \ell(1,0), \\ G &= \frac{\partial \ell}{\partial \widetilde{O}}, & \widehat{\Delta}_{\mathrm{vis}} &= \langle G, O_{\mathrm{vis}} \rangle, & \widehat{\Delta}_{\mathrm{txt}} &= \langle G, O_{\mathrm{txt}} \rangle, \\ \widehat{\mathrm{VRI}} &= \frac{|\widehat{\Delta}_{\mathrm{vis}}|}{|\widehat{\Delta}_{\mathrm{vis}}| + |\widehat{\Delta}_{\mathrm{txt}}| + \epsilon}. \end{aligned} \tag{28}$$

The two conflict directions are separated before heads are ranked by visual reliance:

$$\begin{aligned} \mathcal{H}_A &= \{(\ell,h) : \widehat{\Delta}_{\mathrm{vis}} > 0,\ \widehat{\Delta}_{\mathrm{txt}} < 0\}, \\ \mathcal{H}_B &= \{(\ell,h) : \widehat{\Delta}_{\mathrm{vis}} < 0,\ \widehat{\Delta}_{\mathrm{txt}} > 0\}, \\ s_i &= \frac{i}{n-1}, \qquad g^{(i)}_{\mathrm{txt}} = g_{\min} + (g_{\max} - g_{\min})\,\mathrm{clip}(s_i^\gamma, \epsilon, 1-\epsilon), \qquad g^{(i)}_{\mathrm{vis}} = 1. \end{aligned} \tag{29}$$

The lowest-VRI heads in conflict set A receive mild text suppression ($g_{\mathrm{txt}} \in [0.5, 1]$), whereas those in set B receive stronger suppression ($g_{\mathrm{txt}} \in [0, 0.5]$), followed by the gated decode pass. We use profiles listed in the paper, writing each as $(\ell_{\min}, \ell_{\max}, k, \gamma)$: POPE random $(9, 17, 10, 0.4)$, popular $(9, 17, 9, 0.6)$, and adversarial $(9, 17, 11, 0.5)$; binary general-capability tasks $(10, 24, 10, 0.4)$; open generation $(10, 24, 8, 0.6)$; and AMBER generative and discriminative settings $(8, 19, 9, 0.6)$ and $(8, 19, 10, 0.7)$. The closest disclosed profile for each model family and task type is kept fixed across the evaluation. CRG and FLASH use eager attention because their equations require explicit attention tensors, while SPLIT keeps the native SDPA path. RAL, FLASH, and CRG otherwise retain their native or closest published configurations.

Published-configuration transfer. Baseline settings are taken from released code or the closest disclosed model–task profile and fixed before evaluation. We do not search layer ranges or gains separately on the five target backbones; SPLIT likewise uses one shared $\rho = 1.0$. Proportional depth rescaling is not specified by RAL or FLASH and would define an additional method variant. The comparison therefore measures direct transfer of each published control to new reasoning backbones rather than an oracle-tuned variant for every model and task.

A.5 SPLIT Configuration and Runtime Scope

SPLIT has one user-facing parameter, the nonnegative global strength ρ. Main experiments use the default $\rho = 1.0$ across all five models and eight benchmarks. For sensitivity analysis, we sweep $\rho \in \{0.5, 0.6, \ldots, 1.5\}$ on R1-OneVision-7B and Qwen3-VL-8B-Thinking with all other settings fixed; the main evaluation retains the same default across every model and dataset. Setting $\rho = 0$ is an exact identity path.

All other quantities are architecture-derived: sketch width $s = \min\{d_{\text{model}}, \max(8, 4H)\}$; history capacity H; three cold power iterations; one warm iteration; minimum history length three; layer scale $1/\sqrt{L}$; and machine-precision stabilizers. Coordinate indices are sampled once using seed $42 + 104729\ell$, sorted, and cached for layer ℓ. These coordinates remain fixed across tokens and examples for that loaded model, preserving a common temporal coordinate system. The seed rule is declared once and shared by every backbone and benchmark rather than selected from outcomes; conditional on the cached indices, SPLIT decoding is deterministic.

SPLIT registers a forward pre-hook on each compatible native W_O. The hook reads the last-query head tensor, stores a detached clean view, and applies at most one interpolation around the active anchor. Attention logits, RoPE, softmax, values, KV caching, and the native output projection retain their backbone implementations. Statistics for layers with the same device, dtype, head count, and head width are stacked and finalized in batched kernels between decoding steps. The controller uses detached forward activations and a single decoding branch.

Prefill is observation-only. Its final row is the causal boundary state that already conditions on the complete image and prompt, so it initializes the clean anchor and count without mixing visual-token, instruction-token, and boundary-query roles. No intervention occurs until both robust histories contain three prior values. Every subsequent clean generated observation enters the Welford update in Proposition 1; the boundary state is therefore an initializer, not a fixed reference. At every later step, statistics are computed from the detached clean observation saved before the current intervention, avoiding feedback contamination. A windowed or exponential anchor can replace the exact mean for unbounded streams, whereas the exact clean-mean identity is well matched to the 512-token evaluation horizon used here.

A.6 Complete SPLIT Algorithm

Algorithms 1–4 form a complete modular specification. Algorithm 1 owns the causal state and assembles the three controller blocks: Algorithm 2 constructs the residual-coordinate operator and tracks its mode, Algorithm 3 evaluates the failed-release gate, and Algorithm 4 converts that gate into energy-preserving head gains. Here $\text{tail}_m(a)$ retains the most recent m entries of a. Native prefill, cached decoding, and the task stop rule remain standard MLRM interfaces; Algorithm 1 states the hook order around $W_{O,\ell}$ explicitly. The cold initializer in Algorithm 2 is a deterministic, nonconstant, zero-mean ramp. It introduces no second random source, while ensuring a generic direction with nonzero alignment except for the explicit orthogonality case covered by the cold-start condition in Appendix D.4.

Algorithm 1 SPLIT decoding with lagged release control

Require: Image–prompt pair (I, P); frozen L-layer MLRM f_θ; strength $\rho \geq 0$; seed ξ; horizon T
Ensure: Generated sequence y

for $\ell = 1, \ldots, L$ **do**
  $s_\ell \leftarrow \min\{d_{\text{model}}, \max(8, 4H_\ell)\}$
  $\mathcal{J}_\ell \leftarrow \text{sort}(\text{randperm}_{\xi+104729\ell}(d_{\text{model}})_{1:s_\ell})$
  $S_\ell \leftarrow I_{d_{\text{model}}}[\mathcal{J}_\ell, :]$; $(M_\ell, n_\ell, u_\ell, \mathcal{H}_{c,\ell}, \mathcal{H}_{v,\ell}, g_\ell) \leftarrow (\perp, 0, \perp, \varnothing, \varnothing, \mathbf{1})$
end for
Native prefill (I, P); cache $\mathcal{K}$, save clean $\{X_{\ell,0}\}_{\ell=1}^{L}$, and set $y \leftarrow \varnothing$
for $t = 1, \ldots, T$ **do**
  for $\ell = 1, \ldots, L$ **do**
    if $M_\ell = \perp$ **then**
      $(M_\ell, n_\ell, g_\ell) \leftarrow (X_{\ell,t-1}, 1, \mathbf{1})$
    else
      $R_\ell \leftarrow X_{\ell,t-1} - M_\ell$
      $(\text{valid}_\ell, u_\ell, c_\ell, v_\ell, E_\ell) \leftarrow$ TRACKMODE$(R_\ell, S_\ell, W_{O,\ell}, u_\ell)$ {Algorithm 2}
      if $\text{valid}_\ell = 0$ **then**
        $g_\ell \leftarrow \mathbf{1}$
      else
        $\alpha_\ell \leftarrow$ DETECT$(c_\ell, v_\ell, \mathcal{H}_{c,\ell}, \mathcal{H}_{v,\ell}, H_\ell, L, \rho)$ {Algorithm 3}
        $g_\ell \leftarrow$ RELEASE$(E_\ell, u_\ell, \alpha_\ell)$ {Algorithm 4}
        $\mathcal{H}_{c,\ell} \leftarrow \text{tail}_{H_\ell}(\mathcal{H}_{c,\ell} \,\|\, c_\ell)$; $\mathcal{H}_{v,\ell} \leftarrow \text{tail}_{H_\ell}(\mathcal{H}_{v,\ell} \,\|\, v_\ell)$
      end if
      $n_\ell \leftarrow n_\ell + 1$; $M_\ell \leftarrow M_\ell + (X_{\ell,t-1} - M_\ell)/n_\ell$
    end if
  end for
  Native cached step gives $(y_t, \mathcal{K})$; each pre-hook saves clean $X_{\ell,t}$ before sending $M_\ell + g_\ell \odot (X_{\ell,t} - M_\ell)$ to $W_{O,\ell}$
  $y \leftarrow y \,\|\, y_t$
  if y_t satisfies EOS or the task stop rule **then**
    break
  end if
end for
return y

Algorithm 2 Residual-coordinate mode construction and tracking

Require: Innovation $R \in \mathbb{R}^{H \times d_h}$; selector $S \in \{0,1\}^{s \times d_{\text{model}}}$; blocks $\{W_O^{(h)}\}_{h=1}^{H}$; previous mode u^-
Ensure: Validity flag valid; current mode u; leading share c; innovation energy v; head energies E

$E_h \leftarrow \|R_{h,:}\|_2^2$; $Z_{h,:}^\top \leftarrow S W_O^{(h)} R_{h,:}^\top$; $\widetilde{Z} \leftarrow C_H Z$
if $\mathbf{1}^\top E \leq \epsilon$ **or** $\|\widetilde{Z}\|_F \leq \epsilon$ **then**
  return $(0, u^-, \perp, \perp, E)$
end if
$Y \leftarrow \mathcal{N}(\widetilde{Z})$; $G \leftarrow YY^\top$
if $u^- = \perp$ **then**
  $u_h \leftarrow (2h - H - 1)/\|(2j - H - 1)_{j=1}^{H}\|_2$
  for $k = 1, 2, 3$ **do**
    $u \leftarrow Gu / \max\{\|Gu\|_2, \epsilon\}$
  end for
else
  $u \leftarrow Gu^- / \max\{\|Gu^-\|_2, \epsilon\}$
  $u \leftarrow (1 - 2\mathbf{1}\{\langle u, u^- \rangle < 0\})u$
end if
$c \leftarrow u^\top G u / \max\{\text{tr}(G), \epsilon\}$; $v \leftarrow \|\widetilde{Z}\|_F^2/(Hs)$
return $(1, u, c, v, E)$

Algorithm 3 Robust failed-release detection

Require: (c, v); histories $(\mathcal{H}_c, \mathcal{H}_v)$; head count H; depth L; strength ρ
Ensure: Causal release amplitude $\alpha \in [0, 1]$
1: **if** $|\mathcal{H}_c| < 3$ **or** $|\mathcal{H}_v| < 3$ **then**
2: **return** 0
3: **end if**
4: $\theta \leftarrow 1/H$; $b_c \leftarrow \mathrm{median}(\mathcal{H}_c)$
5: $s_c \leftarrow \mathrm{MAD}(\mathcal{H}_c) \vee (b_c - \theta)_+/H \vee \epsilon$; $q_c \leftarrow \theta \vee (b_c - s_c)$
6: $r \leftarrow \mathrm{clip}_{[0,1]}((c - q_c)/s_c)$; $q \leftarrow \mathrm{clip}_{[0,1]}((c - \theta)_+/\max\{b_c - \theta, \epsilon\})$
7: $b_v \leftarrow \mathrm{median}(\mathcal{H}_v)$; $s_v \leftarrow \mathrm{MAD}(\mathcal{H}_v) \vee |b_v|/H \vee \epsilon$
8: $e \leftarrow \mathrm{clip}_{[0,1]}((v - b_v - s_v)/s_v)$
9: **return** $\mathrm{clip}_{[0,1]}((\rho/\sqrt{L})rqe)$

Algorithm 4 Energy-constrained carrier redistribution

Require: Head energies $E \in \mathbb{R}^H_+$; tracked mode $u \in \mathbb{R}^H$; amplitude $\alpha \in [0, 1]$
Ensure: Energy-preserving head gain $g \in \mathbb{R}^H$
1: **if** $\alpha = 0$ **or** $\mathbf{1}^\top E \le \epsilon$ **then**
2: **return** $\mathbf{1}$
3: **end if**
4: $p \leftarrow (u \odot u)/\max\{\|u\|_2^2, \epsilon\}$; $w \leftarrow E/\max\{\mathbf{1}^\top E, \epsilon\}$; $\bar{p}_w \leftarrow w^\top p$
5: $d \leftarrow (\bar{p}_w \mathbf{1} - p)/\max\{\|\bar{p}_w \mathbf{1} - p\|_\infty, \epsilon\}$
6: $\widetilde{g} \leftarrow \mathbf{1} + \alpha d$; $\beta \leftarrow \sqrt{\mathbf{1}^\top E/\max\{E^\top(\widetilde{g} \odot \widetilde{g}), \epsilon\}}$
7: **return** $\beta\widetilde{g}$

An individual zero row after centering is normalized with the ϵ floor and contributes no direction; zero total or centered innovation energy triggers Algorithm 1's explicit identity path. An incompatible output projection likewise bypasses the controller when selectors are registered. Eigenvector signs are aligned only for warm-start continuity, while gains depend on squared loadings and remain sign-invariant. The implementation batches compatible layers without changing the assembled state transitions.

A.7 Complexity and Persistent State

For a layer with H heads, head width d_h, and sketch width s, the selected-row projection costs $O(Hsd_h) = O(sd_{\mathrm{model}})$, centering and normalization cost $O(Hs)$, the Gram operator costs $O(H^2 s)$, and one warm power update costs $O(H^2)$. The native dense output projection costs $O(d^2_{\mathrm{model}})$, so the leading sketch-to-native arithmetic ratio is approximately s/d_{model} before the small $H \times H$ operations. Cold initialization adds two extra power multiplications only for the first valid operator.

Persistent state consists of clean anchors $O(Ld_{\mathrm{model}})$, two capacity-H histories $O(LH)$, modes and gains $O(LH)$, and cached coordinate indices $O(Ls)$. Selected rows of W_O are cached as views or materialized blocks according to the backend. The per-token diagnostic workspace $O(L(Hs + H^2))$ is reusable. KV-cache storage and context length remain unchanged.

The hook performs collection and an optional elementwise interpolation. Cross-layer sketches and statistics are finalized between model calls in place of a separate spectral pipeline inside every attention layer. Realized latency also depends on memory traffic, kernel fusion, and the framework's pre-hook path, which are captured by the wall-clock measurements.

A.8 Hardware, Software, and Reproducibility

All experiments use an Ubuntu 24.04.3 LTS server with an Intel Xeon Silver 4314 CPU at 2.40 GHz, 440 GB system RAM, and eight NVIDIA RTX 3090 GPUs with 24 GB VRAM each. One GPU is assigned to each run; the remaining devices are used only for concurrent independent jobs, without tensor or model parallelism. The core environment uses Python 3.10, PyTorch 2.5.1+cu121, CUDA

12.1, and cuDNN 9.1.0. The resolved Transformers and processor versions and the checkpoint configuration are stored with each run because Qwen2.5-VL, Qwen3-VL, and InternVL3.5 require different released model interfaces.

Deterministic CuDNN execution is enabled. Seed 42 is fixed in Python, PyTorch, CUDA, the decoding generator, and SPLIT's coordinate selector. Each deterministic table entry is one complete paired run of a model, method, and benchmark configuration. LEAD retains the stochastic discrete phase specified by its released procedure and uses the same declared seed and sample order for the matched main comparison. CPU work is limited to image decoding, dataset loading, parsing, and metric aggregation and is excluded from GPU throughput measurements.

