SEEING BEFORE BELIEVING: MITIGATING HALLUCINATIONS IN MULTIMODAL REASONING VIA VISUALLY-ANCHORED LATENT RECTIFICATION

Anonymous authors
Paper under double-blind review

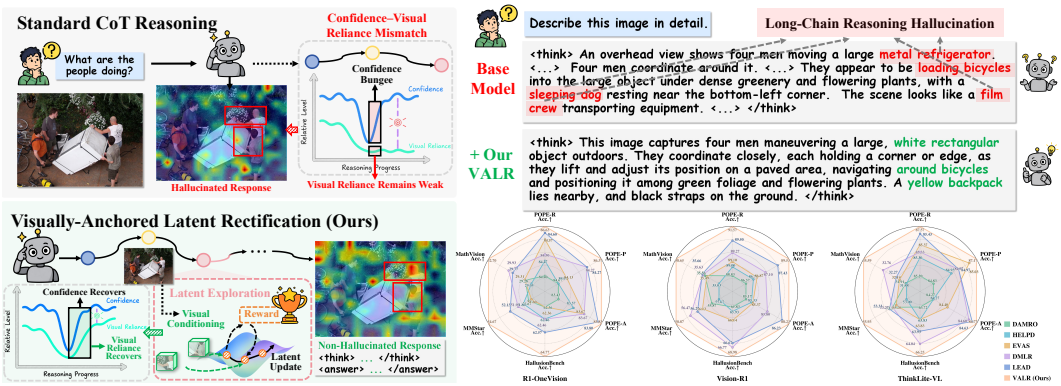


Figure 1: **Confidence–visual reliance mismatch in multimodal reasoning. Left:** Confidence rebounds in both trajectories, but visual reliance remains weak in the hallucinated case (top) and recovers in the non-hallucinated case (bottom). **Upper right:** Hallucinated reasoning versus a visually supported response from VALR. **Bottom right:** Performance gains across models and benchmarks.

ABSTRACT

Although long-chain reasoning aids complex multimodal tasks, the extended thinking trajectory can drift from image evidence and produce hallucinated answers. Recent work attempts to rectify reasoning by optimizing latent states guided by reference-free predictive confidence. However, predictive confidence becomes misleading when greater certainty reflects linguistic predictability rather than visual faithfulness. When reasoning detaches from the image, self-reinforcing language priors can concentrate probability on predictable continuations, producing a bungee-like rebound in confidence and a false sense of certainty without visual support. Optimizing against confidence alone thus promotes visually ungrounded shortcuts, amplifying such high-confidence errors. To address this, we introduce Visually-Anchored Latent Rectification (VALR), a training-free method that uses visual conditioning to guide confidence-based latent refinement. VALR dynamically retrieves and fuses aligned image evidence into candidate states, shaping the reward landscape around visually induced confidence gains to guide exploration and select the state used for answer generation. Across three models, VALR outperforms all compared inference-time methods in hallucination mitigation, while its benefits extend to broader multimodal understanding and mathematical reasoning.

1 INTRODUCTION

Multimodal large reasoning models (MLRMs) approach complex visual problems by developing explicit chains of intermediate reasoning Bai et al. (2025); Qwen Team (2025); Kimi Team et al. (2025); Yang et al. (2025). These chains allow a model to break a question into intermediate steps, connect observations, and refine its prediction as reasoning unfolds. For visual problems, reliable conclusions also depend on whether the observations used along the way are supported by the image. Yet even strong MLRMs can produce coherent responses that contradict the image Tong et al. (2024); Guan et al. (2024); Tian et al. (2026); Xu et al. (2025c). A reasoning chain can therefore remain plausible after its use of the image has weakened, leaving a hallucinated answer difficult to recognize from language alone.

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.

Recent work explores alternative internal states at inference time to improve reasoning Li et al. (2025); Xu et al. (2025b). Model confidence provides a reference-free signal for evaluating these states and guiding their refinement Ye et al. (2026); Liu et al. (2026). Yet language priors can favor predictable continuations of an unsupported reasoning path, raising confidence without renewed image support. We trace confidence and visual reliance along individual reasoning trajectories to examine this ambiguity (Section 3.1). During confidence recovery, correct trajectories typically increase their use of the image, whereas incorrect trajectories more often regain confidence without renewed visual reliance. Rising confidence can thus accompany either a return to image evidence or further development of an unsupported reasoning path, making predictive certainty an ambiguous guide to visual reasoning (Figure 1, left).

When confidence determines which candidate states are favored, increasing certainty can steer exploration along a visually unsupported direction. Confidence-only exploration improves overall accuracy yet increases high-confidence errors, whereas search with visual conditioning reduces them (Section 3.2). To examine how image content constrains predictions, we weaken and then restore the visual evidence available during reasoning (Section 3.3). Weakening visual support turns some previously correct predictions into errors. Restoring the original content corrects more of these predictions than restoring shuffled content, showing that specific image content can constrain the model's predictions.

To address this, we introduce Visually-Anchored Latent Rectification (VALR), a training-free method that uses image evidence to shape and refine latent reasoning states (Figure 1). Question-relevant visual content provides the starting point for exploration. For each sampled candidate, VALR dynamically retrieves and fuses aligned image evidence, using the confidence gain from this evidence as the exploration reward. As candidates evolve, retrieval adapts to their changing representations, keeping the reward tied to the contribution of the current evidence. This reward guides subsequent updates and selects the best visually conditioned state for answer generation.

Across three MLRM backbones, VALR consistently reduces hallucination on POPE Li et al. (2023) and HallusionBench Guan et al. (2024), outperforming the compared inference-time baselines. Its benefits also extend to multimodal understanding and mathematical reasoning, with improved accuracy on MMStar Chen et al. (2024) and MathVision Wang et al. (2024), respectively.

2 Related Work

2.1 Hallucination in Multimodal Reasoning

Reasoning can improve difficult visual inference, yet extended reasoning may weaken attention to the image and introduce perceptual errors Tian et al. (2026). Existing mitigation approaches intervene in output distributions or the visual information available during inference. VCD Leng et al. (2024) adjusts next-token predictions by contrasting output distributions under original and distorted visual inputs. MemVR Zou et al. (2025) reintroduces visual features into feed-forward layers when uncertainty is high. Within reasoning trajectories, Interleaved-Modal CoT Gao et al. (2025) interleaves visual tokens with text, and LEAD Xu et al. (2026) uses uncertainty to switch between latent and discrete decoding and inject visual anchors. For confidence-guided reasoning, we examine whether increasing certainty is accompanied by stronger image support.

2.2 Latent and Test-Time Intervention

Continuous reasoning methods such as Coconut Hao et al. (2025) and SoftCoT Xu et al. (2025a) use latent representations to carry intermediate computation without expressing every intermediate state as text. SoftCoT++ Xu et al. (2025b) further promotes diverse latent reasoning paths for test-time scaling. Other approaches control inference by steering internal representations Liu et al. (2025b;a) or switching between latent and explicit reasoning according to predictive entropy Shi et al. (2026). Input-specific optimization uses self-generated rewards Li et al. (2025) or confidence-based scores Ye et al. (2026); Liu et al. (2026) to guide latent updates, including multimodal search with visual interleaving. VALR uses candidate-aligned visual evidence to define the exploration objective. Question-relevant content initializes the latent states, while matched evaluations measure the confidence gain from retrieved evidence to guide updates and state selection. Appendix C.6 compares these mechanisms with related latent-space methods.