Every output record stores the checkpoint, resolved model geometry, method name, SPLIT strength or baseline profile, prompt hash, sample identifier, seed, dtype, attention backend, generation length, extracted answer, and optional controller diagnostics. SPLIT diagnostics include the leading share, isotropic reference, release baseline and target, innovation baseline and scale, detector factors, realized α, gain range, and active layer count. Together, these fields reconstruct each active gate from its detector state and link it to the conjunction in Section 3.3.

Statistical unit and complementary metrics. Aggregate scores are recomputed from the complete matched prediction sets under the independent units defined above; benchmark strata, partitions, categories, and modes retain their released proportions. For binary tasks, prediction balance, recall, TNR, or Accuracy+ accompanies accuracy, while CHAIR object F1 and AMBER coverage track retained content. These paired and complementary views separate genuine error reduction from gains obtained by suppressing valid objects or affirmative predictions.

Run-level invariants. The manifest makes the paired comparison directly auditable. Sample identifiers and prompt hashes establish that every method receives the same image, rendered query, generation ceiling, stop rule, and evaluator. Fresh controller state, history length, active-layer count, and first active token verify record isolation and the observation-only prefill. Selector seeds, sketch widths, and coordinate hashes establish that the sketch depends only on model geometry, layer index, and the declared seed.

Two numerical checks accompany every SPLIT run. First, $\rho = 0$ and every degenerate detector path are compared against the native tensor and must satisfy the identity tolerance. Second, each active gate records its innovation energy before and after intervention, energy residual, and gain RMS, directly checking the conservation law. Backend, forward-count, gradient-count, and projection-count fields distinguish native-SDPA SPLIT runs from attention-explicit baselines. Finally, parse coverage, EOS and truncation rates, output lengths, synchronized latency, token throughput, and peak allocated and reserved memory are derived from the same stored generations.

Repeated direction across five backbones and heterogeneous benchmarks supplies the aggregate empirical evidence. CHAIR object F1, balance statistics from POPE and POPEv2, AMBER coverage, MME, and VHTest track whether lower hallucination comes from suppressing valid content or affirmative predictions. The evaluation separates aggregate outcomes, matched event dynamics, sensitivity curves, and deployment measurements from the exact algebraic guarantees. The release artifact contains the controller, baseline adapters, preprocessing, evaluation, run manifests, and analysis code with the equation and algorithm references retained in source comments.

B Additional Related Work

The main Related Work isolates the three distinctions that motivate SPLIT. This appendix broadens the literature map to adjacent methods that share an empirical objective but intervene on a different computational object. We organize prior work by the locus of the failure and the locus of the correction. Methods that improve the same hallucination metric can still make different assumptions about evidence, decoder state, and the attention backend.

B.1 Multimodal Reasoning and Trace-Level Hallucination

Modern multimodal reasoning systems acquire explicit long-form deliberation through supervised trajectories, reinforcement learning, outcome or process rewards, and combinations of these ingredi-

ents. R1-OneVision couples cross-modal formalization with supervised and reinforcement stages; Vision-R1 constructs a multimodal cold start and progressively refines reasoning behavior; and ThinkLite-VL uses a compact, difficulty-selected training set for reinforcement fine-tuning (Yang et al., 2025; Huang et al., 2025; Wang et al., 2026b). Active-perception models extend this recipe by allowing the chain to request new visual evidence. VLM-R3 learns when and where region evidence should be acquired, DeepEyes elicits interleaved text and image-based reasoning with learned zooming, and Perception-R1 introduces a perception-specific reward to address the sparsity of answer-only supervision (Jiang et al., 2026; Zheng et al., 2025; Xiao et al., 2025). These advances make the reasoning trace both more capable and more observable.

Observability exposes a distinction that answer-level hallucination benchmarks alone cannot capture. MIRAGE filters for questions whose visual content is correctly perceived and thereby isolates reasoning-induced hallucination across the subsequent chain, while More Thinking, Less Seeing reports that extended reasoning can amplify hallucination and weaken effective use of visual evidence (Dong et al., 2026; Xu et al., 2026a). MCoT locates reasoning stages whose visual states diverge from the image-grounded trajectory; VRGA associates extended chains with dispersion away from question-relevant regions; and C3PO argues that visually diluted, redundant reasoning tokens invite language-prior reliance (Ma et al., 2026; Peng et al., 2026; Fang et al., 2026). See It, Say It, Sorted further focuses on error propagation: an unsupported intermediate premise can become context for every later deduction (Zhang et al., 2026f). These studies motivate an event-level target while leaving the triggering internal statistic open when the evidence and generated prefix are held fixed.

Recent work sharpens this event-level view around transitions in the chain. Cognitive Pivot Points and Visual Anchoring associates high-entropy bifurcation points with a failure to query visual evidence in intermediate layers, then trains hierarchical attention reinforcement and forced reflection around those points (Qian et al., 2026). SDR-MCoT likewise uses entropy to decide whether a query warrants detailed reasoning before applying stepwise visual enhancement, thereby coupling reliability with reduced overthinking (Tu et al., 2026). These findings support adaptive intervention, but their triggers concern uncertainty or learned visual-attention behavior. SPLIT instead tests a contemporaneous change in the geometry of realized head contributions, so uncertainty and spectral concentration remain empirically separable hypotheses rather than interchangeable names for the same event.

B.2 Evidence Revision and Active Perception

A natural response to weak grounding is to change the evidence available to the next reasoning step. VisRef selects a relevant yet diverse coreset of visual tokens and re-injects it during reasoning, whereas Look-Back learns when to refocus without an explicit external crop operation (Ghosal et al., 2026; Yang et al., 2026). SPARC decouples perception from reasoning so that localization and cropping can scale independently, MIRROR trains iterative visual-region reflection, and VLM-R3 learns both the timing and target of additional region acquisition (Avogaro et al., 2026; Zhang et al., 2026d; Jiang et al., 2026). EVRI uses token entropy to trigger renewed perception, while See It, Say It, Sorted maintains a test-time pool of verified visual evidence for successive reasoning steps (Liufu et al., 2026; Zhang et al., 2026f). The common principle is to repair the chain by improving its external or re-encoded premises.

Other approaches reorganize the trace rather than acquire new pixels. ClueTracer follows the question-to-output-to-vision path to recover overlooked clues, while V-Skip and C3PO shorten or select reasoning tokens so that visual anchors are not overwhelmed by redundant language context (Xi et al., 2026; Zhang et al., 2026b; Fang et al., 2026). Such mechanisms are especially appropriate when the relevant region was never resolved or the trace itself is needlessly long. SPLIT addresses the complementary setting in which the current visual encoding and causal prefix already contain the available evidence, and tests whether their processing has entered a spectrally concentrated head-population regime.

The same objective can be pursued through agents or training. DRP assigns strategic reasoning to an LLM and answers its fine-grained visual queries with a separate multimodal observer, explicitly decoupling perception from cognition at test time (Jia et al., 2026). Reflection-V constructs vision-centered cold-start trajectories through interaction between a visual model and a reasoning model, then rewards sustained attention to visual tokens during reinforcement learning (Jian et al., 2025). Cognitive Pivot Points instead internalizes visual anchoring around uncertain transitions through a

hierarchical visual-attention reward (Qian et al., 2026). These approaches can teach or orchestrate the model to look again; SPLIT addresses the narrower deployment setting in which the backbone, prompt, image encoding, and generated prefix are all left unchanged.

B.3 Decoding-, Attention-, and Route-Level Mitigation

Contrastive decoding constructs a corrective token distribution by comparing a clean branch with a deliberately degraded image, instruction, or context. VCD and ICD established influential variants of this training-free strategy; OPERA penalizes over-trusted summary tokens and can roll back generation; and SID masks context-dependent visual regions to obtain an internal negative branch (Leng et al., 2024; Wang et al., 2024; Huang et al., 2024a; Huo et al., 2024). Later work makes the negative branch more selective. DEGAP identifies perturbation points from uncertainty, DVR separates semantic alignment from structural degradation with two visual encoders, and ACD and ASCD move part of the correction into score- or head-aware attention control (Seo et al., 2026; Chen et al., 2026d;c; Wang et al., 2026c). These methods are powerful when a useful counterfactual distribution can be constructed, but that construction may require additional forward computation or explicit score maps.

The contrastive family has consequently diversified the source and fusion of its counterfactuals. MPI-CD contrasts original, salient-region, and nonsalient-region image branches and augments them with hidden-layer memory recall (Ruan et al., 2025). CoFi-Dec generates coarse- and fine-conditioned visual hypotheses and fuses their predictive distributions with a Wasserstein barycentric objective (Cao et al., 2025). VCGD instead introduces a separately trained caption model to supply visual clues and constrains their influence by image confidence (Chen et al., 2026a). These designs enrich corrective evidence through auxiliary image paths, generated hypotheses, or an external model. Residual Decoding takes a lighter output-space route by combining current logits with their historical evolution to counter language-prior bias (Chen et al., 2026b). SPLIT constructs its counterfactual within the already-computed head population, redistributing its current carrier load before projection.

Attention-based methods differ in what they regard as the relevant route. AdaIAT increases attention to generated text; inter-layer discrepancy methods compare visual attention across depth; and gaze- or head-dispersion analyses diagnose shifts away from relevant visual regions (Zhong et al., 2026; Xie et al., 2026; Qi et al., 2025; Li et al., 2026b). Fine-grained causal tracing identifies middle-layer self-attention as a locus of cross-modal aggregation, while causally grounded dual-path control jointly regulates visual and textual paths (Li et al., 2026a; Yu et al., 2026b). Token-Asymmetric Filtering separates disproportionately influential phantom text tokens from underused visual anchors; VisFlow applies token- and head-level attention edits; and Vision-Guided Attention derives grounding from visual-token semantics (Ouyang et al., 2026; Tang et al., 2026; Zhao et al., 2025). Complementary work studies pathological mediators, patch-level grounding, preliminary attention-score detectors, vision-token manipulation, and gradient-based self-reflection (Yu et al., 2026a; Nguyen et al., 2026; Hoang-Xuan et al., 2025; Fa et al., 2026; Wang et al., 2025a). RAL assigns functional roles from attention ratios and applies layer-dependent head gains; FLASH edits score and value spectra; CRG suppresses causally conflicting text routes; and LAD denoises raw attention logits with a Langevin-inspired update (Lu et al., 2026a;b; Cheng et al., 2026b; Luo et al., 2026). Their shared strength is fine internal selectivity, although their required computational objects differ substantially.

That variation determines backend compatibility. A stock fused SDPA operator accepts queries, keys, values, and a mask, then returns their aggregated output; probability matrices and raw scores are not exposed as output tensors. Queries, keys, and values remain available to the caller, making value-side transformations compatible with the fused operator. Interventions that consume materialized probabilities or scores, perform score-dependent surgery, or require route-resolved products and gradients instead use eager attention, additional kernels, or a custom fused implementation. This group includes map-dependent methods such as OPERA, SID, AdaIAT, inter-layer discrepancy, fine-grained grounding, TAF, and VisFlow; ACD, which contrasts normalized attention maps; raw-logit editors such as ASCD and LAD; and the published formulations of RAL, FLASH, CRG, and dual-path attention reweighting (Huang et al., 2024a; Huo et al., 2024; Zhong et al., 2026; Xie et al., 2026; Nguyen et al., 2026; Ouyang et al., 2026; Tang et al., 2026; Chen et al., 2026c; Wang et al., 2026c; Lu et al., 2026a;b; Cheng et al., 2026b; Yu et al., 2026b). VGA provides an output-side alternative: in its notation, $(\alpha + \beta G)V = \alpha V + \beta GV$ avoids materializing attention weights and remains compatible with FlashAttention (Zhao et al., 2025). Output-distribution and post-attention-state controls can

likewise retain fused attention, although contrastive variants may add branches. SPLIT reads and gates the concatenated per-head result at the input of W_O, after stock SDPA has completed.

B.4 Latent and Residual-Space Intervention

Latent-state methods avoid treating the final token distribution as the only useful control surface. LEAD detects uncertain reasoning phases and switches between discrete and latent reasoning while injecting a visual guidance vector (Xu et al., 2026b). AFTER uses factual activation directions, RFI transports states with rectified flows, HulluEdit estimates an online visual-evidence subspace, and REVIS performs sparse latent steering (Wang et al., 2026a; Cheng et al., 2026a; Lin et al., 2026; Wu et al., 2026). PTI edits modality-specific key–value components once during prefill, whereas RUDDER derives a persistent visual anchor from prefill residual updates and gates its injection per token (Zhang et al., 2026a; Zou et al., 2025). Token-level visual-sensitivity steering and geometry-aware projection against language priors provide two further adaptive correction directions (Zhang et al., 2026e; Zhuang et al., 2026). Causal layer localization, counterfactual low-rank subspaces, and gradient-based influence correction emphasize that a useful edit should be localized rather than uniformly amplified (Dang et al., 2026; Dastmalchi et al., 2026; Wang et al., 2025a).

These studies establish that decoder activations contain controllable factual and visual structure. Most learn, extract, or otherwise define a persistent direction, subspace, layer set, or mode of transition. SPLIT recomputes the release direction from the current token's head population and retains it only for the adjacent causal update. The distinction is between steering toward a preidentified semantic direction and reducing excessive instantaneous exposure to an endogenous coordination mode. Robust local histories make the latter control rule self-calibrating across layers and models.

Head-output methods are particularly close in computational object but different in supervision and causality. VIB-Probe stacks pre-projection attention-head outputs across layers, trains a variational information-bottleneck detector on hallucination labels, attributes its risk score to heads by gradients, and reruns risky tokens with selected heads suppressed (Zhang et al., 2026c). Fine-grained causal tracing corrupts images and patches clean activations back into particular token–component–layer locations to estimate recovery, then reinforces object information at the identified sites (Li et al., 2026a). Both demonstrate that head outputs and component-local interventions are informative beyond raw attention. SPLIT derives detection and release from an online population statistic and applies the resulting gate on the next causal step of the same decoding trajectory.

B.5 Attribution, Spectral Structure, and Online Control

Mechanistic attribution distinguishes raw attention coefficients from realized contribution. Attention rollout and flow account for mixing across layers, transformer relevance propagation incorporates values and residual pathways, and mechanistic transformer analyses make the output projection and residual stream explicit (Abnar & Zuidema, 2020; Sun & Marasović, 2021; Chefer et al., 2021b;a; Elhage et al., 2021). Practical multimodal attribution reaches the same conclusion: the relevant object depends on how attention, values, projections, and later layers compose (Rai et al., 2024). SPLIT therefore defines each head by its realized contribution in post-W_O residual coordinates and subtracts a clean temporal anchor before estimating coordination. Head identity is represented by its token-local loading in the population mode, allowing that role to change across prompts and generation steps.

Spectral analyses of attention show that dominant modes and eigengaps can govern signal propagation and rank collapse. Deep-attention theory studies collapse across depth, random-matrix analyses characterize outlier-driven rank or entropy behavior, and recent initialization results relate width and depth scaling to stable propagation (Noci et al., 2022; Nait Saada et al., 2025; Giorlandino & Goldt, 2026). These results concern token representations, initialization, or asymptotic regimes; a large leading mode is therefore not, by itself, evidence of hallucination in a pretrained MLRM. SPLIT uses *spectral herding* for a different, operationally testable regime: a mobile head population remains collectively concentrated in residual coordinates. Its operator is constructed over headwise temporal innovations, row energy is normalized, and a within-row coordinate permutation supplies a null that preserves marginal energy while destroying cross-head coordination. The matched analysis of factual and hallucinated events then asks not whether herding exists, but whether it fails to release when innovation rises.