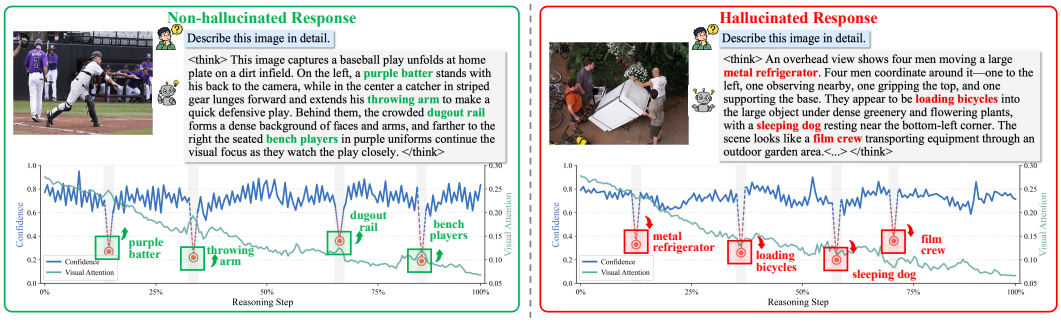


Figure 2: Confidence (blue) and visual-token attention (teal) along non-hallucinated and hallucinated image descriptions. Both trajectories regain confidence after a decline, but visual attention rises near the rebound in the non-hallucinated case and remains weak in the hallucinated case.

3 When Confidence Outruns Visual Grounding

Model confidence offers a reference-free signal for guiding multimodal reasoning. A recovery after an earlier decline indicates more concentrated predictions, which can make subsequent reasoning appear more reliable. Yet increasingly predictable continuations may still follow a reasoning path that lacks image support. We trace confidence and visual reliance to examine whether renewed certainty is accompanied by greater use of the image.

3.1 Confidence and Visual Reliance Can Diverge

We first examine how attention to the image changes as confidence recovers. In Figure 2, confidence recovers in both image descriptions, while visual-token attention rises near the rebound only in the non-hallucinated case.

To test whether this contrast extends beyond individual cases, we compare confidence and visual reliance across correct and incorrect trajectories. At step t, normalized inverse entropy measures confidence, while the visual-attention share serves as an attention-based proxy for visual reliance:

$$\mathrm{Conf}_t = 1 - \frac{\mathcal{H}(P_t)}{\log |\mathcal{Y}|}, \qquad \mathrm{VAR}_t = \frac{A_t^{\mathrm{vis}}}{A_t^{\mathrm{all}}}. \tag{1}$$

Here, P_t is the next-token distribution over vocabulary $\mathcal{Y}$, $\mathcal{H}$ denotes entropy, and the A_t terms denote visual and total attention. Together, these measures distinguish predictive concentration from relative visual-token attention along the reasoning trajectory.

We randomly select 500 confidence-declining trajectories produced by Vanilla R1-OneVision (Yang et al., 2025) on HallusionBench (Guan et al., 2024) and group them by final-answer correctness. Measurement details appear in Appendices B.1 and B.4.

Correct and incorrect trajectories recover confidence at similar rates but differ in visual reliance (Table 1). Among recovered trajectories, 63.8% of incorrect cases show no visual-reliance increase, versus 30.8% of correct cases. Across three backbones on HallusionBench and MathVision, mean reliance rises in recovered correct trajectories but falls in incorrect ones (Appendix Table 9). Confidence recovery can make incorrect reasoning seem reliable without renewed image support.

Table 1: Confidence recovery and visual-reliance changes in correct and incorrect trajectories.

Measure	**Correct**	**Incorrect**
Declining trajectories	293	207
Confidence recovered	201 (68.6%)	149 (72.0%)
Among recovered trajectories		
Δ confidence	$+0.039$	$+0.047$
Δ visual reliance	$+0.010$	-0.003
No reliance increase	**30.8%**	**63.8%**

3.2 Confidence Alone Is an Unreliable Exploration Signal

The preceding analysis shows that higher confidence can accompany incorrect reasoning without renewed visual reliance. Selecting states by confidence may therefore favor certainty without improving image support. We compare Vanilla reasoning, confidence-only exploration, and search with visual conditioning. Both exploration methods retain the original image input.

On HallusionBench, we measure confidence in the predicted answer $\hat{a}_i \in \{\text{Yes}, \text{No}\}$ by normalizing its probability over the two alternatives:

$$c_i = \frac{P(\hat{a}_i \mid x_i)}{P(\text{Yes} \mid x_i) + P(\text{No} \mid x_i)}. \tag{2}$$

Here, x_i contains the image, question, and preceding reasoning, and P scores the complete candidate string. We report the fraction of all 951 image-based questions answered incorrectly with $c_i \geq 0.90$. Additional results and scoring details appear in Appendix B.2.

Despite improving overall accuracy (Appendix Table 10), confidence-only exploration increases high-confidence errors across all three backbones, whereas visual conditioning lowers them below Vanilla levels (Table 2). On R1-OneVision, the rate falls from 22.3% to 14.2%. The ambiguity thus extends to exploration: favoring more confident candidates can increase highly confident but incorrect answers.

Table 2: High-confidence errors on HallusionBench ($c_i \geq 0.90$).

Backbone	Vanilla	Conf. only	+ Vis. cond.
R1-OneVision	18.5	22.3	14.2
Vision-R1	16.8	20.7	12.4
ThinkLite-VL	18.3	22.8	14.6

3.3 Visual Evidence Functionally Constrains Predictions

To examine how visual content affects predictions, we intervene during reasoning on HallusionBench samples that R1-OneVision originally answers correctly. We suppress visual-token attention at a selected step, then test whether restoring visual content recovers the predictions. The induced hallucination rate (IHR) measures the fraction of this original cohort that becomes hallucinated.

If the induced errors simply reflect a general disturbance, non-visual suppression or norm-matched noise should have similar effects. These controls and visual suppression at a random reasoning step induce hallucinations in 2.1–4.6% of the cohort. In contrast, selected-step visual suppression yields 22.6% (Table 3).

Table 3: Induced hallucination under visual interventions.

Intervention	IHR (%)
No intervention	0.0
Suppression and controls	
Visual (selected step)	22.6
Visual (random step)	3.8
Non-visual suppression	2.1
Norm-matched noise	4.6
Restoration after suppression	
Shuffled visual content	**7.4**
Original visual content	**3.4**

To isolate the role of restored content, we compare real and shuffled restoration after the same suppression on the same originally correct cohort. Real restoration leaves fewer hallucinations than shuffled restoration (3.4% versus 7.4%, exact McNemar $p = 0.00061$). This difference shows that specific image content helps constrain predictions (Appendix B.3).

4 Visually-Anchored Latent Rectification

Section 3 shows that confidence can rise without renewed image support. VALR therefore bases exploration on the relative confidence change associated with retrieved visual content. The search starts from question-relevant evidence and retrieves aligned visual content as candidates evolve. Matched evaluation of each candidate then guides state updates and final selection (Figure 3).