Recent empirical spectral work provides an adjacent but distinct precedent. Geometry of Reason treats token-to-token attention matrices as dynamic graphs and uses graph connectivity, smoothness, high-frequency energy, and spectral entropy to distinguish valid from invalid mathematical reasoning (Noël, 2026). Its object is a token graph and its goal is proof-validity classification. SPLIT instead constructs an operator over headwise residual innovations in a multimodal decoder, compares concentration with a coordination-breaking null, and uses the resulting event strength to control the next causal step. The two views therefore support the broad premise that reasoning validity can leave spectral signatures, while differing in graph, supervision, task, and intervention.

Efficient deployment draws on a separate line of numerical and statistical work. Randomized subspace embeddings make a high-dimensional covariance accessible from a compact coordinate sketch; high-dimensional covariance control and eigenvector perturbation quantify when the leading structure is preserved; and streaming PCA analyzes the tracking of a drifting dominant subspace (Martinsson & Tropp, 2020; Vershynin, 2026; Greenbaum et al., 2020; Bhardwaj & Vu, 2024; Bienstock et al., 2022). SPLIT specializes these ingredients to the small head-population operator, where the effective rank is bounded by the number of heads rather than decoder width. Its release rule is then naturally expressed as a constrained steepest-descent step under an energy-weighted metric, followed by retraction to the feasible sphere (Amari, 2016; Martens, 2020; Boumal, 2023). Appendix D separates the probabilistic conditions needed for sketch fidelity and temporal tracking from the identity, permutation, conservation, and budget properties that hold exactly for the implemented controller.

C Metric Definitions

This appendix defines every metric named in Section 4.1. Rates are written on the unit scale below and may be multiplied by 100 in tables. The equations follow the published metric definitions and state the aggregation used in our evaluation, together with the direction of improvement and the auxiliary quantities that track answer coverage or class balance.

C.1 CHAIR and Object Coverage

For image n, let $O_{\mathrm{gt}}^{(n)}$ be the synonym-normalized set of ground-truth COCO objects and $O_{\mathrm{cap}}^{(n)}$ the multiset of object mentions extracted from the generated caption (Rohrbach et al., 2018). The hallucinated multiset is

$$O_{\mathrm{hall}}^{(n)} = \{o \in O_{\mathrm{cap}}^{(n)} : o \notin O_{\mathrm{gt}}^{(n)}\}. \tag{30}$$

Sentence-level and instance-level CHAIR are

$$\mathrm{CHAIR}_S = \frac{1}{N}\sum_{n=1}^{N} \mathbf{1}[|O_{\mathrm{hall}}^{(n)}| > 0], \tag{31}$$

$$\mathrm{CHAIR}_I = \frac{\sum_n |O_{\mathrm{hall}}^{(n)}|}{\sum_n |O_{\mathrm{cap}}^{(n)}|}. \tag{32}$$

Lower is better for both. To expose a method that lowers CHAIR by mentioning fewer objects, let $c_n(o)$ be the number of normalized mentions of object o in caption n and define

$$\mathrm{Prec}_{\mathrm{obj}} = \frac{\sum_n \sum_{o \in O_{\mathrm{gt}}^{(n)}} c_n(o)}{\sum_n \sum_o c_n(o)}, \tag{33}$$

$$\mathrm{Rec}_{\mathrm{obj}} = \frac{\sum_n \sum_{o \in O_{\mathrm{gt}}^{(n)}} \mathbf{1}[c_n(o) > 0]}{\sum_n |O_{\mathrm{gt}}^{(n)}|}, \tag{34}$$

$$\mathrm{F1}_{\mathrm{obj}} = \frac{2\,\mathrm{Prec}_{\mathrm{obj}}\mathrm{Rec}_{\mathrm{obj}}}{\mathrm{Prec}_{\mathrm{obj}} + \mathrm{Rec}_{\mathrm{obj}}}. \tag{35}$$

We additionally log generated length and the number of recognized object mentions as descriptive controls, but neither is part of CHAIR.

C.2 POPE

POPE treats object presence as binary classification (Li et al., 2023). Let $y_i \in \{0, 1\}$ be the label and $\hat{y}_i$ the parsed answer. From true positives, false positives, true negatives, and false negatives,

$$\mathrm{Acc} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}, \tag{36}$$

$$\mathrm{Prec} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \qquad \mathrm{Rec} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{37}$$

$$\mathrm{F1} = \frac{2\,\mathrm{Prec}\,\mathrm{Rec}}{\mathrm{Prec} + \mathrm{Rec}}. \tag{38}$$

Metrics are computed separately for random, popular, and adversarial splits. For corpus $d \in \{\mathrm{COCO}, \mathrm{AOKVQA}\}$, a split average is

$$\overline{m}_d = \frac{1}{3} \sum_{s \in \{\mathrm{random}, \mathrm{popular}, \mathrm{adversarial}\}} m_{d,s}. \tag{39}$$

COCO and A-OKVQA are reported separately; a cross-corpus average is used only when explicitly labeled. Table 5 reports every split-level accuracy and F1 value underlying the corpus means in Table 1.

C.3 POPEv2

POPEv2 retains the binary metrics above but emphasizes counterfactual negatives created by removing target objects (Li et al., 2026c). The true-negative rate is

$$\mathrm{TNR} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}, \tag{40}$$

and prediction-balance offset is

$$\mathrm{PBO} = \left(\frac{\mathrm{TP} + \mathrm{FP}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} - \frac{1}{2} \right) \times 100. \tag{41}$$

TNR is higher-is-better. PBO is signed: positive values indicate an affirmative-answer preference, negative values indicate a rejection preference, and values closer to zero are more balanced. We therefore report PBO together with accuracy and TNR rather than optimizing its signed value in isolation. Table 6 reports accuracy, F1, TNR, and the signed PBO for every backbone and method.

C.4 AMBER

AMBER evaluates both generated descriptions and discriminative questions (Wang et al., 2023). For response n, let M_n be the recognized object mentions, G_n the annotated ground-truth objects, $H_n = M_n \setminus G_n$ the unsupported mentions, and A_n the annotated hallucinatory target objects. Define $c_n = |M_n \cap G_n|$, $a_n = |M_n \cap A_n|$, and $d_n = \max(1, |M_n|)$. The response-level generative metrics are macro-averaged over the N queries:

$$\mathrm{CHAIR}_{\mathrm{AMBER}} = \frac{1}{N} \sum_{n=1}^{N} \frac{|H_n|}{d_n}, \tag{42}$$

$$\mathrm{Cover} = \frac{1}{N} \sum_{n=1}^{N} \frac{c_n}{|G_n|}, \tag{43}$$

$$\mathrm{Hal} = \frac{1}{N} \sum_{n=1}^{N} \mathbf{1}[|H_n| > 0], \tag{44}$$

$$\mathrm{Cog} = \frac{1}{N} \sum_{n=1}^{N} \frac{a_n}{d_n}. \tag{45}$$

Table 5: Complete POPE results used to form the corpus-level means in Table 1. R, P, and A denote the random, popular, and adversarial splits. Bold and underlining mark the best and second-best result within each backbone.

	A-OKVQA						COCO					
Method	R-Acc↑	R-F1↑	P-Acc↑	P-F1↑	A-Acc↑	A-F1↑	R-Acc↑	R-F1↑	P-Acc↑	P-F1↑	A-Acc↑	A-F1↑
R1-OneVision												
Vanilla	85.77	85.25	81.87	81.85	77.33	78.47	82.87	81.19	80.73	79.31	80.77	79.34
LEAD[CVPR'26]	87.23	86.38	84.00	83.51	78.27	79.38	84.70	83.24	82.90	81.66	82.23	80.55
MCoT[CVPR'26]	86.30	85.78	82.00	82.12	77.50	78.61	83.63	82.07	82.40	81.01	82.33	81.17
RAL[CVPR'26]	84.97	84.84	81.60	81.44	77.07	78.05	81.93	81.18	80.17	79.07	80.10	78.87
CRG[ICML'26]	85.83	85.93	82.03	81.92	77.47	78.72	83.13	81.75	81.17	80.18	81.10	79.87
FLASH[ICML'26]	85.83	85.87	82.10	81.95	77.37	78.71	83.17	81.50	81.13	79.90	81.17	79.61
SPLIT (ours)	**88.10**	**87.25**	**84.83**	**84.36**	**79.37**	**79.97**	**86.43**	**84.64**	**84.33**	**82.48**	**82.97**	**81.98**
Vision-R1												
Vanilla	91.10	91.27	85.63	86.62	78.73	81.45	91.06	90.52	88.63	88.24	86.63	86.48
LEAD[CVPR'26]	91.23	91.41	85.37	86.43	78.97	81.57	91.23	90.68	88.20	87.85	86.93	86.77
MCoT[CVPR'26]	91.10	91.27	85.83	86.78	78.93	81.53	91.00	90.45	88.67	88.27	86.80	86.59
RAL[CVPR'26]	89.90	90.11	84.93	86.10	77.97	81.10	90.33	89.96	88.07	88.11	85.97	86.15
CRG[ICML'26]	91.37	91.51	85.73	86.78	79.13	81.86	91.37	90.73	88.63	88.61	86.83	86.61
FLASH[ICML'26]	91.27	91.42	85.83	86.79	79.03	81.72	91.43	90.71	88.67	88.33	86.93	86.48
SPLIT (ours)	**92.07**	**92.09**	**86.67**	**87.65**	**80.07**	**82.51**	**92.17**	**91.67**	**89.37**	**89.13**	**87.70**	**87.38**
ThinkLite-VL												
Vanilla	90.70	90.20	87.73	87.47	82.30	82.87	87.40	85.73	86.47	84.84	85.63	84.10
LEAD[CVPR'26]	91.03	90.54	88.17	87.92	81.90	82.50	87.70	86.08	86.90	85.33	85.60	84.05
MCoT[CVPR'26]	91.00	90.55	88.07	87.84	82.63	83.23	87.60	85.95	86.63	85.01	85.83	84.25
RAL[CVPR'26]	89.93	89.55	87.27	87.10	81.93	82.43	87.07	85.02	86.03	84.48	84.93	83.41
CRG[ICML'26]	91.13	90.71	88.07	87.71	82.57	82.99	87.77	86.05	86.77	85.08	85.20	83.68
FLASH[ICML'26]	91.07	90.66	87.93	87.90	82.53	83.05	87.73	86.06	86.90	85.05	85.07	83.57
SPLIT (ours)	**91.93**	**91.51**	**88.63**	**88.38**	**83.17**	**83.79**	**88.50**	**87.12**	**87.23**	**85.80**	**86.57**	**85.16**
Qwen3-VL-Thinking												
Vanilla	92.20	92.07	86.93	87.38	82.17	83.53	89.57	88.56	87.47	86.57	86.57	85.71
LEAD[CVPR'26]	92.50	92.35	87.40	87.78	81.97	83.38	89.87	88.91	87.53	86.67	87.00	86.24
MCoT[CVPR'26]	92.27	92.11	87.20	87.59	82.37	83.67	89.77	88.77	87.57	86.67	86.87	86.03
RAL[CVPR'26]	91.87	91.98	86.67	87.03	82.03	83.18	89.07	88.12	87.10	85.95	86.03	84.99
CRG[ICML'26]	92.53	92.42	87.47	87.83	82.33	83.73	89.67	88.73	87.73	86.92	86.87	86.08
FLASH[ICML'26]	92.57	92.24	87.37	87.80	82.53	83.82	89.73	88.65	87.77	86.72	86.70	85.91
SPLIT (ours)	**93.10**	**92.95**	**88.07**	**88.41**	**83.37**	**84.65**	**90.07**	**89.15**	**88.57**	**87.45**	**87.73**	**86.72**
InternVL3.5												
Vanilla	92.67	92.88	84.83	86.29	78.83	81.85	91.17	90.82	87.10	87.12	84.33	84.78
LEAD[CVPR'26]	92.77	92.96	85.10	86.49	78.13	81.27	91.17	90.76	87.33	87.32	84.90	85.20
MCoT[CVPR'26]	92.87	93.09	85.13	86.60	78.80	81.92	91.30	90.97	87.30	87.33	84.90	85.29
RAL[CVPR'26]	92.07	92.14	84.27	85.73	78.57	81.34	90.67	90.03	86.47	86.38	84.07	84.30
CRG[ICML'26]	92.97	92.86	85.13	86.47	79.13	81.93	91.43	91.12	87.43	87.38	84.70	85.12
FLASH[ICML'26]	92.97	93.15	85.10	86.52	79.10	82.05	91.37	91.11	87.47	87.49	84.57	85.03
SPLIT (ours)	**93.73**	**93.91**	**85.90**	**87.22**	**79.90**	**82.73**	**92.27**	**91.95**	**87.90**	**88.13**	**85.27**	**85.68**

Lower is better for CHAIR, Hal, and Cog; higher is better for Cover. In particular, Cog is normalized by the objects recognized in each response, not by the number of annotated target objects. We retain the released object-recognition pipeline and apply the response-level aggregation defined in the AMBER paper; an empty recognized-object parse contributes zero to CHAIR and Cog through d_n and remains visible through zero coverage.

For AMBER's discriminative partition, the official evaluator treats "no" as the positive hallucination-avoidance class. Let $z_i = 1$ when the gold answer is no and $\hat{z}_i = 1$ when the model answer is no. Precision, recall, and F1 are computed from $(z_i, \hat{z}_i)$ as usual, while accuracy is computed over all yes and no questions. Reporting both quantities reveals one-class response policies. On the unit scale, the composite score is

$$\mathrm{AMBER} = \frac{(1 - \mathrm{CHAIR}_{\mathrm{AMBER}}) + \mathrm{F1}_{\mathrm{AMBER}}}{2}. \tag{46}$$

The main table retains CH, F1, and the composite score as the most compact view of generative faithfulness, discriminative reliability, and overall performance. Table 6 provides the complete seven-metric AMBER breakdown alongside POPEv2.

Table 6: Complete POPEv2 and AMBER results. For POPEv2, PBO is signed and values closer to zero indicate a more balanced prediction rate. For AMBER, CH, Cov, Hal, and Cog are the generative diagnostics; Acc and F1 evaluate the discriminative partition; Sc is the official composite score. Bold and underlining mark the best and second-best result within each backbone.