4.1 Latent Candidate Exploration

To give exploration an image-specific starting direction, VALR initializes the latent state with visual evidence prioritized by the question. We represent this state as K continuous input embeddings appended to the multimodal context Ye et al. (2026); Liu et al. (2026). The context comprises text $\mathcal{T}$ and projected image embeddings $\mathcal{V} = \{v_j\}_{j=1}^{N}$, with $v_j \in \mathbb{R}^d$.

Question-to-image attention ranks visual tokens and distributes high-ranked evidence across the latent positions. Each position receives an attention-weighted visual context $v^q_{\text{ctx},i}$. With e_{lat} and v_{pad} denoting the frozen model's latent placeholder and visual anchor embeddings, we initialize each position as

$$h_i^{(0)} = \lambda e_{\text{lat}} + (1-\lambda)\,\text{NM}\big(v_{\text{pad}} + v^q_{\text{ctx},i}; \|e_{\text{lat}}\|_2\big)\,. \tag{3}$$

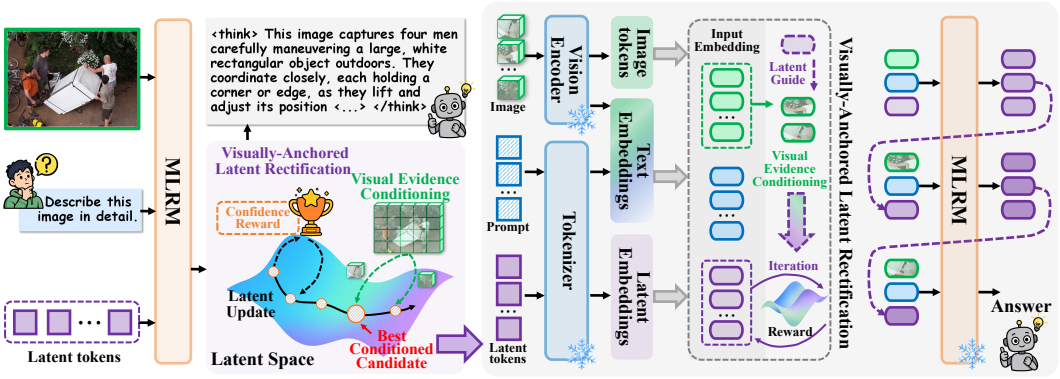


Figure 3: **Overview of VALR.** Question-relevant visual content initializes the search. Each candidate receives aligned image evidence and is evaluated against a matched control. Their entropy difference guides updates, and the best-scoring visually conditioned state is retained for generation.

Here, $\mathrm{NM}(x; r) = rx/\|x\|_2$ preserves the model-native latent scale, while λ balances the latent and visual contributions. The initialized positions form the search center $\mathcal{H}^{(0)} = [h_i^{(0)}]_{i=1}^K \in \mathbb{R}^{K\times d}$. Appendix C.2 specifies attention aggregation and token allocation.

Starting from this visually anchored center, local Gaussian perturbations explore alternative reasoning states within an input-specific neighborhood:

$$\begin{aligned} \widetilde{\mathcal{H}}^{(t)} &= \mathcal{H}^{(t)} + \epsilon^{(t)}, \qquad \epsilon^{(t)} \sim \mathcal{N}(0, \sigma^2 I), \\ \pi(\widetilde{\mathcal{H}}^{(t)} \mid \mathcal{H}^{(t)}) &= \mathcal{N}(\mathcal{H}^{(t)}, \sigma^2 I). \end{aligned} \tag{4}$$

We fix the perturbation scale σ relative to the initial-state magnitude, keeping exploration proportional to its scale. Only $\mathcal{H}$ is optimized, with the original image and prompt retained in context.

4.2 Visual Evidence Conditioning

Question-guided initialization identifies relevant evidence at the starting point. As the latent representation evolves, candidate-dependent retrieval refreshes its visual context to reflect the candidate's current semantics. Each latent position scores the full image pool, selects $\mathcal{N}_i^{(t)} = \mathrm{Top}_{\min(R,N)}(s_{ij}^{(t)})$, and aggregates the selected tokens as

$$s_{ij}^{(t)} = \frac{(\widetilde{h}_i^{(t)})^\top v_j}{\|\widetilde{h}_i^{(t)}\|_2 \, \|v_j\|_2}, \tag{5}$$

$$v_{\mathrm{ctx},i}^{(t)} = \sum_{j\in\mathcal{N}_i^{(t)}} \frac{\exp(s_{ij}^{(t)})}{\sum_{\ell\in\mathcal{N}_i^{(t)}} \exp(s_{i\ell}^{(t)})} \, v_j. \tag{6}$$

This allows each position to assemble evidence aligned with the current candidate while preserving contributions from multiple retrieved tokens. To keep fusion on the candidate's representation scale, we match the anchored visual context to its norm and form the conditioned state:

$$v_{\mathrm{lat},i}^{(t)} = \mathrm{NM}\Big(v_{\mathrm{pad}} + v_{\mathrm{ctx},i}^{(t)}; \|\widetilde{h}_i^{(t)}\|_2\Big), \tag{7}$$

$$\overline{\mathcal{H}}^{(t),+} = \lambda\widetilde{\mathcal{H}}^{(t)} + (1-\lambda)\mathcal{V}_{\mathrm{lat}}^{(t)}, \qquad \mathcal{V}_{\mathrm{lat}}^{(t)} = [v_{\mathrm{lat},i}^{(t)}]_{i=1}^K. \tag{8}$$

Confidence is therefore evaluated on a state shaped by evidence aligned with the current candidate.

4.3 Paired Confidence Evaluation and Latent Update

The confidence of a conditioned candidate reflects both the sampled state and the retrieved evidence. To isolate the relative contribution of that evidence, VALR compares the same proposal with a matched control that retains its state and visual anchor but omits the retrieved context:

$$\begin{aligned} b_i^{(t)} &= \lambda\widetilde{h}_i^{(t)} + (1-\lambda)\,\mathrm{NM}(v_{\mathrm{pad}}; \|\widetilde{h}_i^{(t)}\|_2), \\ \overline{h}_i^{(t),-} &= \mathrm{NM}(b_i^{(t)}; \|\overline{h}_i^{(t),+}\|_2). \end{aligned} \tag{9}$$

These embeddings form the control state $\overline{\mathcal{H}}^{(t),-} = [\bar{h}_i^{(t),-}]_{i=1}^K$. Both branches retain the original image, prompt, and visual information carried by the sampled state, while matched final norms keep their representation scales comparable.

Let $P_i^{(t),+}$ and $P_i^{(t),-}$ denote the conditioned and control next-token distributions at latent position i. To compare them over the same vocabulary entries, we form $\mathcal{U}_i^{(t)} = \mathrm{Top}_k(P_i^{(t),+}) \cup \mathrm{Top}_k(P_i^{(t),-})$ and renormalize both branches on this shared support:

$$q_i^{(t),\pm}(u) = \frac{P_i^{(t),\pm}(u)}{\sum_{v \in \mathcal{U}_i^{(t)}} P_i^{(t),\pm}(v)}, \qquad u \in \mathcal{U}_i^{(t)}. \tag{10}$$

The mean entropy reduction across latent positions defines the reward, using $\mathrm{H}(q) = -\sum_u q(u) \log q(u)$:

$$r^{(t)} = \mathcal{R}(\widetilde{\mathcal{H}}^{(t)}) = \frac{1}{K} \sum_{i=1}^{K} \left[\mathrm{H}(q_i^{(t),-}) - \mathrm{H}(q_i^{(t),+}) \right]. \tag{11}$$

A positive reward indicates greater predictive concentration under retrieved content than under the matched control. Candidates receive little reward when both branches have similar entropy, even if their absolute confidence is high.

We optimize the expected paired reward with a single-sample Gaussian score-function update (Williams, 1992):

$$\mathcal{J}(\mathcal{H}) = \mathbb{E}_{\widetilde{\mathcal{H}} \sim \pi(\cdot|\mathcal{H})}[\mathcal{R}(\widetilde{\mathcal{H}})],$$
$$\mathcal{H}^{(t+1)} = \mathcal{H}^{(t)} + \eta r^{(t)} \frac{\epsilon^{(t)}}{\sigma^2}. \tag{12}$$

The same paired criterion updates the search center and ranks evaluated candidates, aligning final selection with the evidence-induced confidence gain optimized during exploration. We retain the best evaluated candidate $\mathcal{H}^* = \overline{\mathcal{H}}^{(t^*),+}$, where $t^* = \arg\max_{0 \le t < N_{\mathrm{opt}}} r^{(t)}$. The retained state carries the visual content used in candidate evaluation into autoregressive generation, together with the original image and prompt. Appendix C provides the derivation and complete algorithm.

5 Experiments

Our experiments examine whether incorporating visual evidence into latent exploration leads to more reliable multimodal reasoning. We assess VALR on hallucination mitigation and broader reasoning tasks, then analyze the role of visual conditioning and the performance–cost trade-off of exploration.

5.1 Experimental Settings

Benchmarks. POPE (Li et al., 2023) evaluates object hallucination under Random, Popular, and Adversarial sampling, while HallusionBench (Guan et al., 2024) evaluates image-grounded consistency. MMStar (Chen et al., 2024) and MathVision (Wang et al., 2024) evaluate broader multimodal understanding and mathematical reasoning, respectively. We report accuracy throughout and F1 on POPE and HallusionBench, with Yes as the positive class. Prompts and implementation details appear in Appendices A and C.1. Percentage-point gains are calculated before rounding.

Backbones and baselines. Experiments cover the 7B versions of R1-OneVision (Yang et al., 2025), Vision-R1 (Huang et al., 2026), and ThinkLite-VL (Wang et al., 2025). We compare with VCD (Leng et al., 2024), DAMRO (Gong et al., 2024), HELPD (Yuan et al., 2024), EVAS (Zhang et al., 2025), MemVR (Zou et al., 2025), DMLR (Liu et al., 2026), and LEAD (Xu et al., 2026).

Implementation. VALR uses $K = 8$ visually initialized latent positions and retrieves $R = 16$ tokens per position. Paired scoring uses the union of each branch's top-k predictions, with $k = 16$. Unless stated otherwise, we set $N_{\mathrm{opt}} = 15$ and $\eta = 10^{-3}$ and use the 10% normalized perturbation scale in Equation 23. The validation-selected fusion weight λ is 0.70, 0.73, and 0.75 for R1-OneVision, Vision-R1, and ThinkLite-VL, respectively. The backbones remain frozen, and experiments use NVIDIA RTX PRO 6000 GPUs.

Category	Method	R1-OneVision				Vision-R1				ThinkLite-VL			
		Acc↑	Δ	F1↑	Δ	Acc↑	Δ	F1↑	Δ	Acc↑	Δ	F1↑	Δ
Random	Vanilla	83.47	+0.00	83.92	+0.00	86.93	+0.00	87.13	+0.00	84.27	+0.00	84.75	+0.00
	+VCD Leng et al. (2024)	84.20	+0.73	84.56	+0.64	88.37	+1.43	88.52	+1.39	85.00	+0.73	85.41	+0.65
	+DAMRO Gong et al. (2024)	84.40	+0.93	84.80	+0.88	88.83	+1.90	89.01	+1.88	85.20	+0.93	85.58	+0.83
	+HELPD Yuan et al. (2024)	84.47	+1.00	84.82	+0.90	89.00	+2.07	89.12	+1.99	85.30	+1.03	85.71	+0.96
	+EVAS Zhang et al. (2025)	84.57	+1.10	84.90	+0.99	89.10	+2.17	89.23	+2.10	85.37	+1.10	85.75	+1.00
	+MemVR Zou et al. (2025)	84.63	+1.17	84.97	+1.05	89.33	+2.40	89.46	+2.33	85.47	+1.20	85.84	+1.09
	+DMLR Liu et al. (2026)	84.50	+1.03	84.89	+0.97	89.27	+2.33	89.42	+2.29	85.33	+1.07	85.75	+1.00
	+LEAD Xu et al. (2026)	84.60	+1.13	84.92	+1.00	89.50	+2.57	89.59	+2.46	85.43	+1.17	85.84	+1.09
	+VALR (Ours)	**86.63**	**+3.17**	**86.77**	**+2.85**	**91.57**	**+4.63**	**91.54**	**+4.41**	**87.57**	**+3.30**	**87.76**	**+3.00**
Popular	Vanilla	83.13	+0.00	83.55	+0.00	84.83	+0.00	85.03	+0.00	83.93	+0.00	84.40	+0.00
	+VCD Leng et al. (2024)	83.70	+0.57	84.04	+0.49	86.23	+1.40	86.34	+1.31	84.63	+0.70	85.03	+0.63
	+DAMRO Gong et al. (2024)	84.07	+0.93	84.40	+0.85	86.57	+1.73	86.65	+1.62	84.83	+0.90	85.22	+0.82
	+HELPD Yuan et al. (2024)	84.07	+0.93	84.39	+0.84	86.77	+1.93	86.84	+1.81	84.93	+1.00	85.30	+0.89
	+EVAS Zhang et al. (2025)	84.13	+1.00	84.49	+0.93	86.87	+2.03	86.97	+1.94	85.03	+1.10	85.41	+1.01
	+MemVR Zou et al. (2025)	84.20	+1.07	84.54	+0.99	87.10	+2.27	87.20	+2.17	85.10	+1.17	85.49	+1.09
	+DMLR Liu et al. (2026)	84.27	+1.13	84.60	+1.05	87.10	+2.27	87.19	+2.16	84.97	+1.03	85.32	+0.92
	+LEAD Xu et al. (2026)	84.27	+1.13	84.58	+1.03	87.43	+2.60	87.50	+2.48	84.97	+1.03	85.33	+0.93
	+VALR (Ours)	**86.37**	**+3.23**	**86.48**	**+2.93**	**89.53**	**+4.70**	**89.49**	**+4.46**	**87.17**	**+3.23**	**87.34**	**+2.94**
Adversarial	Vanilla	82.67	+0.00	83.04	+0.00	83.47	+0.00	83.61	+0.00	83.40	+0.00	83.85	+0.00
	+VCD Leng et al. (2024)	83.27	+0.60	83.63	+0.59	84.60	+1.13	84.65	+1.04	84.13	+0.73	84.50	+0.64
	+DAMRO Gong et al. (2024)	83.43	+0.77	83.80	+0.76	85.17	+1.70	85.27	+1.66	84.27	+0.87	84.61	+0.75
	+HELPD Yuan et al. (2024)	83.57	+0.90	83.87	+0.83	85.33	+1.87	85.37	+1.76	84.27	+0.87	84.62	+0.76
	+EVAS Zhang et al. (2025)	83.67	+1.00	83.93	+0.89	85.37	+1.90	85.40	+1.79	84.40	+1.00	84.71	+0.85
	+MemVR Zou et al. (2025)	83.73	+1.07	84.03	+0.99	85.60	+2.13	85.63	+2.02	84.50	+1.10	84.87	+1.02
	+DMLR Liu et al. (2026)	83.67	+1.00	84.02	+0.98	85.80	+2.33	85.85	+2.24	84.60	+1.20	84.93	+1.08
	+LEAD Xu et al. (2026)	83.80	+1.13	84.13	+1.09	86.23	+2.77	86.31	+2.70	84.63	+1.23	84.99	+1.14
	+VALR (Ours)	**85.83**	**+3.17**	**85.93**	**+2.89**	**88.23**	**+4.77**	**88.15**	**+4.54**	**86.77**	**+3.37**	**86.93**	**+3.08**