	POPEv2				AMBER						
Method	Acc↑	PBO→ 0	F1↑	TNR↑	CH↓	Cov↑	Hal↓	Cog↓	Acc↑	F1↑	Sc↑
R1-OneVision											
Vanilla	80.6	7.8	82.00	72.8	7.2	63.9	40.2	2.3	82.4	86.4	89.6
LEAD[CVPR'26]	82.1	7.3	83.32	74.8	7.7	64.1	44.1	2.3	81.7	85.9	89.1
MCoT[CVPR'26]	81.2	7.8	82.56	73.4	6.7	64.3	34.7	2.2	82.4	86.4	89.8
RAL[CVPR'26]	79.9	8.1	81.51	72.1	6.9	63.4	38.6	2.2	82.1	85.9	89.5
CRG[ICML'26]	79.1	9.2	80.04	69.3	6.5	62.4	33.9	2.1	82.7	86.7	90.1
FLASH[ICML'26]	79.4	8.6	80.89	71.0	6.6	62.7	35.3	2.1	82.7	86.5	90.0
SPLIT (ours)	**83.2**	**6.8**	**84.27**	**76.4**	**6.1**	**64.5**	**31.7**	**1.9**	**83.6**	**87.5**	**90.7**
Vision-R1											
Vanilla	81.4	12.8	83.51	68.6	4.5	66.4	32.6	1.9	85.6	88.8	92.2
LEAD[CVPR'26]	82.4	12.2	84.31	70.2	4.2	**66.6**	30.1	1.8	85.7	88.8	92.3
MCoT[CVPR'26]	82.2	12.6	84.19	69.6	4.3	66.4	32.3	2.1	85.8	88.9	92.3
RAL[CVPR'26]	81.3	12.9	83.35	68.2	4.3	59.2	30.2	1.8	82.1	85.7	90.7
CRG[ICML'26]	79.8	15.2	82.47	64.6	4.1	62.5	29.7	1.7	85.8	88.9	92.4
FLASH[ICML'26]	77.7	17.1	80.96	60.6	4.2	64.3	31.4	1.9	85.8	89.0	92.4
SPLIT (ours)	**82.9**	**10.9**	**85.04**	**73.6**	**3.7**	**66.6**	**27.2**	**1.5**	**86.5**	**89.5**	**92.9**
ThinkLite-VL											
Vanilla	89.4	3.8	89.79	85.6	4.9	64.1	27.9	1.8	86.9	90.4	92.8
LEAD[CVPR'26]	89.8	3.8	90.17	86.0	4.9	63.8	27.9	**1.4**	87.0	90.5	92.8
MCoT[CVPR'26]	89.8	4.0	90.19	85.8	4.8	**64.4**	27.6	1.6	86.9	90.5	92.8
RAL[CVPR'26]	87.0	7.0	87.85	80.0	4.7	63.5	26.7	1.7	86.1	89.5	92.4
CRG[ICML'26]	86.2	5.8	86.96	80.4	4.5	62.6	25.2	1.6	87.3	90.8	93.2
FLASH[ICML'26]	85.4	7.6	86.43	77.8	4.6	63.2	26.1	1.7	87.2	90.7	93.1
SPLIT (ours)	**91.1**	**3.0**	**91.21**	**86.9**	**4.1**	**64.4**	**23.8**	**1.4**	**87.9**	**91.4**	**93.7**
Qwen3-VL-Thinking											
Vanilla	86.2	7.2	87.13	79.0	7.2	77.4	65.2	5.3	87.3	90.3	91.5
LEAD[CVPR'26]	86.9	8.5	87.93	78.4	6.9	**77.9**	65.0	5.3	87.4	90.4	91.8
MCoT[CVPR'26]	86.5	7.1	87.39	79.4	6.8	77.4	60.5	4.9	87.4	90.4	91.8
RAL[CVPR'26]	86.7	7.5	87.63	79.2	7.0	76.8	62.3	5.0	86.9	89.7	91.3
CRG[ICML'26]	87.0	5.8	87.71	81.2	6.6	75.2	60.8	4.9	87.7	90.5	92.0
FLASH[ICML'26]	87.5	6.1	88.22	81.4	6.8	77.1	61.6	5.0	87.6	90.4	91.8
SPLIT (ours)	**88.8**	**5.0**	**89.04**	**82.0**	**6.3**	77.7	**57.5**	**4.6**	**87.9**	**91.1**	**92.4**
InternVL3.5											
Vanilla	81.4	13.6	83.63	67.8	7.7	71.4	53.2	4.2	83.4	86.9	89.6
LEAD[CVPR'26]	78.4	15.8	81.35	62.6	7.5	71.5	51.0	3.7	83.0	86.6	89.5
MCoT[CVPR'26]	82.8	13.0	84.78	69.8	7.6	71.5	51.7	3.9	83.8	87.2	89.8
RAL[CVPR'26]	82.9	12.5	84.80	70.4	7.4	71.1	51.8	3.8	83.1	86.5	89.5
CRG[ICML'26]	81.9	13.5	84.08	68.3	7.0	69.9	50.4	3.6	83.7	87.5	90.3
FLASH[ICML'26]	81.8	13.0	84.17	68.2	7.2	70.8	50.9	3.7	83.8	87.2	90.0
SPLIT (ours)	**84.5**	**11.3**	**85.71**	**72.0**	**6.7**	**71.7**	**46.3**	**3.3**	**84.4**	**88.1**	**90.7**

C.5 MME

MME groups paired binary questions into fourteen categories (Fu et al., 2023). For category c, ordinary accuracy is

$$\mathrm{Acc}_c = \frac{1}{|\mathcal{Q}_c|} \sum_{q \in \mathcal{Q}_c} \mathbf{1}[\hat{y}_q = y_q]. \tag{47}$$

Let $\mathcal{P}_c$ be the image-level question pairs. Accuracy+ requires every question in a pair to be correct:

$$\mathrm{Acc}_c^{+} = \frac{1}{|\mathcal{P}_c|} \sum_{p \in \mathcal{P}_c} \prod_{q \in p} \mathbf{1}[\hat{y}_q = y_q]. \tag{48}$$

The category score is

$$\mathrm{MME}_c = 100(\mathrm{Acc}_c + \mathrm{Acc}_c^+). \tag{49}$$

Scores are summed over the ten perception categories and four cognition categories. The released protocol therefore has maxima of 2000 for perception, 800 for cognition, and 2800 overall.

C.6 VHTEST

VHTest defines eight modes (Huang et al., 2024b):

$$\mathcal{K} = \{\mathrm{counting}, \mathrm{shape}, \mathrm{color}, \mathrm{orientation}, \mathrm{size}, \mathrm{position}, \mathrm{OCR}, \mathrm{existence}\}. \tag{50}$$

For mode k,

$$\mathrm{Acc}_k = \frac{1}{|\mathcal{Q}_k|} \sum_{q \in \mathcal{Q}_k} \mathbf{1}[\hat{y}_q = y_q]. \tag{51}$$

The headline score is the unweighted mode average

$$\mathrm{VHTest}_{\mathrm{All}} = \frac{1}{8} \sum_{k \in \mathcal{K}} \mathrm{Acc}_k. \tag{52}$$

We preserve all eight mode scores to show whether a method helps only existence questions or transfers to attributes, spatial properties, counting, and OCR.

C.7 MIRAGE

MIRAGE isolates reasoning-induced hallucination by retaining questions whose visual content is correctly perceived (Dong et al., 2026). We report official final-answer accuracy,

$$\mathrm{Acc}_{\mathrm{MIRAGE}} = \frac{1}{N} \sum_{n=1}^{N} \mathbf{1}[\mathsf{Norm}(\hat{a}_n) = \mathsf{Norm}(a_n)]. \tag{53}$$

The released LLM-based parser extracts the terminal answer from the complete generated trace and applies task-aware normalization for multiple-choice, deterministic free-form, and approximate numerical answers; approximate questions use the tolerance attached to the corresponding benchmark item. All methods use the same released parser prompt and configuration over all 1,329 questions.

C.8 MATHVISTA$_{\mathrm{mini}}$

MathVista testmini uses type-aware exact accuracy (Lu et al., 2024):

$$\mathrm{Acc}_{\mathrm{MathVista}} = \frac{1}{N} \sum_{n=1}^{N} \mathbf{1}[\mathsf{Parse}(\hat{a}_n) = \mathsf{Parse}(a_n)]. \tag{54}$$

Multiple-choice responses are mapped to option identifiers or option strings; integer and floating-point answers are normalized numerically; short free-form strings follow the official normalizer. For a metadata subgroup g with records S_g,

$$\mathrm{Acc}(g) = \frac{1}{|S_g|} \sum_{n \in S_g} \mathbf{1}[\mathsf{Parse}(\hat{a}_n) = \mathsf{Parse}(a_n)]. \tag{55}$$

Subgroups may correspond to question type, answer type, source dataset, language skill, visual skill, or reasoning skill. The overall score remains the unweighted accuracy over testmini examples, not an average of subgroup percentages.

D EXTENDED THEORY AND PROOFS

The analysis follows the deployed controller in causal order: population estimation in post-W_O residual coordinates, streaming mode tracking, robust event detection, energy-constrained release,

and depth-aware perturbation control. It yields three complementary kinds of guarantees: high-probability estimation guarantees for the coordinate sketch, tracking guarantees under temporal drift, and exact algebraic guarantees for every realized gate.

The organization follows the controller's causal order and isolates the role of each mathematical component. The sketch results characterize when the compressed operator preserves the leading population structure; the tracking result explains how a smooth mode can coexist with mobile head carriers; scale equivariance and robust histories support a shared detector across layers and backbones; the release analysis identifies the peak-to-bulk control direction; and the final identity, energy, and perturbation results make the effect of ρ explicit. Each result therefore attaches a precise guarantee to an object used by the deployed controller.

D.1 Full Operator, Sketched Operator, and Assumptions

For a fixed layer and token, let $R \in \mathbb{R}^{H\times d_h}$ contain clean temporal innovations. The full residual-stream contribution of head h is

$$a_h = W_O^{(h)} r_h \in \mathbb{R}^{d_{\text{model}}}, \tag{56}$$

and $A \in \mathbb{R}^{H\times d_{\text{model}}}$ stacks $a_h^\top$. Define

$$Y^\star = \mathcal{N}(C_H A), \qquad G^\star = Y^\star (Y^\star)^\top. \tag{57}$$

The coordinate selector S produces $Z_h = S a_h$ and

$$\widehat{Y} = \mathcal{N}(C_H Z), \qquad \widehat{G} = \widehat{Y}\widehat{Y}^\top. \tag{58}$$

When every centered row is nonzero, both operators are positive semidefinite with unit diagonal and trace H. Their leading shares therefore lie in $[1/H, 1]$.

Degenerate rows and the active population. Let $\mathcal{I}$ be the common subset of heads whose centered full and sketched rows are nonzero. If $\mathcal{I} \neq \varnothing$, the estimator statements below apply to the principal operators indexed by $\mathcal{I}$, with $|\mathcal{I}|$ replacing H; an isolated zero row simply removes that head from the active carrier set. Numerically, $\mathcal{N}$ uses the same ϵ floor as the deployed controller, so a zero row contributes neither alignment nor a spurious direction. If the layer has no usable innovation, the explicit zero-energy bypass sets $g = \mathbf{1}$. Partial degeneracy is therefore handled by restriction and layer-wide degeneracy by exact native identity.

The estimator and tracker operate in the following sufficient regime.

1. **Embedding accuracy.** The normalized operator error $\|\widehat{G} - G^\star\|_2$ is controlled by coordinate coherence, effective rank, and sketch width.
2. **Spectral identifiability.** The full leading eigengap dominates the embedding error, making its leading direction stable.
3. **Temporal trackability.** Adjacent-token mode drift remains inside the attraction region of one warm power step.

Randomized numerical linear algebra supplies the coordinate-embedding result below, high-dimensional probability converts embedding error into operator error, and streaming PCA characterizes tracking under drift (Martinsson & Tropp, 2020; Vershynin, 2026; Bienstock et al., 2022). The resulting conditions are expressed through observable operator agreement, eigengaps, and adjacent-token angles, keeping approximation quality, identifiability, and trackability distinct.

D.2 Causal Ledger and Clean-Anchor Invariance

Let X_t denote the clean pre-W_O head output saved by a layer hook, and let $\mathcal{F}_t = \sigma(X_0, \ldots, X_t)$ be the information available after token t. The active gate at token $t+1$ is finalized before the next model call using only observations through t.

Proposition 1 (Strict causality and clean-anchor identity)**.** *For every layer, the gain g_{t+1} is $\mathcal{F}_t$-measurable. If the hook stores X_t before applying the active gain and updates equation 3 once per valid visit, then*

$$M_t = \frac{1}{n_t} \sum_{i \in \mathcal{V}_t} X_i, \tag{59}$$

where $\mathcal{V}_t$ is the set of valid clean observations through t. In particular, the reference process is not a running mean of already edited tensors.

Proof. At initialization, $M_0 = X_0$ and equation 59 holds. Assume it holds after n_{t-1} valid observations. The next update is

$$M_t = M_{t-1} + \frac{X_t - M_{t-1}}{n_{t-1}+1} = \frac{n_{t-1}M_{t-1} + X_t}{n_{t-1}+1}, \tag{60}$$

which is the mean of the augmented clean set. Because X_t is saved before the interpolation in equation 16, the induction never substitutes X'_t for X_t. All statistics used to construct g_{t+1} are deterministic functions of the clean observations, cached selectors, and controller state through t, so g_{t+1} is $\mathcal{F}_t$-measurable. □

The proposition establishes the controller property used throughout the analysis: conditional on the realized autoregressive trajectory, each layer-local reference is formed from clean hook observations rather than recursively filtered states. Earlier emitted tokens remain part of that realized trajectory, preserving the causal effect of decoding decisions while isolating the controller's internal reference update.

D.3 Sketch-to-Operator Stability

Let $u^\star$ be a unit leading eigenvector of $G^\star$, $\widehat{u}$ a unit leading eigenvector of $\widehat{G}$, $c^\star = \lambda_1(G^\star)/H$, and $\widehat{c} = \lambda_1(\widehat{G})/H$.

Proposition 2 (Leading share and participation stability)**.** *Assume every centered full and sketched row is nonzero, $\|\widehat{G} - G^\star\|_2 \le \eta$, and the leading eigengap $\gamma = \lambda_1(G^\star) - \lambda_2(G^\star)$ satisfies $\gamma > 2\eta$. After aligning eigenvector signs,*

$$|\widehat{c} - c^\star| \le \eta/H, \tag{61}$$

$$\sin\angle(\widehat{u}, u^\star) \le \frac{\eta}{\gamma - \eta} \le \frac{2\eta}{\gamma}, \tag{62}$$

$$\|\widehat{u} \odot \widehat{u} - u^\star \odot u^\star\|_1 \le 2\,\sin\angle(\widehat{u}, u^\star). \tag{63}$$

The same statements hold on a common nonzero active set, with its cardinality replacing H.

Proof. Weyl's inequality gives $|\lambda_1(\widehat{G}) - \lambda_1(G^\star)| \le \eta$; division by the common trace H proves equation 61. The Davis–Kahan sin-Θ bound for a simple leading eigenvalue yields $\sin\angle(\widehat{u}, u^\star) \le \eta/(\gamma - \eta)$; the gap assumption gives the second inequality. Finally,

$$\widehat{u} \odot \widehat{u} - u^\star \odot u^\star = (\widehat{u} - u^\star) \odot (\widehat{u} + u^\star). \tag{64}$$

Cauchy–Schwarz bounds its ℓ_1 norm by $\|\widehat{u} - u^\star\|_2 \|\widehat{u} + u^\star\|_2$. Under sign alignment, the chord–angle identities give $\|\widehat{u} - u^\star\|_2 = 2\sin(\theta/2)$ and $\|\widehat{u} + u^\star\|_2 = 2\cos(\theta/2)$, whose product is exactly $2\sin\theta$. □

The next result makes the coordinate-sketch condition concrete. Let $B = C_H A \in \mathbb{R}^{H \times d}$, let $r = \operatorname{rank}(B)$, and let $V \in \mathbb{R}^{d \times r}$ be an orthonormal basis for the row space of B. Define its coordinate coherence by

$$\mu = \frac{d}{r} \max_{j \in [d]} \|e_j^\top V\|_2^2 \in [1, d/r]. \tag{65}$$

Proposition 3 (Finite-sample coordinate-sketch guarantee)**.** *Sample s output coordinates uniformly without replacement and let B_Ω denote the selected columns. For $0 < \varepsilon < 1$ and $0 < \delta < 1$, there is a universal constant C such that*

$$s \ge C\,\mu r\,\varepsilon^{-2}\log(2r/\delta) \tag{66}$$

implies, with probability at least $1 - \delta$,

$$(1 - \varepsilon)BB^\top \preceq \frac{d}{s} B_\Omega B_\Omega^\top \preceq (1 + \varepsilon)BB^\top. \tag{67}$$

If all centered head rows are nonzero, the full and row-normalized sketched Gram operators satisfy

$$\max_{h,k} |\widehat{G}_{hk} - G^\star_{hk}| \le \frac{2\varepsilon}{1-\varepsilon}, \tag{68}$$

$$\|\widehat{G} - G^\star\|_2 \le \frac{2H\varepsilon}{1-\varepsilon}, \tag{69}$$

$$|\widehat{c} - c^\star| \le \frac{2\varepsilon}{1-\varepsilon}. \tag{70}$$

Whenever this operator error is below half the full eigengap, Proposition 2 also controls the participation vector.