Table 4: **POPE results.** We report accuracy and F1 (%) under Random, Popular, and Adversarial sampling. Δ denotes percentage-point gains over Vanilla.

Method	R1-OneVision				Vision-R1				ThinkLite-VL			
	Acc↑	Δ	F1↑	Δ	Acc↑	Δ	F1↑	Δ	Acc↑	Δ	F1↑	Δ
Vanilla	61.41	+0.00	58.90	+0.00	64.14	+0.00	61.03	+0.00	62.78	+0.00	60.84	+0.00
+VCD Leng et al. (2024)	62.04	+0.63	59.75	+0.85	65.40	+1.26	62.05	+1.02	63.41	+0.63	61.59	+0.75
+DAMRO Gong et al. (2024)	62.36	+0.95	60.04	+1.14	65.83	+1.68	62.60	+1.57	63.72	+0.95	61.88	+1.04
+HELPD Yuan et al. (2024)	62.46	+1.05	60.02	+1.12	65.93	+1.79	62.84	+1.82	63.83	+1.05	61.95	+1.11
+EVAS Zhang et al. (2025)	62.36	+0.95	59.87	+0.96	66.14	+2.00	62.90	+1.87	63.83	+1.05	62.03	+1.19
+MemVR Zou et al. (2025)	62.46	+1.05	60.20	+1.30	66.35	+2.21	63.22	+2.19	63.93	+1.16	62.27	+1.43
+DMLR Liu et al. (2026)	62.46	+1.05	60.29	+1.39	66.77	+2.63	63.34	+2.31	64.04	+1.26	62.08	+1.24
+LEAD Xu et al. (2026)	62.57	+1.16	60.09	+1.19	66.67	+2.52	63.10	+2.07	63.93	+1.16	62.18	+1.34
+VALR (Ours)	**64.77**	**+3.36**	**62.57**	**+3.67**	**68.98**	**+4.84**	**65.50**	**+4.47**	**66.25**	**+3.47**	**64.37**	**+3.53**

Table 5: **HallusionBench results.** We compare accuracy and F1 (%) across three backbones, with Δ denoting percentage-point gains over Vanilla.

5.2 Main Results

Hallucination Mitigation. VALR achieves the highest accuracy and F1 across the three backbones on POPE and HallusionBench (Tables 4 and 5). The gains hold across Random, Popular, and Adversarial sampling on POPE, including negative questions about absent objects that commonly co-occur with visible ones. On POPE-Adversarial, Vision-R1 reaches 88.23% accuracy, 2.00 percentage points above LEAD. The gains extend beyond object-existence questions to HallusionBench, where accuracy improves by 3.36–4.84 percentage points over Vanilla. On POPE-Adversarial and HallusionBench, paired gains over the strongest baselines have 95% confidence intervals above zero, and VALR's three-seed accuracy standard deviations are 0.11–0.21 percentage points (Appendix Tables 14 and 15).

Broader Reasoning Performance. MMStar and MathVision extend the evaluation to broader multimodal understanding and mathematical reasoning. VALR achieves higher accuracy than Vanilla and the compared baselines on both benchmarks across all three backbones (Table 6). On MathVision, accuracy gains over Vanilla range from 4.18 to 4.38 percentage points across backbones. Hallucination mitigation is therefore accompanied by improved performance on these broader reasoning tasks, where accurate perception must support subsequent inference from the image.

5.3 Ablation Studies

Module Ablation. Removing either visual initialization or paired scoring lowers accuracy and F1 on both benchmarks (Table 7(a), with Vanilla as a reference). The respective variants replace question-guided initialization with placeholder embeddings or the paired reward with absolute-confidence scoring, retaining candidate-dependent retrieval, fusion, and iterative exploration. The

Method	R1-OneVision				Vision-R1				ThinkLite-VL			
	MMStar		MathVision		MMStar		MathVision		MMStar		MathVision	
	Acc↑	Δ	Acc↑	Δ	Acc↑	Δ	Acc↑	Δ	Acc↑	Δ	Acc↑	Δ
Vanilla	50.73	+0.00	28.45	+0.00	54.60	+0.00	34.28	+0.00	52.40	+0.00	31.41	+0.00
+VCD Leng et al. (2024)	51.27	+0.53	28.91	+0.46	55.40	+0.80	34.93	+0.66	52.93	+0.53	31.68	+0.26
+DAMRO Gong et al. (2024)	51.67	+0.93	29.28	+0.82	55.87	+1.27	35.56	+1.28	53.27	+0.87	31.91	+0.49
+HELPD Yuan et al. (2024)	51.80	+1.07	29.05	+0.59	55.67	+1.07	35.03	+0.76	53.00	+0.60	31.88	+0.46
+EVAS Zhang et al. (2025)	51.60	+0.87	29.31	+0.86	56.13	+1.53	35.49	+1.22	53.20	+0.80	32.14	+0.72
+MemVR Zou et al. (2025)	52.00	+1.27	29.64	+1.18	56.33	+1.73	35.95	+1.68	53.47	+1.07	32.43	+1.02
+DMLR Liu et al. (2026)	51.93	+1.20	29.93	+1.48	56.47	+1.87	35.63	+1.35	53.27	+0.87	32.76	+1.35
+LEAD Xu et al. (2026)	52.13	+1.40	29.77	+1.32	56.27	+1.67	35.66	+1.38	53.33	+0.93	32.27	+0.86
+VALR (Ours)	**54.67**	**+3.93**	**32.70**	**+4.24**	**58.87**	**+4.27**	**38.65**	**+4.38**	**55.93**	**+3.53**	**35.59**	**+4.18**

Table 6: **Results on MMStar and MathVision.** We compare accuracy (%) across three backbones, with Δ denoting percentage-point gains over Vanilla.

Table 7: **Ablation studies of VALR.** Component ablations and matched controls use R1-OneVision, while hyperparameter sensitivity is evaluated across all three backbones.

(a) Component ablation

Visual init.	Paired reward	POPE-A		HallusionBench	
		Acc.↑	F1↑	Acc.↑	F1↑
–	–	82.67	83.04	61.41	58.90
✓	–	84.03	84.30	62.78	60.49
–	✓	84.30	84.78	63.09	60.87
✓	✓	**85.83**	**85.93**	**64.77**	**62.57**

(b) Matched controls

Variant	POPE-A		HallusionBench	
	Acc.↑	F1↑	Acc.↑	F1↑
Vanilla	82.67	83.04	61.41	58.90
Random search	82.93	83.36	61.72	59.19
Random visual tokens	83.20	83.56	62.15	59.64
Full VALR	**85.83**	**85.93**	**64.77**	**62.57**

(c) Hyperparameter sensitivity

Model	K					R					λ				
	Val.	POPE-A		HallusionBench		Val.	POPE-A		HallusionBench		Val.	POPE-A		HallusionBench	
		Acc↑	F1↑	Acc↑	F1↑		Acc↑	F1↑	Acc↑	F1↑		Acc↑	F1↑	Acc↑	F1↑
R1-OneVision	6	85.23	85.38	64.25	62.05	12	85.13	85.28	64.04	61.83	0.65	85.20	85.35	64.14	61.98
	8	**85.83**	**85.93**	**64.77**	**62.57**	16	**85.83**	**85.93**	**64.77**	**62.57**	0.70	**85.83**	**85.93**	**64.77**	**62.57**
	10	85.47	85.63	64.46	62.28	20	85.40	85.53	64.35	62.21	0.78	85.30	85.46	64.25	62.05
Vision-R1	6	87.40	87.39	68.14	64.81	12	87.30	87.25	67.93	64.66	0.67	87.37	87.34	68.03	64.73
	8	**88.23**	**88.15**	**68.98**	**65.50**	16	**88.23**	**88.15**	**68.98**	**65.50**	0.73	**88.23**	**88.15**	**68.98**	**65.50**
	10	87.87	87.82	68.45	65.12	20	87.77	87.73	68.35	64.96	0.79	87.70	87.65	68.24	64.88
ThinkLite-VL	6	85.63	85.85	64.88	63.05	12	85.53	85.74	64.77	62.90	0.66	85.60	85.81	64.77	62.98
	8	**86.77**	**86.93**	**66.25**	**64.37**	16	**86.77**	**86.93**	**66.25**	**64.37**	0.75	**86.77**	**86.93**	**66.25**	**64.37**
	10	86.17	86.37	65.40	63.57	20	86.03	86.27	65.19	63.43	0.82	85.97	86.18	65.19	63.34

two comparisons support incorporating question-relevant evidence before sampling and evaluating the confidence change induced by newly retrieved content. In fixed diagnostic pools, paired reward improves selection accuracy over absolute confidence, with statistical support on HallusionBench (Appendix Table 16).

Control Ablation. Random search fixes the sampling center at the visual initialization while retaining candidate conditioning, paired scoring, and best-state selection. Random visual tokens instead retain the full search procedure but replace aligned retrieval with same-count random sets at each iteration. Both controls preserve the visual initialization and use 15 candidate pairs. Full VALR outperforms both controls in accuracy and F1, supporting adaptive exploration with aligned retrieval (Table 7(b)). Appendix D.2 specifies the matching of retrieval, scoring, and exploration settings.

Hyperparameter Ablation. We vary the number of latent positions K, retrieved visual tokens R, and fusion weight λ across three backbones to examine latent capacity, retrieval breadth, and conditioning strength. All tested settings improve accuracy and F1 over Vanilla, with the defaults performing best and larger K or R bringing no further gains within the tested ranges (Table 7(c)). Appendix Tables 17 and 18 examine the update and proposal scales and compare soft with hard aggregation, supporting the defaults while showing that aggregation preferences vary across tasks.

5.4 Analysis and Discussion

Visual Evidence Dependence. We test whether VALR's predictions depend on the retrieved content using R1-OneVision on the 951 image-based HallusionBench questions. Masking selected tokens and rebuilding initialization and exploration lowers accuracy by 2.42 percentage points versus 0.95 for same-count random masking and more than doubles the answer-flip rate (Table 8(a)). The larger log-probability drop likewise indicates greater prediction sensitivity to the selected tokens. At the image level, the clean–counterfactual visual margin M_{vis} compares the length-normalized log-probability of

Table 8: **Visual evidence dependence and the performance–cost trade-off of VALR.**

(a) Visual-token deletion

Masked tokens	HallusionBench		Answer change	
	Acc.	Δ	Flip rate	Log-p drop
None	64.77	—	—	—
Random	63.83	−0.95	8.7%	0.33
VALR-selected	62.36	−2.42	**18.2%**	**0.82**

(b) Clean–counterfactual visual margin

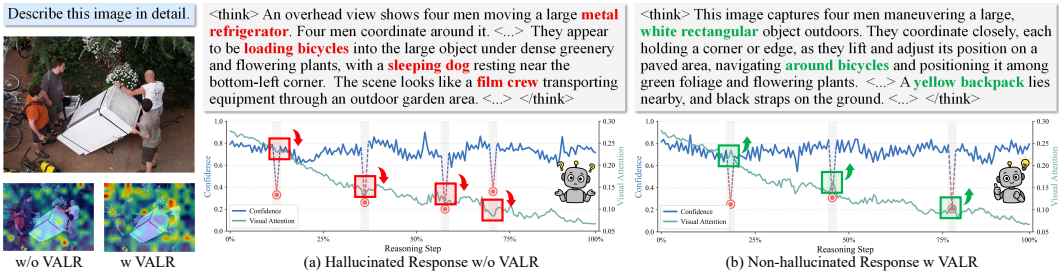
Group	N	Vanilla	VALR	Δ
All	951	0.41	**0.58**	+0.17
Vanilla wrong	367	0.27	**0.52**	+0.25
Vanilla correct	584	0.50	**0.62**	+0.12
VALR corrected	96	0.23	**0.55**	+0.32

(c) Accuracy / inference time

Method	R1-OneVision	Vision-R1	ThinkLite-VL
Vanilla	61.41 / 14:54	64.14 / 16:24	62.78 / 14:06
VCD	62.04 / 32:18	65.40 / 35:48	63.41 / 30:36
DAMRO	62.36 / 22:32	65.83 / 23:18	63.72 / 21:48
DMLR	62.46 / 17:32	66.77 / 18:20	64.04 / 17:05
LEAD	62.57 / 18:53	66.67 / 19:42	63.93 / 18:12
VALR	**64.77 / 17:48**	**68.98 / 18:36**	**66.25 / 17:21**

(d) Optimization budget

N_{opt}	POPE-A		HallusionBench		Time	Mem.
	Acc.↑	F1↑	Acc.↑	F1↑		
0	82.67	83.04	61.41	58.90	14:54	base
3	84.13	84.43	63.09	60.52	15:32	2.1G
5	84.93	85.12	63.93	61.59	15:53	2.2G
8	85.60	85.73	64.56	62.35	16:21	2.1G
15	**85.83**	**85.93**	**64.77**	**62.57**	17:48	2.3G



(a) Hallucinated Response w/o VALR (b) Non-hallucinated Response w VALR

Figure 4: **Representative trajectories with and without VALR.** Responses are shown with their confidence and visual-reliance traces.

the same reference answer under original and spatially shuffled images. A larger margin indicates a greater reference-answer log-likelihood advantage for the original image over its shuffled counterpart. VALR increases this margin in all groups, most strongly among corrected examples, from 0.23 to 0.55 (Table 8(b)). Appendix D.4 gives the deletion and scoring protocols.