Proof. Uniform coordinate sampling of a rank-r subspace with coherence μ gives the subspace-embedding inequality equation 67 at the sample complexity in equation 66 (Martinsson & Tropp, 2020). Write $Q = BB^\top$ and $\widetilde{Q} = (d/s)B_\Omega B_\Omega^\top$. The PSD sandwich implies

$$|\widetilde{Q}_{hk} - Q_{hk}| \le \varepsilon\sqrt{Q_{hh}Q_{kk}}, \qquad (1-\varepsilon)Q_{hh} \le \widetilde{Q}_{hh} \le (1+\varepsilon)Q_{hh}. \tag{71}$$

The global factor d/s cancels under row normalization. Adding and subtracting $Q_{hk}/\sqrt{\widetilde{Q}_{hh}\widetilde{Q}_{kk}}$, using $|Q_{hk}| \le \sqrt{Q_{hh}Q_{kk}}$, and applying the diagonal bounds gives

$$\left| \frac{\widetilde{Q}_{hk}}{\sqrt{\widetilde{Q}_{hh}\widetilde{Q}_{kk}}} - \frac{Q_{hk}}{\sqrt{Q_{hh}Q_{kk}}} \right| \le \frac{2\varepsilon}{1-\varepsilon}. \tag{72}$$

The spectral norm of an $H \times H$ matrix is at most H times its largest absolute entry, which proves equation 69. Weyl's inequality and the common trace H then give equation 70. □

Because $r \le H$, the result explains why sketch width should scale with the head population rather than the much larger ambient width d: the relevant dimension is the effective head-population rank. The deployed rule $s = \min\{d, \max(8, 4H)\}$ instantiates this linear scaling uniformly across model geometries; the coefficient four is a fixed architecture rule rather than a claim that one constant satisfies the worst-case bound for arbitrary coherence. Equation equation 66 states the corresponding fidelity regime through μ, ε, and δ, and indicates when a wider sketch is needed. Its probability is over the selector sampled once at initialization. Conditional on that cached selector, decoding is deterministic, while the exact identity, energy, and budget guarantees below hold for every selector independently of the embedding event. Proposition 2 then translates any realized operator fidelity into leading-share and participation fidelity. First-order eigenvector perturbation expansions and infinity-norm Davis–Kahan bounds can further sharpen individual squared-loading control under structured low-rank perturbations (Greenbaum et al., 2020; Bhardwaj & Vu, 2024).

D.4 Warm Tracking of a Drifting Mode

Let $u_t^\star$ be the unit leading eigenvector of the current sketched G_t and define $\kappa_t = \lambda_2(G_t)/\lambda_1(G_t) < 1$.

Proposition 4 (One-step spectral tracking)**.** *Let $\widehat{u}_{t-1}$ be the previous unit estimate, $\theta_{t-1} = \angle(\widehat{u}_{t-1}, u_{t-1}^\star)$, and $\delta_t = \angle(u_{t-1}^\star, u_t^\star)$. If $\theta_{t-1} + \delta_t < \pi/2$, the normalized one-step update in equation 8 obeys*

$$\tan\angle(\widehat{u}_t, u_t^\star) \le \kappa_t \tan(\theta_{t-1} + \delta_t). \tag{73}$$

If the applied operator has an additional perturbation of norm η_t and eigengap γ_t, the recursion gains an additive angular term of order η_t/γ_t.

Proof. The triangle inequality for principal angles gives $\angle(\widehat{u}_{t-1}, u_t^\star) \le \theta_{t-1} + \delta_t$. Expand $\widehat{u}_{t-1} = au_t^\star + bv$ for a unit $v \perp u_t^\star$. Multiplication by G_t scales the leading component by λ_1 and the orthogonal component by at most λ_2. Normalization does not change their ratio, hence

$$\tan\angle(\widehat{u}_t, u_t^\star) \le \frac{\lambda_2}{\lambda_1}\left|\frac{b}{a}\right| \le \kappa_t \tan(\theta_{t-1} + \delta_t). \tag{74}$$

A Davis–Kahan perturbation step between the intended and applied operator adds an angle proportional to η_t/γ_t when the gap dominates the perturbation. □

The bound explains the empirical conjunction in Figure 2a: a population direction may be locally trackable even while its largest coordinates turn over. Repeated application converges to a contraction-plus-drift floor rather than a single static vector, matching the qualitative regime of streaming PCA with a time-varying covariance (Bienstock et al., 2022). The same proof gives the three-step cold-start guarantee $\tan\theta^{(3)} \le \kappa_t^3 \tan\theta^{(0)}$ whenever the deterministic initializer has nonzero leading-mode alignment.

Why three cold steps suffice. Three cold steps provide a coarse initializer, after which the warm recursion contracts toward its temporal perturbation floor. Let $e_t = \tan\angle(\widehat{u}_t, u_t^\star)$, $\tau_{t+1} = \tan\delta_{t+1}$, and let ξ_{t+1} denote the additive tangent error from operator perturbation. The tangent-addition identity makes Proposition 4 explicit:

$$e_{t+1} \le \kappa_{t+1}\frac{e_t + \tau_{t+1}}{1 - e_t\tau_{t+1}} + \xi_{t+1}. \tag{75}$$

In a local region $e_t\tau_{t+1} \le \chi < 1$, suppose $\kappa_{t+1}/(1-\chi) \le \bar\kappa < 1$ and $\bar\kappa\tau_{t+1} + \xi_{t+1} \le \nu$. Then $e_{t+1} \le \bar\kappa e_t + \nu$, and unrolling gives

$$e_{t+m} \le \bar\kappa^m e_t + \frac{1-\bar\kappa^m}{1-\bar\kappa}\nu. \tag{76}$$

For a fixed cold-start operator, $\tau = \xi = 0$ and three steps give $e^{(3)} \le \kappa^3 e^{(0)}$; even $\kappa = 0.8$ halves the initial tangent error. Subsequent tokens then supply inexpensive warm contractions toward the local perturbation floor $\nu/(1-\bar\kappa)$. Exact eigensolution is also unnecessary for the leading-share statistic: with eigenvalues ordered $\lambda_1 \ge \cdots \ge \lambda_H$, a unit estimate at angle θ from the leading eigenvector satisfies

$$0 \le \lambda_1 - \widehat{u}^\top G\widehat{u} \le (\lambda_1 - \lambda_H)\sin^2\theta. \tag{77}$$

The three-step start therefore provides a usable mode and Rayleigh share, while temporal regularity performs the remaining refinement online instead of front-loading repeated matrix multiplications at the first token.

D.5 Robust Self-Calibration Across Layers and Models

The detector combines bounded medians and MADs with the architecture-specific isotropic share $q_0 = 1/H$. Its dimensionless factors have two useful invariances.

Proposition 5 (Scale equivariance and majority robustness)**.** *Consider the detector in equations 10–12 with machine floors inactive.*

1. *For any $a > 0$, transform every share in the current value and history by $c' = q_0 + a(c - q_0)$. Then r_t and q_t are unchanged. Transform every innovation energy by $v' = av$; then e_t is unchanged. Consequently, the realized detector product $r_t q_t e_t$ is invariant to these positive rescalings.*

2. *Let a history contain m values, more than half of which lie in an interval $[A, B]$. Its median lies in $[A, B]$ and its raw MAD is at most $B - A$, irrespective of the magnitudes of the remaining values. Thus a minority of arbitrarily extreme history values cannot drive either robust reference without bound.*

Proof. Positive affine maps commute with the median, and the MAD scales by a. For the share history,

$$b_c' = q_0 + a(b_c - q_0), \qquad \mathrm{MAD}(\mathcal{H}_c') = a\,\mathrm{MAD}(\mathcal{H}_c), \tag{78}$$

while $(b_c' - q_0)_+/H = a(b_c - q_0)_+/H$. Hence $s_c' = as_c$ and $q_c' = q_0 + a(q_c - q_0)$. Substitution cancels a in both r_t and q_t. For energy, $b_v' = ab_v$, $\mathrm{MAD}(\mathcal{H}_v') = a\,\mathrm{MAD}(\mathcal{H}_v)$, and $|b_v'|/H = a|b_v|/H$, so $s_v' = as_v$ and e_t is unchanged.

For the second statement, every median must lie between two central order statistics. If more than half the observations belong to $[A, B]$, those central statistics lie in $[A, B]$. Relative to such a median, more than half the absolute deviations are at most $B - A$, so their median is also at most $B - A$. □

The proposition gives a formal reason that one global ρ can be shared across layer scales and model widths: model-specific magnitudes are absorbed by robust local references, while the isotropic point $1/H$ adjusts automatically to head count. The machine floors preserve continuity in quiet histories and make the implemented detector well-defined in every degenerate case.

D.6 Symmetries and Head Reindexing

An attention-head index is a parameterization coordinate, not an intrinsic semantic name. Let P be an $H \times H$ permutation matrix and consistently reindex head outputs, blocks of W_O, energies, histories, and the warm state.

Proposition 6 (Sign invariance and permutation equivariance)**.** *Under consistent reindexing,*

$$Z \mapsto PZ, \quad G \mapsto PGP^\top, \quad u \mapsto Pu, \quad p \mapsto Pp, \quad g \mapsto Pg. \tag{79}$$

Moreover, $u \mapsto -u$ leaves c, p, α, and g unchanged. Thus the converged rule and every consistently reindexed warm update privilege no fixed semantic head identity.

Proof. C_H commutes with every permutation and row normalization commutes with row reordering, hence G transforms by conjugation. Matrix-vector multiplication then gives $Gu \mapsto PGu$, coordinate squaring gives $p \mapsto Pp$, and energy-weighted averaging and radial retraction commute with the same permutation. Sign invariance follows because $u^\top Gu$ and $u \odot u$ are unchanged by $u \mapsto -u$. The deterministic cold initializer must itself be reindexed with controller state for an exact finite-iteration equivariance statement; after initialization, the warm recursion preserves equivariance exactly. □

This token-local view agrees with mechanistic work that assigns context-dependent functions to heads across layers, models, prompts, and decompositions (Elhage et al., 2021; Rai et al., 2024). SPLIT may therefore assign different gains to the same head index as its loading changes over time.

D.7 Release as Constrained Steepest Descent

Normalize the lagged energies so that $\sum_h w_h = 1$ and define the weighted energy sphere

$$\mathcal{M} = \left\{ g \in \mathbb{R}^H : \sum_h w_h g_h^2 = 1 \right\} \tag{80}$$

with metric $\langle a, b \rangle_w = \sum_h w_h a_h b_h$. Let the local dominant-mode exposure surrogate be $J(g) = \sum_h w_h p_h g_h$.

Proposition 7 (Constrained steepest release)**.** *At the identity $\mathbf{1} \in \mathcal{M}$, the vector $\bar{p}_w \mathbf{1} - p$ used in equation 14 is the negative Riemannian-gradient direction of J, where $\bar{p}_w = \sum_h w_h p_h$. If $D = \|\bar{p}_w \mathbf{1} - p\|_\infty > 0$, then*

$$\frac{\mathrm{d}}{\mathrm{d}\tau} J\left(\mathbf{1} + \tau \frac{\bar{p}_w \mathbf{1} - p}{D} \right)\Big|_{\tau=0} = -\frac{\mathrm{Var}_w(p)}{D} \leq 0. \tag{81}$$

The normalization in equation 15 is a radial retraction to $\mathcal{M}$ and exactly conserves the reference-token innovation energy used to construct the lagged gate:

$$\sum_h E_h g_h^2 = \sum_h E_h. \tag{82}$$

Proof. The tangent space at identity is $T_{\mathbf{1}}\mathcal{M} = \{d : \sum_h w_h d_h = 0\}$. Under the weighted metric, the ambient gradient of J is p. Its tangent projection is $p - \bar{p}_w \mathbf{1}$, so the negative Riemannian gradient is $\bar{p}_w \mathbf{1} - p$. Substitution into the directional derivative gives

$$\frac{\bar{p}_w^2 - \sum_h w_h p_h^2}{D} = -\frac{\mathrm{Var}_w(p)}{D}. \tag{83}$$

Finally, Equation 15 divides the provisional vector by its weighted norm:

$$\sum_h w_h g_h^2 = \frac{\sum_h w_h \widetilde{g}_h^2}{\sum_h w_h \widetilde{g}_h^2} = 1. \tag{84}$$

Multiplying by $\sum_h E_h$ proves equation 82. This identity concerns the token-t energies that define g_{t+1}; the next-token innovation is unobserved when the gate is finalized and is governed instead by the drift and perturbation results below. □

The tangent projection and radial retraction are standard operations for optimization on an embedded sphere, while information geometry and natural-gradient analyses supply the broader metric-aware steepest-descent interpretation (Amari, 2016; Martens, 2020; Boumal, 2023). When p is uniform, $\mathrm{Var}_w(p) = 0$ and the population offers no preferred release direction.

D.8 Retraction Is a Second-Order Correction

The radial normalization enforces exact energy conservation while preserving the first-order release direction, as quantified next.

Proposition 8 (Local retraction expansion). *Let $d = (\bar{p}_w \mathbf{1} - p)/D$, so $\sum_h w_h d_h = 0$, and let*

$$g(\alpha) = \frac{\mathbf{1} + \alpha d}{\sqrt{\sum_h w_h (1 + \alpha d_h)^2}}. \tag{85}$$

Then $g(0) = \mathbf{1}$, $g'(0) = d$, and

$$\|g(\alpha) - (\mathbf{1} + \alpha d)\|_w \le \frac{\alpha^2}{2} \|d\|_w^2. \tag{86}$$

Thus the exact retraction changes the tangent release only at second order.

Proof. Because $\sum_h w_h d_h = 0$ and $\sum_h w_h = 1$,

$$\sum_h w_h (1 + \alpha d_h)^2 = 1 + \alpha^2 \|d\|_w^2. \tag{87}$$

Differentiation at zero gives $g'(0) = d$. Let $q = \alpha^2 \|d\|_w^2$. Since $\|\mathbf{1} + \alpha d\|_w = \sqrt{1+q}$,

$$\|g(\alpha) - (\mathbf{1} + \alpha d)\|_w = \sqrt{1+q} - 1 \le q/2, \tag{88}$$

where the final inequality follows from concavity of the square root. □

For $\alpha \in [0, 1]$ and $d_h \in [-1, 1]$, provisional gains are nonnegative. The retraction multiplier is also nonnegative, so final gains remain nonnegative. Exact conservation controls their energy-weighted RMS, while Proposition 10 supplies a complementary coordinatewise and downstream perturbation envelope that scales directly with α.

D.9 Lagged Descent Under Adjacent-Token Drift

SPLIT applies the direction estimated at token t to token $t + 1$. The following result quantifies when that causal direction remains a descent direction for the next population.

Proposition 9 (One-token-lag descent margin). *Let $a_t = w_t \odot p_t$, $D_t = \|\bar{p}_{w,t} \mathbf{1} - p_t\|_\infty > 0$, and $d_t = (\bar{p}_{w,t} \mathbf{1} - p_t)/D_t$. For the next-token linear exposure*

$$J_{t+1}(g) = \langle a_{t+1}, g \rangle, \qquad \zeta_t = \|a_{t+1} - a_t\|_1, \tag{89}$$

the radial-retraction path generated from the lagged direction satisfies

$$\frac{\mathrm{d}}{\mathrm{d}\alpha} J_{t+1}(g_t(\alpha)) \bigg|_{\alpha=0} \le -\frac{\mathrm{Var}_{w_t}(p_t)}{D_t} + \zeta_t. \tag{90}$$

Hence the lagged gate remains a strict current-token descent direction whenever $\zeta_t < \mathrm{Var}_{w_t}(p_t)/D_t$.