Qualitative Analysis. Figure 4 illustrates how rising confidence can make visually unsupported details appear reliable (Section 3.1). Without VALR, the response introduces an unsupported *metal refrigerator* as confidence rises despite weak visual reliance. VALR avoids this detail, with renewed confidence accompanied by greater attention to the image.

Inference Cost Analysis. On HallusionBench, VALR achieves the highest accuracy among the timed methods, exceeding DMLR by 2.21–2.31 percentage points with 16 seconds of additional inference time across the three backbones (Table 8(c)). For R1-OneVision, using 8 rather than 15 exploration steps reduces time from 17:48 to 16:21 while retaining about 93% and 94% of the full-budget accuracy gains over Vanilla on POPE-A and HallusionBench, respectively (Table 8(d)). The remaining steps provide smaller gains, making shorter exploration useful when inference time is constrained. Appendix Tables 19 and 20 report reasoning-length and optional early-stopping results.

6 Conclusion

Incorrect multimodal reasoning trajectories can regain confidence while visual reliance remains weak, making renewed certainty an unreliable indication of image support. Using confidence alone to guide exploration can amplify this mismatch and increase high-confidence errors. Controlled interventions show that weakening visual evidence disrupts correct predictions, while restoring its original content reduces the resulting errors. We introduced Visually-Anchored Latent Rectification (VALR), a training-free method that lets image evidence shape candidate states and uses the resulting confidence gain to guide exploration and select the state for generation. Across the evaluated models and benchmarks, this evidence-guided exploration reduces hallucinations relative to the compared inference-time baselines and improves performance on broader multimodal understanding and mathematical reasoning tasks.

AI USE STATEMENT

Generative AI tools assisted with literature review, experimental-design discussions, code checks, analysis of reported results, translation, manuscript editing, and figure and table preparation. Research decisions and interpretation remained under author direction. The authors take responsibility for the final manuscript and all AI-assisted content, including citations, code, experimental results, and scientific claims.

ETHICS STATEMENT

VALR is intended to improve the reliability of multimodal reasoning by reducing visually unsupported responses. Our evaluation uses existing multimodal benchmarks. Although VALR reduces hallucinations in the evaluated settings, the underlying models can still produce confident but unsupported answers and retain biases inherited from pretraining data. Applications involving consequential decisions require task-specific validation and appropriate human oversight.

REPRODUCIBILITY STATEMENT

Section 4 specifies visual initialization, candidate conditioning, and paired-reward updates. Appendix C details token allocation, matched-branch evaluation, and the complete algorithm, while Section 5.1 describes backbones, hyperparameters, and evaluation metrics. Appendices A and B document dataset prompts, trajectory measurements, answer-confidence scoring, and controlled visual interventions. Appendix D additionally reports visual-dependence protocols, statistical analyses, and inference-cost details to support reproduction of the reported comparisons.

REFERENCES

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yiheng Xu, Jiabo Ye, Xi Zhang, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? In *NeurIPS*, volume 37, pp. 27056–27087, 2024.

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.

Jun Gao, Yongqi Li, Ziqiang Cao, and Wenjie Li. Interleaved-modal chain-of-thought. In *CVPR*, 2025.

Xuan Gong, Tianshi Ming, Xinpeng Wang, and Zhihua Wei. Damro: Dive into the attention mechanism of lvlm to reduce object hallucination. In *EMNLP*, pp. 7696–7712, 2024.

Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. Hallusionbench: An advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *CVPR*, pp. 14375–14385, 2024.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. In *COLM*, 2025.

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. In *ICLR*, 2026.

Kimi Team, Angang Du, Bohong Yin, Bowei Xing, Bowen Qu, Bowen Wang, Cheng Chen, Chenlin Zhang, Chenzhuang Du, Chu Wei, et al. Kimi-vl technical report. *arXiv preprint arXiv:2504.07491*, 2025.

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 *CVPR*, pp. 13872–13882, 2024.

Hengli Li, Chenxi Li, Tong Wu, Xuekai Zhu, Yuxuan Wang, Zhaoxin Yu, Eric Hanchen Jiang, Song-Chun Zhu, Zixia Jia, Ying Nian Wu, and Zilong Zheng. Seek in the dark: Reasoning via test-time instance-level policy gradient in latent space. *arXiv preprint arXiv:2505.13308*, 2025.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *EMNLP*, 2023.

Chengzhi Liu, Yuzhe Yang, Yue Fan, Qingyue Wei, Sheng Liu, and Xin Eric Wang. Reasoning within the mind: Dynamic multimodal interleaving in latent space. In *CVPR findings*, 2026.

Sheng Liu, Tianlang Chen, Pan Lu, Haotian Ye, Yizheng Chen, Lei Xing, and James Zou. Fractional reasoning via latent steering vectors improves inference time compute. *arXiv preprint arXiv:2506.15882*, 2025a.

Sheng Liu, Haotian Ye, and James Zou. Reducing hallucinations in large vision-language models via latent space steering. In *ICLR*, 2025b.

Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *NeurIPS*, pp. 2507–2521, 2022.

Hongyu Qu, Guangming Yao, Ling Xing, Xiaobin Hu, Rongxing Ding, Guibin Zhang, Fan Zhang, Yi Yuan, Xiangbo Shu, and Shuicheng Yan. Beyond retrieval: Progressive latent memory evolution for streaming video understanding. *arXiv preprint arXiv:2609.04131*, 2026.

Qwen Team. Qwen3-vl: Sharper vision, deeper thought, broader action, 2025.

Dachuan Shi, Abedelkadir Asi, Keying Li, Xiangchi Yuan, Leyan Pan, Wenke Lee, and Wen Xiao. SwiReasoning: Switch-thinking in latent and explicit for pareto-superior reasoning LLMs. In *ICLR*, pp. 137060–137093, 2026.

Xinyu Tian, Shu Zou, Zhaoyuan Yang, Mengqi He, Fabian Waschkowski, Lukas Wesemann, Peter Tu, and Jing Zhang. More thought, less accuracy? on the dual nature of reasoning in vision-language models. In *ICLR*, 2026.

Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *CVPR*, pp. 9568–9578, 2024.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. Measuring multimodal mathematical reasoning with math-vision dataset. In *NeurIPS*, volume 37, pp. 95095–95169, 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. In *NeurIPS*, volume 38, pp. 131702–131734, 2025.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992.

Yige Xu, Xu Guo, Zhiwei Zeng, and Chunyan Miao. Softcot: Soft chain-of-thought for efficient reasoning with llms. In *ACL*, pp. 23336–23351, 2025a.

Yige Xu, Xu Guo, Zhiwei Zeng, and Chunyan Miao. Softcot++: Test-time scaling with soft chain-of-thought reasoning. *arXiv preprint arXiv:2505.11484*, 2025b.

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. In *NeurIPS*, volume 38, 2025c.

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 *CVPR*, 2026.

Yi Yang, Xiaoxuan He, Hongkun Pan, Xiyan Jiang, Yan Deng, Xingtao Yang, Haoyu Lu, Dacheng Yin, Fengyun Rao, Minfeng Zhu, Bo Zhang, and Wei Chen. R1-Onevision: Advancing generalized multimodal reasoning through cross-modal formalization. In *ICCV*, pp. 2376–2385, 2025.