Proof. Proposition 8 gives $g_t'(0) = d_t$. Decompose the derivative as

$$\langle a_{t+1}, d_t \rangle = \langle a_t, d_t \rangle + \langle a_{t+1} - a_t, d_t \rangle \tag{91}$$

$$= -\frac{\mathrm{Var}_{w_t}(p_t)}{D_t} + \langle a_{t+1} - a_t, d_t \rangle. \tag{92}$$

Because $\|d_t\|_\infty \le 1$, Hölder's inequality bounds the final term by ζ_t, proving equation 90. □

This margin connects the control lag to the same temporal regularity measured in Figure 2: stronger instantaneous participation contrast enlarges the descent margin, while slower adjacent-token drift reduces ζ_t. The controller therefore benefits precisely from the observed regime of smooth population modes with mobile individual carriers.

D.10 Strength-Controlled Local and Network Perturbations

Energy preservation controls the detector metric; the next result additionally converts the user-facing strength into a norm bound on the tensor actually injected into the next forward pass.

Proposition 10 (Local edit and downstream envelope)**.** *Let $q = \alpha^2\|d\|_w^2 \le \alpha^2$ and $g = (\mathbf{1} + \alpha d)/\sqrt{1+q}$, with $\langle \mathbf{1}, d\rangle_w = 0$, $\|d\|_\infty \le 1$, and $0 \le \alpha \le 1$. Then*

$$\|g - \mathbf{1}\|_w^2 = 2\left(1 - \frac{1}{\sqrt{1+q}}\right) \le q \le \alpha^2, \tag{93}$$

$$\|g - \mathbf{1}\|_\infty \le \alpha + \frac{\alpha^2}{2} \le \frac{3}{2}\alpha. \tag{94}$$

For the clean next-token innovation R_{t+1} and its post-projection edit ΔY_{t+1},

$$\|X'_{t+1} - X_{t+1}\|_F \le \frac{3}{2}\alpha_t \|R_{t+1}\|_F, \tag{95}$$

$$\|\Delta Y_{t+1}\|_2 \le \frac{3}{2}\alpha_t \|W_O\|_2 \|R_{t+1}\|_F. \tag{96}$$

If the downstream map from layer ℓ's residual injection to final logits is locally K_ℓ-Lipschitz along the intervened path, then the multi-layer logit perturbation obeys

$$\|\Delta z_{t+1}\|_2 \le \frac{3}{2}\rho \left(\sum_{\ell=1}^{L} K_\ell^2 \|W_{O,\ell}\|_2^2 \|R_{\ell,t+1}\|_F^2\right)^{1/2}. \tag{97}$$

Proof. Both g and $\mathbf{1}$ have unit weighted norm, and $\langle g, \mathbf{1}\rangle_w = 1/\sqrt{1+q}$. Expanding their squared distance proves the equality in equation 93; the inequality follows from $1 - (1+q)^{-1/2} \le q/2$. For each coordinate,

$$|g_h - 1| \le \alpha|d_h| + \left(\sqrt{1+q} - 1\right) \le \alpha + q/2 \le \alpha + \alpha^2/2, \tag{98}$$

where the denominator $\sqrt{1+q} \ge 1$ only tightens the bound. Applying this coordinate bound to $X'_{t+1} - X_{t+1} = (g_t - \mathbf{1}) \odot R_{t+1}$ gives equation 95; submultiplicativity gives equation 96. Finally, telescope the layerwise downstream changes, apply the local Lipschitz constants and Cauchy–Schwarz, and use Proposition 11, $\sum_\ell \alpha_{\ell,t}^2 \le \rho^2$, to obtain equation 97. □

Equation 97 complements exact energy preservation: ρ controls not only the detector amplitude but also an explicit upper envelope on the actual residual-stream and downstream edits. At $\rho = 0$ the envelope collapses to exact native identity.

D.11 Identity Cases and Network-Level Budget

Proposition 11 (Exact safeguards and depth-independent detector budget)**.** *If $\rho = 0$, either robust history is unready, any multiplicative detector factor is zero, total innovation energy is zero, or p is uniform, SPLIT returns the native model tensor for that layer. Otherwise, equation 82 holds for the reference-token innovation used to construct the lagged gate and*

$$0 \le \alpha_{\ell,t} \le \min\{1, \rho/\sqrt{L}\}, \qquad \sum_{\ell=1}^{L} \alpha_{\ell,t}^2 \le \rho^2. \tag{99}$$

Proof. The first four identity cases set $\alpha = 0$ or invoke the explicit zero-energy bypass, so $\widetilde{g} = g = \mathbf{1}$. If p is uniform, then $\bar{p}_w \mathbf{1} - p = 0$ and the release direction is defined to return identity. For a valid event, $r, q, e \in [0,1]$ and the outer clipping in equation 13 maps to $[0,1]$, yielding $\alpha_{\ell,t} \le \min\{1, \rho/\sqrt{L}\}$. Squaring and summing over L layers gives the second inequality. Energy conservation is Proposition 7. □

The $1/\sqrt{L}$ factor reflects the signal-propagation principle that layerwise perturbations should be budgeted at network scale rather than repeated at full strength in every block (Noci et al., 2022; Giorlandino & Goldt, 2026). Combined with Proposition 10, it turns the single user-facing ρ into a depth-independent envelope on detector amplitudes and on the resulting local and downstream edits.

D.12 Guarantee Hierarchy and Observable Conditions

The results form a nested guarantee hierarchy, from estimator accuracy to exact controller algebra:

- Proposition 3 gives a high-probability coordinate-sampling regime; Proposition 2 converts its operator error into leading-share and participation error.
- Proposition 4 provides a streaming contraction, and Proposition 9 converts adjacent-token regularity into a causal descent margin.
- Proposition 5 proves scale transfer and resistance to a minority of extreme history values, explaining why a single detector form can be shared across layers and models.
- Propositions 7 and 8 identify the release as constrained steepest descent followed by a second-order retraction.
- Propositions 10 and 11 give exact identity, energy, depth-budget, and strength-dependent perturbation guarantees for the deployed gate.

The hierarchy also exposes its observable conditions: operator agreement and eigengap govern sketch fidelity, warm-step angle and adjacent-token drift govern tracking, and detector activation, energy residual, and realized edit norm describe the executed intervention. This decomposition maps each sufficient condition to a concrete controller quantity without mixing probabilistic estimator assumptions with exact gate identities. Figure 4 provides the event-conditioned comparison between factual and hallucinated traces, while the five-backbone evaluation establishes the intervention's outcome-level effect. Time-uniform confidence sequences offer an extension under an explicit stochastic process model (Howard et al., 2021); the deployed controller instead calibrates its local histories online through medians and MADs.

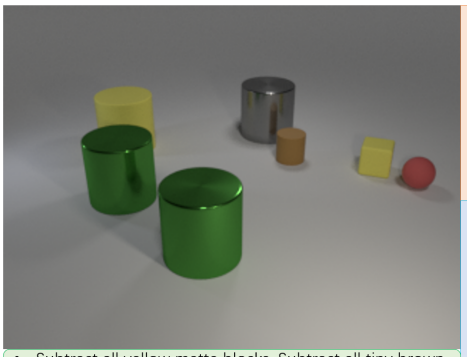
The theory characterizes the full estimator–tracker–controller chain: the population statistic is recoverable under a coordinate-embedding regime, the mode is trackable under temporal regularity, the release follows the steepest feasible peak-to-bulk direction, and the realized gate obeys exact conservation and strength-controlled perturbation bounds.

E Additional Qualitative Examples

This appendix presents 24 paired generations spanning all five evaluated backbones and three complementary settings: visual mathematical reasoning on MathVista$_{\mathrm{mini}}$ (Lu et al., 2024), fine-grained hallucination decisions on VHTest (Huang et al., 2024b), and open-ended descriptions on AMBER (Wang et al., 2023). Each panel holds the image, prompt, backbone, and decoding protocol fixed. The peach response uses vanilla decoding and the blue response uses the same backbone with SPLIT. Red highlights unsupported or incorrect content; green highlights visually grounded evidence recovered or preserved by SPLIT.

E.1 Visual Mathematical Reasoning

Figures 8–10 cover object arithmetic, charts, a stem-and-leaf plot, and a financial table. In each vanilla trace, an early, semantically plausible reading persists despite the fine-grained visual evidence needed to revise it. SPLIT couples retained concentration with the arriving innovation and reallocates the mode's current carriers, allowing local object, chart, and document evidence to redirect the trace before arithmetic hardens the premise into an answer.



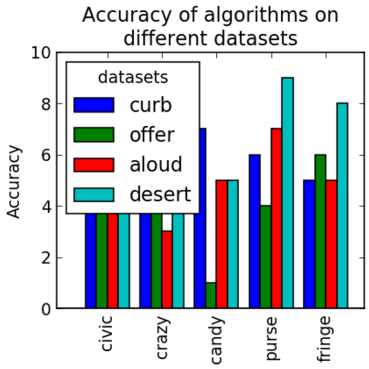
- Subtract all yellow matte blocks. Subtract all tiny brown cylinders. How many objects are left?

R1-Onevision

- <think>\nThe image shows eight objects: two green cylinders, one yellow cylinder, one gray cylinder, one small brown cylinder, one yellow cube, one red sphere, and one small yellow cube. The question asks to subtract all yellow matte blocks and all tiny brown cylinders.\n\nFirst, let's identify the objects to be subtracted:\n- Yellow matte blocks: The image shows one small yellow cube. Since it is a cube, it is a block, and it is yellow. Therefore, we subtract one object.\n- Tiny brown cylinders: The image shows one small brown cylinder. This is a cylinder, and it is brown. Therefore, we subtract one object.\n\nAfter subtracting these two objects, we are left with the following objects:\n- Two green cylinders\n- One yellow cylinder\n- One gray cylinder\n- One small brown cylinder\n- One red sphere\n- One small yellow cube\n\nCounting these remaining objects, we have a total of six objects.\n\nTherefore, the number of objects left is 6.\n</think>\n\nThe image shows eight objects: two green cylinders, one yellow cylinder, one gray cylinder, one small brown cylinder, one yellow cube, one red sphere, and one small yellow cube. Subtracting the one yellow cube and one small brown cylinder leaves six objects.\n\nAnswer: 6.

R1-Onevision+SPLIT

- <think>\nThe image shows seven objects: two green cylinders, one yellow cylinder, one gray cylinder, one small brown cylinder, one yellow cube, and one red sphere. The question asks to subtract all yellow matte blocks and all tiny brown cylinders.\n\nFirst, let's identify the objects to be subtracted:\n- Yellow matte blocks: The image shows one yellow cylinder and one small yellow cube. One is yellow and matte, so it should be subtracted.\n- Tiny brown cylinders: The image shows one small brown cylinder, which is tiny and brown, so it should also be subtracted.\n\nNow, let's subtract these objects from the total count:\n- Total objects initially: 7\n- Subtract yellow matte block: 7 - 1 = 6\n- Subtract tiny brown cylinders: 6 - 1 = 5\n\nTherefore, the number of objects left is 5.\n</think>\n\nThe image shows seven objects: two green cylinders, one yellow cylinder, one gray cylinder, one small brown cylinder, one yellow cube, and one red sphere. Subtracting the one yellow cube and the one small brown cylinder leaves five objects.\n\nAnswer: 5



- What is the sum of accuracies of the algorithm candy for all the datasets?

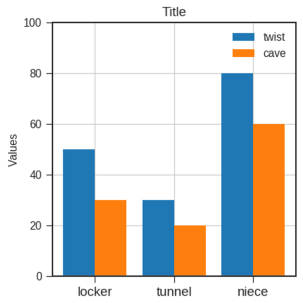
R1-Onevision

- <think>\nOkay, so I need to figure out the sum of accuracies for the algorithm named \"candy\" across all the datasets. Let me start by understanding the problem.\n\nFirst, I see that the image is a bar chart with four datasets: curb, offer, aloud, and desert. Each dataset has five bars corresponding to the categories 'civic', 'crazy', 'candy', 'purse', and 'fringe'. The y-axis represents accuracy, which I assume is a score out of 10 since the values go up to 9.\n\nLooking at the data provided in the image, each dataset has its own set of accuracy values for each category. For example, the 'curb' dataset has accuracies [5, 6, 7, 8, 9] for the categories in order. So, the first value is for 'civic', the second for 'crazy', and so on.\n\nSince the question is about the algorithm named \"candy\", I need to find the accuracy value for 'candy' across all datasets. That means I should look at the third element in each dataset's accuracy list because the categories are in the order 'civic', 'crazy', 'candy', 'purse', 'fringe'.\n\nLet me list out the accuracies for 'candy' from each dataset as seen in the image:\n\n- curb: 7\n- offer: 4\n- aloud: 5\n- desert: 6\n\nNow, I need to sum these values. Adding them together: 7 + 4 + 5 + 6. Let me do the math step by step.\n\n7 plus 4 is 11. Then, 11 plus 5 is 16. Finally, 16 plus 6 equals 22.\n\nSo, the sum of accuracies for the algorithm \"candy\" across all datasets is 22.\n</think>\n\nThe sum of accuracies for the algorithm \"candy\" across all datasets is calculated by adding the values for 'candy' in each dataset. From the image, these values are 7 (curb), 4 (offer), 5 (aloud), and 6 (desert). Adding these together: 7 + 4 + 5 + 6 = 22.\n\n22\n\nAnswer: 22

R1-Onevision+SPLIT

- <think>\nOkay, so I need to figure out the sum of accuracies for the algorithm named \"candy\" across all the datasets. Let me start by understanding the problem.\n\nFirst, I see that the image is a bar chart with four datasets: curb, offer, aloud, and desert. Each dataset has five categories, which are the algorithms: civic, crazy, candy, purse, and fringe. So, each category has four bars corresponding to each dataset.\n\nThe question is specifically about the algorithm \"candy.\" I need to find its accuracy across all four datasets. That means I should look at the values for \"candy\" in each dataset.\n\nAs seen in the image, the accuracies for \"candy\" are as follows:\n- Curb: 7\n- Offer: 1\n- Aloud: 5\n- Desert: 5\n\nNow, I'll add these values together. Let's do the math step by step:\n\n7 (curb) + 1 (offer) = 8\n8 + 5 (aloud) = 13\n13 + 5 (desert) = 18\n\nSo, the sum of accuracies for the algorithm \"candy\" across all datasets is 18.\n</think>\n\nThe sum of accuracies for the algorithm \"candy\" across all datasets is calculated by adding the values for each dataset: 7 (curb) + 1 (offer) + 5 (aloud) + 5 (desert) = 18.\n\n**Answer:** 18

Figure 8: **R1-OneVision on MathVista.** Top: SPLIT counts seven objects without duplicating the yellow cube and returns 5 after the two requested removals. Bottom: it reads the four "candy" bars as $7, 1, 5, 5$ and obtains the correct sum of 18.



- What is the value of the smallest individual bar in the whole chart?

R1-Onevision

- <think>\nOkay, so I need to figure out the value of the smallest individual bar in the whole chart. Let me start by understanding the problem. The image shows a bar chart with two sets of bars for each category: 'twist' and 'cave'. There are three categories: 'locker', 'tunnel', and 'niece'.\n\nFirst, I'll look at the 'twist' bars. The image shows the values for 'twist' as 50, 30, and 90. So, the smallest value here is 30, which corresponds to the 'tunnel' category.\n\nNext, I'll examine the 'cave' bars. The image shows the values for 'cave' as 20, 10, and 60. The smallest value here is 10, which is in the 'tunnel' category as well.\n\nComparing the two smallest values, 30 and 10, the smallest is clearly 10. Therefore, the smallest individual bar in the whole chart is 10.\n</think>\n\n\nThe smallest individual bar in the chart is 10.\n\n\nAnswer: 10

R1-Onevision+SPLIT

- <think>\nOkay, so I need to figure out the value of the smallest individual bar in the whole chart. Let me start by understanding the problem. The image is a bar chart with two sets of bars for each category: 'twist' and 'cave'. There are three categories: locker, tunnel, and niece.\n\nAs seen in the image, the 'twist' bars have values of 50, 30, and 80 for locker, tunnel, and niece respectively. The 'cave' bars have values of 30, 20, and 60 for the same categories.\n\nI need to find the smallest individual bar. That means I should look at each bar's value and find the minimum. Let's list them out:\n\n- Locker: twist=50, cave=30\n- Tunnel: twist=30, cave=20\n- Niece: twist=80, cave=60\n\nNow, comparing all these values: 50, 30, 30, 20, 80, 60. The smallest number here is 20. So, the 'cave' bars for tunnel has a value of 20, which is the smallest in their respective categories.\n\nTherefore, the smallest individual bar in the whole chart is 20.\n</think>\n\n\nThe smallest individual bar in the chart is 20.\n\n\nAnswer: 20

Sit-up competition (number of sit-ups)

Stem	Leaf
0	0 0 2
1	4 5
2	1 2
3	
4	2 5 6
5	5 5
6	5 5 8
7	3
8	2 3 6

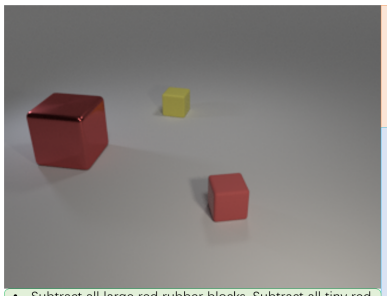
- Miss Foley ran a sit-up competition among her P.E. students and monitored how many sit-ups each students could do. What is the largest number of sit-ups done? (Unit: sit-ups)

R1-Onevision

- <think>\nOkay, so I need to figure out the largest number of sit-ups done by the students based on the provided stem-and-leaf plot. Let me start by understanding what a stem-and-leaf plot is. It's a way to display quantitative data where each data point is split into a \"stem\" (the leading digit or digits) and a \"leaf\" (the trailing digit).\n\nLooking at the image, the stem represents the tens place, and the leaf represents the ones place. So, for example, in the row where the stem is 0, the leaves are 0, 0, 2. That means the numbers there are 00, 00, and 02, which are 0, 0, and 2. Similarly, for the stem 1, the leaves are 4 and 5, so the numbers are 14 and 15.\n\nNow, the question is asking for the largest number of sit-ups. To find that, I need to look for the highest value in the leaf column because each leaf corresponds to a single digit. Scanning through the leaves, I see numbers like 0, 0, 2, 4, 5, 1, 2, 5, 5, 8, 3, 2, 6. The highest number here is 8.\n\nBut wait, I should make sure that 8 is indeed the highest. Let me check all the leaves again. The highest number I see is 8, which is in the row where the stem is 8. So, the number is 88. That would be the largest number of sit-ups done.\n\nI don't think there's any higher number beyond 88 because the next highest leaf after 8 is 3, which is in the row with stem 7. So, 88 is the maximum value in the dataset.\n\nTherefore, the largest number of sit-ups done is 88.\n</think>\n\n\nThe largest number of sit-ups done is 88.\n\n\nAnswer: 88

R1-Onevision+SPLIT

- <think>\nOkay, so I need to figure out the largest number of sit-ups done by the students based on the provided stem-and-leaf plot. Let me start by understanding what a stem-and-leaf plot is. It's a way to display quantitative data where each data point is split into a \"stem\" (the leading digit or digits) and a \"leaf\" (the trailing digit).\n\nLooking at the image, the stem represents the tens place, and the leaf represents the ones place. So, for example, in the row where the stem is 0, the leaves are 0, 0, 2. That means the numbers there are 00, 00, and 02, which are 0, 0, and 2. Similarly, for the stem 1, the leaves are 4 and 5, so the numbers are 14 and 15.\n\nNow, I need to find the largest number. To do that, I should look for the highest stem and then the highest leaf in that stem. The stems go from 0 up to 8, so the highest stem is 8. In the row where the stem is 8, the leaves are 2, 3, and 6. So the numbers there are 82, 83, and 86.\n\nComparing these, 86 is the highest. Therefore, the largest number of sit-ups done is 86.\n</think>\n\n\nThe largest number of sit-ups done is 86.\n\n\nAnswer: 86



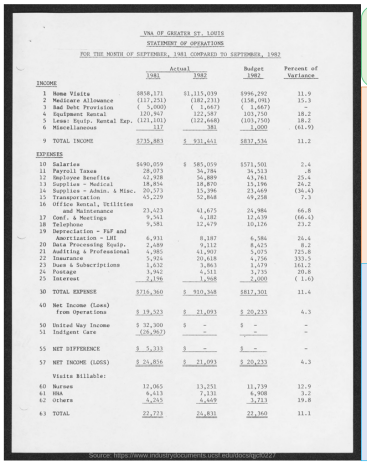
- Subtract all large red rubber blocks. Subtract all tiny red matte objects. How many objects are left?

Qwen3-VL-Thinking

- Got it, let's see. The problem says to subtract all large red rubber blocks and all tiny red matte objects. First, let's identify the objects in the scene. There's a large red metallic cube (so not rubber), a tiny yellow cube, and a tiny red matte cube. \n\nFirst, subtract large red rubber blocks. But the large red cube is metallic, not rubber, so that doesn't get subtracted. Then subtract tiny red matte objects. The tiny red cube is matte, so that gets subtracted. Now, what's left? The yellow cube. So count the remaining objects: 1.\n</think>\n\n1

Qwen3-VL-Thinking+SPLIT

- Got it, let's see. The problem says to subtract all large red rubber blocks and all tiny red matte objects. First, let's identify the objects in the scene. There's a large red metallic cube (so not rubber), a tiny yellow cube, and a tiny red matte cube. \n\nWait, the large red rubber blocks—there's no large red rubber block here. The large red cube is metallic, so it's not rubber. Then the tiny red matte objects: there's one tiny red cube, which is matte. So we need to subtract those. \n\nOriginal objects: 3 (large red metal, tiny yellow, tiny red matte). Subtract large red rubber blocks: none, so still 3. Subtract tiny red matte objects: there's one, so 3 - 1 = 2. Wait, but let's check again. The large red cube is metallic, so not rubber, so it's not subtracted. The tiny red matte cube is subtracted. The yellow cube is not red, so not subtracted. So remaining objects: large red metal (1) and yellow cube (1). So total 2.\n</think>\n\n\n2



- Which year has more actual total income?

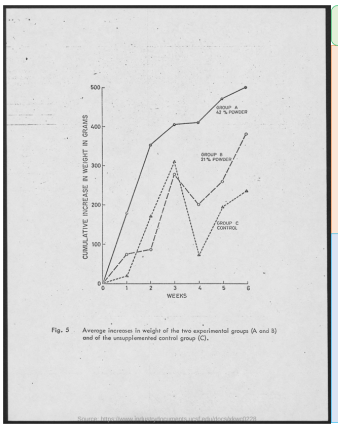
Qwen3-VL-Thinking

- Got it, let's look at the \"TOTAL INCOME\" row. The actual values for 1981 and 1982 are $7,358,883 and $931,441 respectively. Wait, 7,358,883 is bigger than 931,441. So 1981 has more actual total income.\n</think>\n\n\n1981

Qwen3-VL-Thinking+SPLIT

- Got it, let's look at the \"TOTAL INCOME\" row. The actual values for 1981 and 1982 are $735,883 and $931,441 respectively. Comparing these two numbers, 931,441 is larger than 735,883. So 1982 has more actual total income.\n</think>\n\n\n1982

Figure 9: **R1-OneVision and Qwen3-VL-Thinking on MathVista.** From top to bottom, SPLIT identifies the smallest bar as 20, reads the final stem-and-leaf entry as 86, retains the large metallic cube to return 2, and selects 1982 after recovering the correct income entries.



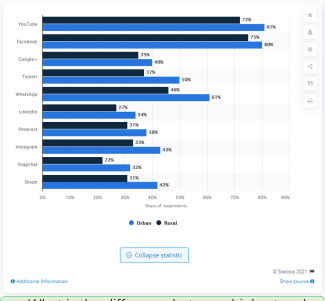
- For Group C, in which week is the cumulative increase in weight , the highest?

InternVL3.5

- <think>\nSo, looking at the graph for Group C, which is the dashed line with triangle markers, I need to find the week where the cumulative increase in weight is the highest. Let's analyze the data points for Group C over the weeks.\n\nFrom the graph, the data points for Group C are as follows:\n- Week 0: 0 grams\n- Week 1: Approximately 10 grams\n- Week 2: Approximately 100 grams\n- Week 3: Approximately 300 grams (peak)\n- Week 4: Approximately 200 grams\n- Week 5: Approximately 250 grams\n- Week 6: Approximately 350 grams\n\nThe highest point on the graph for Group C seems to be around Week 6, where the cumulative increase reaches approximately 350 grams. This is the highest value compared to the other weeks.\n\nTherefore, the week with the highest cumulative increase in weight for Group C is Week 6.\n</think>\n\n6

InternVL3.5+SPLIT

- <think>\nSo, looking at the graph for Group C, which is the dashed line with triangle markers, I need to find the week where the cumulative increase in weight is the highest. Let's analyze the data point by data point.\n\nAt Week 0, the weight increase is 0 grams. Then, in Week 1, it's still around 0. By Week 2, there's a slight increase, but it's not significant. In Week 3, there's a noticeable spike, reaching the highest point on the graph for Group C. After Week 3, the weight increase starts to decline, hitting a low around Week 4, and then gradually increasing again towards Week 6.\n\nThe highest point for Group C is clearly at Week 3, where the cumulative increase peaks before declining again. So, the answer should be Week 3.\n</think>\n\nThe highest cumulative increase in weight for Group C occurs in Week 3.\n\n3



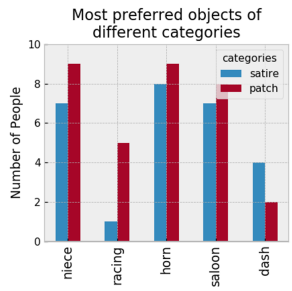
- What is the difference between highest and lowest value of dark blue bar?

InternVL3.5

- <think>\nSo, looking at the dark blue bars, which represent the Rural share of respondents, I need to find the highest and lowest values. The highest value for the dark blue bar is 80% for Facebook, and the lowest is 22% for Snapchat. To find the difference, I subtract the lowest from the highest: 80% minus 22% equals 58%. So the difference is 58 percentage points.\n</think>\n\nThe difference between the highest and lowest value of the dark blue bar (Rural) is calculated by subtracting the smallest value from the largest value. \n\n- **Highest value (Rural):** 80% (Facebook)\n- **Lowest value (Rural):** 22% (Snapchat)\n\n**Calculation:** \n80% - 22% = 58%\n\n**Answer:** 58

InternVL3.5+SPLIT

- <think>\nSo, looking at the dark blue bars, which represent the Rural share of respondents, I need to find the highest and lowest values. The highest value for Rural is 75% for Facebook, and the lowest is 22% for Snapchat. To find the difference, I subtract the lowest from the highest: 75% - 22% equals 53%. So the difference is 53 percentage points.\n</think>\n\nThe difference between the highest and lowest value of the dark blue bar (Rural) is calculated by subtracting the smallest value from the largest value. \n\n- **Highest value (Rural):** 75% (Facebook)\n- **Lowest value (Rural):** 22% (Snapchat)\n\n**Calculation:** \n75% - 22% = 53%\n\n**Answer:** 53



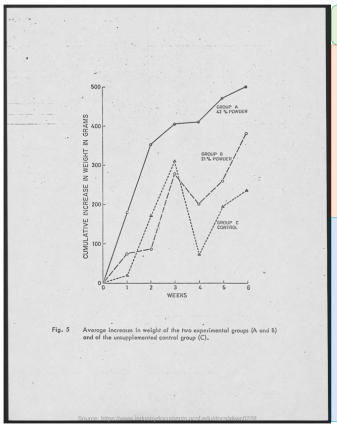
- How many objects are preferred by more than 7 people in at least one category?

InternVL3.5

- <think>\nSo, looking at the bar chart, I need to figure out how many objects have at least one category where the preference is more than 7 people. Let's go through each object one by one.\n\nFirst, there's \"niece.\" The categories are \"satire\" and \"patch.\" For \"satire,\" niece has 7 people, and for \"patch,\" it's 9. So, the patch category here is more than 7. That counts as one.\n\nNext is \"racing.\" Satire shows 1 person, and patch is 5. Neither of these are more than 7, so racing doesn't count.\n\nThen there's \"horn.\" Satire is 8, and patch is 9. Both are more than 7, but since we're looking for at least one category, horn counts here too. That's the second object.\n\nMoving on to \"saloon.\" Satire is 7, and patch is 7. Neither is more than 7, so saloon doesn't make the cut.\n\nLastly, \"dash.\" Satire is 4, and patch is 2. Again, neither exceeds 7, so dash isn't counted.\n\nAdding up the ones that qualify: niece and horn. That makes a total of 2 objects where at least one category has more than 7 preferences.\n</think>\n\n2

InternVL3.5+SPLIT

- <think>\nSo, looking at the bar chart, I need to figure out how many objects have at least one category where the preference is more than 7 people. Let's go through each object one by one.\n\nFirst, there's \"niece.\" The categories are \"satire\" and \"patch.\" For \"satire,\" the bar goes up to 7, and for \"patch,\" it's 9. So, the \"patch\" category here is more than 7. That counts as one.\n\nNext is \"racing.\" The \"satire\" category shows a bar at 1, and \"patch\" is at 5. Neither of these are more than 7, so \"racing\" doesn't count.\n\nThen there's \"horn.\" The \"satire\" category is at 8, and \"patch\" is also at 9. Both are more than 7, but since we're looking for at least one category, this definitely counts. That's the second object.\n\nMoving on to \"saloon,\" the \"satire\" category is at 7, and \"patch\" is at 8. The \"patch\" category here is more than 7, so \"saloon\" counts as well. That's the third object.\n\nLastly, \"dash\" has \"satire\" at 4 and \"patch\" at 2. Neither of these are more than 7, so \"dash\" doesn't count.\n\nAdding up the ones that qualify: niece, horn, and saloon. That makes a total of 3 objects where at least one category has a preference of more than 7 people.\n</think>\n\n3



- What is cumulative increase in weight (in grams) for \"GROUP A\" in third week (give an approximate value) ?

InternVL3.5

- <think>\nSo, looking at the graph, I need to find the cumulative increase in weight for \"GROUP A\" in the third week. Let's break this down step by step.\n\nFirst, I identify the line for \"GROUP A\" on the graph. It's the solid line with circles, labeled \"42% POWDER\" and \"GROUP A\".\n\nNext, I look at the x-axis, which represents weeks, and find the third week (week 3).\n\nNow, I trace vertically from the point at week 3 on the \"GROUP A\" line up to the y-axis, which shows the cumulative increase in weight in grams.\n\nThe y-axis has increments, and the point for \"GROUP A\" at week 3 seems to align with approximately 300 grams.\n\nSo, the approximate cumulative increase in weight for \"GROUP A\" in the third week is 300 grams.\n</think>\n\n300

InternVL3.5+SPLIT

- <think>\nSo, looking at the graph, I need to find the cumulative increase in weight for \"GROUP A\" in the third week. Let's break this down step by step.\n\nFirst, I identify the line for \"GROUP A\" on the graph. It's the solid line with circles, labeled \"42% POWDER\" and \"GROUP A\".\n\nNext, I look at the x-axis, which represents weeks, and find the third week (week 3).\n\nNow, I trace vertically from the point at week 3 on the \"GROUP A\" line up to the y-axis, which shows the cumulative increase in weight in grams.\n\nThe y-axis has increments, and the point at week 3 for \"GROUP A\" seems to align with approximately 400 grams.\n\nSo, the approximate cumulative increase in weight for \"GROUP A\" in the third week is 400 grams.\n</think>\n\n400

Figure 10: **InternVL3.5 on MathVista.** From top to bottom, SPLIT locates Group C's peak at week 3, computes the rural-range difference as $75 - 22 = 53$, counts all three qualifying objects, and traces Group A at week 3 to approximately 400 grams.

E.2 Fine-Grained Hallucination Decisions

Figures 11 and 12 use Vision-R1 to cover existence, shape, OCR, relative size, and counterfactual object removal. The vanilla traces exhibit semantic completion, such as normalizing "WHEST" to "WEST" or inferring an object from its packaging despite contradictory visual evidence. SPLIT uses squared mode loadings to identify the current carriers, then reallocates their load toward the rest of the contemporaneous head population. The corrected traces recover literal evidence in both positive and negative decisions rather than shifting uniformly toward either answer class.



- Are the clothes visible in the picture?

Vision-R1

- <think> Okay, let's see. The user is asking if the clothes are visible in the picture. The image provided displays that there's a blanket draped over the arm of the sofa, but it specifically displays that there are no other visible articles of clothing. So, the blanket is there, but that's not considered clothing. The answer should be no. </think><answer> Final Answer:No </answer>

Vision-R1+SPLIT

- <think> Okay, let's see. The user is asking if the clothes are visible in the picture. The image provided displays that there's a piece of clothing or fabric draped over the arm of the sofa on the right side. So, according to the details given, yes, the clothes are visible. The answer should be yes. </think><answer> Final Answer:Yes </answer>



- Does the cake in the picture have the shape of a 3/4 cylinder?

Vision-R1

- <think> Okay, let's see. The user is asking if the cake in the picture has the shape of a 3/4 cylinder. The image provided displays that the cake is a slice of a multi-layered cake with a rectangular shape. It also displays that the cake does not have the shape of a 3/4 cylinder. So, based on the given information, the answer should be no. </think><answer> Final Answer:No </answer>

Vision-R1+SPLIT

- <think> Okay, let's see. The user is asking if the cake in the picture has the shape of a 3/4 cylinder. The image provided displays that the cake is a slice of a multi-layered cake with a rectangular shape and a triangular missing section, which indicates it's a 3/4 cylinder. The key part here is \"3/4 cylinder\" mentioned in the image. So, the answer should be yes. </think><answer> Final Answer:Yes </answer>



- Is the rightmost English word located in the third row in the picture written as \"WEST\"?

Vision-R1

- <think> Okay, let's see. The user is asking if the rightmost English word in the third row is written as \"WEST\". According to the image provided, the image has a signpost with different directions. The third row from the top has the word \"WHEST\" on the rightmost sign. The image specifically displays that the rightmost word in the third row is \"WHEST\" and notes that it's a variation of \"WEST\". So, even though it's spelled slightly differently, the key here is that it's a variation of \"WEST\". The question is asking if the word is written as \"WEST\", and since it's a variation, the answer should be yes. </think><answer> Final Answer:Yes </answer>

Vision-R1+SPLIT

- <think> Okay, let's see. The user is asking if the rightmost English word in the third row is written as \"WEST\". According to the image provided, the image has a signpost with different directions. The third row from the top has the word \"WHEST\" on the rightmost sign. The image specifically displays that the rightmost word in the third row is \"WHEST\". Now, the question is whether that word is written as \"WEST\". Wait, \"WHEST\" doesn't look like \"WEST\". But maybe there's a typo or a transformation here. The image displays displays the rightmost word in the third row is \"WHEST\". The user is asking if it's written as \"WEST\". Hmm, maybe the image has a mistake? Or perhaps the image is in a different language where \"WHEST\" could be a corruption of \"WEST\". For example, if the signpost uses a different script or has some decorative elements that might alter the letters. But the image doesn't mention any such alterations. It just displays the word is \"WHEST\". So according to the given information, the answer would be no, because the word is \"WHEST\" not \"WEST\". But wait, the user is asking if it's written as \"WEST\". So the answer is no. But wait, maybe the image has a typo? Let me check again. The image displays displays: \"The rightmost English word in the third row is 'WHEST'.\" So the answer is based on that. The user is asking if it's written as \"WEST\". Since the image displays displays it's \"WHEST\", the answer is no. But maybe the image is in a different language where \"WHEST\" could be a different word. For example, in some languages, words are written differently. But the image doesn't provide any context about the language. It just displays the word is \"WHEST\". So the answer should be no. </think><answer> Final Answer:No </answer>



- Is the plant that is the second shortest the fourth one from the left in the row?

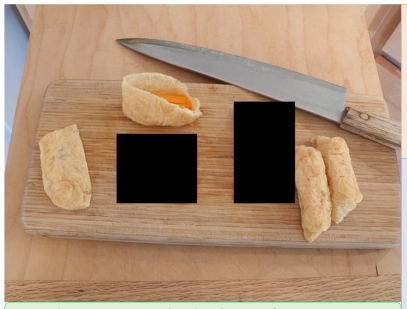
Vision-R1

- <think> Okay, let's see. The question is asking if the plant that is the second shortest is the fourth one from the left in the row. The image displays displays there are five potted plants in a row. The heights from shortest to tallest are: second shortest is the third plant from the left. Wait, the image displays displays, \"The second shortest plant is the third one from the left in the row.\" So the second shortest is the third plant. The question is asking if the second shortest is the fourth one from the left. But according to the given info, the second shortest is the third. So the answer should be no. </think><answer> Final Answer:No </answer>

Vision-R1+SPLIT

- <think> Okay, let's see. The question is asking if the second shortest plant is the fourth one from the left. The image displays displays there are five potted plants in a row. The heights vary, with the second shortest being the fourth from the left. So, the second shortest is the fourth one. Therefore, the answer should be yes. </think><answer> Final Answer:Yes </answer>

Figure 11: **Vision-R1 on VHTest.** From top to bottom, SPLIT recognizes the draped garment, uses the missing cake sector to identify the $3/4$-cylinder, reads "WHEST" literally, and locates the second-shortest plant fourth from the left.



- Is there a carrot in the image?

Vision-R1

- <think> Okay, let's see. The user is asking if there's a carrot in the image. The image provided displays that there's a piece of orange-colored food that looks like a carrot. It's in a pastry shell, possibly a carrot-filled pastry. The other items are the knife and the cutting board. The key detail here is the orange-colored food resembling a carrot. Since the image explicitly displays that, even though it's in a pastry, the color and shape are similar to a carrot. So the answer should be yes. </think><answer> Final Answer:Yes </answer>

Vision-R1+SPLIT

- <think> Okay, let's see. The user is asking if there's a carrot in the image. The image provided displays that there are bread-like items, possibly cheese or similar, and a knife on a wooden cutting board. It specifically displays that the items are light-colored and have a texture like bread or pastry. The knife is next to them. Then, the image notes that there are no visible orange or carrot-like features. So, the key points here are the absence of orange color and any carrot-like features. Since carrots are typically orange and have a distinct shape, the image explicitly displays there's none of that. Therefore, the answer should be no. </think><answer> Final Answer:No </answer>



- Is there a sandwich in the image?

Vision-R1

- <think> Okay, let's see. The user is asking if there's a sandwich in the image. The image provided displays a Coca-Cola cup with a lid and straw, and some white paper wrappers. The key detail here is that the white paper wrappers are \"typical of those used for sandwiches or similar food items.\" The image also displays that the arrangement suggests the presence of a sandwich. So, even though the sandwich isn't explicitly shown, the context clues point towards it. The white paper wrappers are commonly associated with sandwiches, and the image's setup implies that the sandwich is likely there. Therefore, the answer should be yes. </think><answer> Final Answer:Yes </answer>

Vision-R1+SPLIT

- <think> Okay, let's see. The user is asking if there's a sandwich in the image. The image provided displays a Coca-Cola cup with a lid and straw, and some white paper or napkins. It specifically displays that there are no visible sandwiches or food items besides the cup and the paper. So, based on that information, the answer should be no. </think><answer> Final Answer:No </answer>



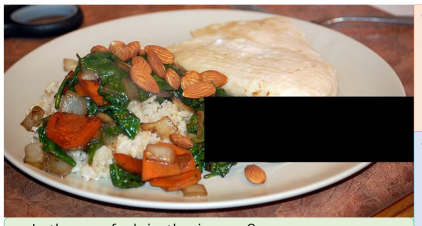
- Is there a bowl in the image?

Vision-R1

- <think> Okay, let's see. The user is asking if there's a bowl in the image. The image provided displays that in the background, there are objects on a table, including a green bowl and a red bowl. So, according to the information given, yes, there are bowls present. The answer should be straightforward. </think><answer> Final Answer:Yes </answer>

Vision-R1+SPLIT

- <think> Okay, let's see. The user is asking if there's a bowl in the image. The image provided displays that there are no visible bowls on the table or elsewhere in the image. The items listed include plates, a book, a red ribbon, and a tag, but specifically notes that there are no bowls. So based on the information given, the answer should be no. </think><answer> Final Answer:No </answer>



- Is there a fork in the image?

Vision-R1

- <think> Okay, let's see. The user is asking if there's a fork in the image. The image provided displays that in the background, there's a fork partially visible, with its handle extending towards the top left and the tines towards the center. So, based on that information, the answer should be yes. The fork is mentioned as being partially visible, so even though part of it is cut off in the image, it's still present. The key details here are the handle and the tines, which confirm it's a fork. </think><answer> Final Answer:Yes </answer>

Vision-R1+SPLIT

- <think> Okay, let's see. The user is asking if there's a fork in the image. The image provided displays that the image shows a plate with chicken, rice, vegetables, and almonds. It specifically displays that there are no visible utensils like a fork, knife, or spoon in the image. The focus is on the food, and the background is a wooden surface. So, based on the information given, the answer should be no. </think><answer> Final Answer:No </answer>

Figure 12: **Vision-R1 on VHTest.** SPLIT grounds existence in visible evidence. From top to bottom, it rejects context-driven hallucinations of a carrot, sandwich, bowl, and partially visible fork after the queried objects have been removed or occluded.

E.3 Open-Ended Grounded Description

Figure 13 shows ThinkLite-VL on AMBER correcting the water level, device arrangement, visible orange portions, and lioness–vehicle relation while preserving the surrounding scene description. Carrier redistribution weakens the unsupported attributes and relations, while radial retraction preserves reference innovation energy and keeps supported content available. Supported details remain in the response after the corrections.



Figure 13: **ThinkLite-VL on AMBER.** From top to bottom, SPLIT corrects the water level, replaces an unsupported "neat row" with the visible device arrangement, describes the visible orange portions without adding a whole fruit, and grounds the lioness's orientation relative to the vehicles.

E.4 Failure Cases and Scope

Figure 14 isolates two residual errors at different interfaces. In the VHTest example, the target train is removed, but rails, overhead wires, and steam remain jointly predictive of it; both traces therefore convert correlated context into an existence decision. This is an evidence-identifiability boundary: after counterfactual masking, the retained scene remains observationally compatible with an occluded train, so population coordination alone cannot determine whether context is residual evidence or a cue to the removed target. In the MathVista example, SPLIT correctly reads all four percentages and explicitly orders $57\% > 21\% > 15\% > 7\%$, yet later binds "second largest" to 15% and returns $7/15 = 0.47$ instead of $7/21 = 1/3$. The visual premise is recovered, but a downstream symbolic-consistency error survives. The first failure arises before a reliable target premise is available, whereas the second occurs after that premise has been established; a uniformly stronger gain would conflate the two error sources.



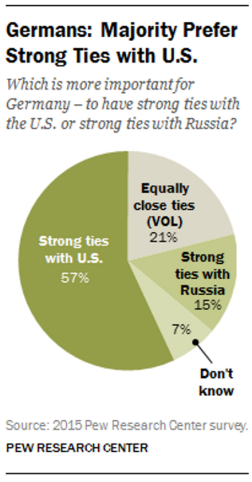
- Is there a train in the image?

Vision-R1

- <think> Okay, let's see. The user is asking if there's a train in the image. The image displays displays that there's a black rectangular area in the center, which is likely a train. The tracks are visible, and the black area is positioned where a train would be. Also, the context suggests a train is present. So even though the train isn't fully visible, the clues point to yes. </think><answer> Final Answer:Yes </answer>

Vision-R1+SPLIT

- <think> Okay, let's see. The user is asking if there's a train in the image. The image displays displays that there's a black rectangular area in the center, which is likely a train. The tracks are visible, and the black area is positioned along them. Also, there's smoke or steam coming from that area, which is typical for a steam train. The surroundings include railway infrastructure like overhead wires and signs. The presence of the black area on the tracks with smoke suggests that yes, there is a train there. Even though the image is partially obscured, the details point to a train being present. </think><answer> Final Answer:Yes </answer>



- What's the ratio of smallest segment and second largest segment?

R1-Onevision

- <think>\nOkay, so I need to figure out the ratio of the smallest segment to the second largest segment in the pie chart. Let me start by understanding the problem.\n\nFirst, the image shows a pie chart with four segments. The segments are labeled as follows: \"Strong ties with U.S.\" at 57%, \"Strong ties with Russia\" at 15%, \"Equally close ties (VOL)\" at 21%, and \"Don't know\" at 7%.\n\nThe question asks for the ratio of the smallest segment to the second largest segment. So, I need to identify which segments are the smallest and the second largest.\n\nLooking at the percentages: 57%, 15%, 21%, and 7%. The smallest segment is clearly the one with 7%, which is \"Don't know.\" The next smallest is 15%, which is \"Strong ties with Russia.\" So, the second largest segment is 15%.\n\nNow, to find the ratio, I take the smallest segment (7%) divided by the second largest (15%). That gives me 7/15. Simplifying that fraction, 7 divided by 15 is approximately 0.4666...\n\nRounding to two decimal places, that's 0.47. So, the ratio is approximately 0.47.\n\nWait, let me double-check. The smallest is 7%, and the second largest is 15%. Yes, 7 divided by 15 is indeed 0.4666..., which rounds to 0.47. So, the ratio is 0.47.\n</think>\n\nThe ratio of the smallest segment (7%) to the second largest segment (15%) is calculated by dividing 7 by 15, resulting in approximately 0.47.\n\n**Answer:** 0.47

R1-Onevision+SPLIT

- <think>\nOkay, so I need to figure out the ratio of the smallest segment to the second largest segment in this pie chart. Let me start by understanding the data provided.\n\nThe image shows four segments: \"Strong ties with U.S.\" at 57%, \"Strong ties with Russia\" at 15%, \"Equally close ties (VOL)\" at 21%, and \"Don't know\" at 7%.\n\nFirst, I should identify the smallest segment. As seen in the image, the smallest segment is \"Don't know\" at 7%.\n\nNext, I need to find the second largest segment. The largest segment is \"Strong ties with U.S.\" at 57%, followed by \"Equally close ties (VOL)\" at 21%, then \"Strong ties with Russia\" at 15%, and the smallest is \"Don't know\" at 7%.\n\nSo, the second largest segment is \"Strong ties with Russia\" at 15%.\n\nNow, to find the ratio of the smallest segment to the second largest, I take the value of the smallest segment divided by the value of the second largest segment. That would be 7% divided by 15%.\n\nCalculating that, 7 divided by 15 equals approximately 0.466666...\n\nRounding that to two decimal places, it becomes 0.47.\n\nTherefore, the ratio is 0.47.\n</think>\n\nThe ratio of the smallest segment (\"Don't know\" at 7%) to the second largest segment (\"Strong ties with Russia\" at 15%) is calculated by dividing 7% by 15%, resulting in approximately 0.47.\n\n**Answer:** 0.47

Figure 14: **Failure modes at two interfaces.** Top: correlated railway evidence sustains an incorrect positive decision after the train is removed. Bottom: SPLIT recovers the chart values and their order, but a later binding error substitutes 15% for the second-largest 21% segment.