Wengao Ye, Yan Liang, and Lianlei Shan. Thinking on the fly: Test-time reasoning enhancement via latent thought policy optimization. In *ICLR*, 2026.

Fan Yuan, Chi Qin, Xiaogang Xu, and Piji Li. Helpd: Mitigating hallucination of lvlms by hierarchical feedback learning with vision-enhanced penalty decoding. In *EMNLP*, pp. 1768–1785, 2024.

Xiaofeng Zhang, Yihao Quan, Chen Shen, Chaochen Gu, Xiaosong Yuan, Shaotian Yan, Jiawei Cao, Hao Cheng, Kaijie Wu, and Jieping Ye. Shallow focus, deep fixes: Enhancing shallow layers vision attention sinks to alleviate hallucination in lvlms. In *EMNLP*, pp. 3512–3534, 2025.

Xin Zhang, Qiqi Tao, Jiawei Du, Moyun Liu, and Joey Tianyi Zhou. Visual latents know more than they say: Unsilencing latent reasoning in MLLMs. *arXiv preprint arXiv:2605.02735*, 2026.

Xin Zou, Yizhou Wang, Yibo Yan, Yuanhuiyi Lyu, Kening Zheng, Sirui Huang, Junkai Chen, Peijie Jiang, Jia Liu, Chang Tang, et al. Look twice before you answer: Memory-space visual retracing for hallucination mitigation in multimodal large language models. In *ICML*, 2025.

APPENDIX OVERVIEW

This supplementary material provides prompt templates, diagnostic protocols, the complete VALR procedure, comparisons with related latent-space methods, and additional experimental results. The contents are organized as follows:

- **Prompt Templates for Datasets (§A):** Prompt formats used for POPE, HallusionBench, MMStar, and MathVision.
- **Confidence and Visual-Evidence Analyses (§B):** Trajectory aggregation, supplementary confidence comparisons, and intervention protocols for Section 3.
- **Complete VALR Procedure (§C):** Model and evaluation settings, parameter selection, visual initialization, matched-branch scoring, update derivation, pseudocode, and method comparisons (§C.6).
- **More Experimental Results and Analysis (§D):** Statistical robustness, complete ablations, behavioral controls, cost details, and supplementary visualizations.
- **Qualitative Case Studies (§E):** Image-description comparisons examining invented objects, scene interpretations, and overly specific claims.

A PROMPT TEMPLATES FOR DATASETS

The boxes illustrate dataset-specific question and answer formats. We serialize inputs with each checkpoint's native chat template, with USER and ASSISTANT denoting conversational roles. Appendix C.1 specifies preprocessing, decoding, and answer evaluation.

A.1 PROMPT TEMPLATES FOR POPE DATASET

The POPE benchmark Li et al. (2023) evaluates object-level hallucination through image-grounded binary judgments. In our setting, each POPE sample is formatted as a visual question together with two candidate answers corresponding to *yes* and *no*, so that all models follow a consistent answer space during evaluation. The input format is shown below, where the prompt is highlighted in **green** and the image placeholder is highlighted in **red**.

Model-specific chat template.

USER: **IMAGE**

Is there a snowboard in the image?
Options: A. yes B. no

ASSISTANT:

A.2 PROMPT TEMPLATES FOR HALLUSIONBENCH DATASET

HallusionBench Guan et al. (2024) evaluates whether the model can answer visually grounded questions without being misled by image-grounded inconsistencies or visual illusions. In our experiments, we preserve the original question for each sample and directly query the model with the associated image. The original Yes/No question is retained, and evaluation extracts the final binary answer. An illustrative template is shown below:

Model-specific chat template.

USER: IMAGE

Please answer the question according to the image.
Question: Is this the official KFC logo?

ASSISTANT:

A.3 Prompt Templates for MMStar Dataset

MMStar Chen et al. (2024) is a multimodal reasoning benchmark composed primarily of multiple-choice questions. For this benchmark, we concatenate the original question with its candidate options and ask the model to reason over the image before choosing the final answer. The prompt template is shown below:

Model-specific chat template.

USER: IMAGE

Which option describes the object relationship in the image correctly?
Options: A. The suitcase is on the book.
B. The suitcase is beneath the cat.
C. The suitcase is beneath the bed.
D. The suitcase is beneath the book.

Please think step by step and then provide your answer.

ASSISTANT:

A.4 Prompt Templates for MathVision Dataset

MathVision Wang et al. (2024) evaluates science-oriented multimodal mathematical reasoning. In our experiments, we preserve the original problem statement and append a reasoning-oriented instruction so that the model produces explicit long-chain reasoning before giving the final answer. An illustrative template is shown below:

Model-specific chat template.

USER: IMAGE

Solve the problem and output the answer in the format of \boxed{your answer}.
Question: Bettina chooses five points A, B, C, D and E on a circle and draws the tangent to the circle at point A. She realizes that the five angles marked x are all equally big. How big is the angle $\angle ABD$?
Options: A. $66°$**, B.** $70.5°$**, C.** $72°$**, D.** $75°$**, E.** $77.5°$

Please think step by step and provide your detailed solution.

ASSISTANT:

B Details for Confidence and Visual-Evidence Analyses

The three analyses in Section 3 use different populations and measurements. Trajectory recovery is evaluated within a selected declining cohort, high-confidence errors are measured across a fixed

question set, and induced hallucinations are measured on originally correct samples. This section details these measurements and protocols and presents the supplementary comparisons.

B.1 Trajectory Measurements and Recovery Aggregation

Trajectory confidence in Equation 1 uses full-vocabulary next-token entropy:

$$\mathcal{H}(P_t) = -\sum_{u \in \mathcal{Y}} P_t(u) \log P_t(u).$$

The same logarithm base is used in the entropy and its normalization. Trajectory analysis uses this full-vocabulary statistic, whereas latent exploration uses the paired entropy reward defined in Section 4.3.

For a decoder with L layers, we use zero-based layers $\lceil 0.75L \rceil, \ldots, L-1$, with L read from the checkpoint configuration, and include all query heads in those layers. Attention weights are collected after softmax in inference mode. For each layer and head, we divide the attention mass on actual image patch/token keys by the mass on all valid, causally visible keys. Image boundary markers are excluded from the numerator, while padding and causally masked positions are excluded from the denominator. The per-layer, per-head ratios are then averaged with equal weights to obtain VAR_t. This statistic measures the relative share of attention allocated to image tokens at each reasoning step.

The reasoning span is the body between `<think>` and `</think>` when these markers are present. Otherwise, it runs from the beginning of the assistant body to the fixed final-answer delimiter used by the evaluation parser. We exclude prompt tokens, special markers, the final answer, and EOS. Reliable reasoning-span parsing is required, with failures recorded separately. We define the relative windows using reasoning-body token counts. At each query position, attention and next-token logits come from the same forward pass, pairing visual reliance with the next-token distribution predicted at that position.

For Table 1, we first identify Vanilla R1-OneVision trajectories on HallusionBench whose mean confidence in the middle window (40–60% of the reasoning span) is below that in the early window (0–20%). We randomly select 500 trajectories from this declining population and group them by final-answer correctness. Confidence recovery requires the late-window mean (80–100%) to exceed the middle-window mean. This yields 201 recovered trajectories among 293 Correct trajectories and 149 among 207 Incorrect trajectories. The recovery percentages are therefore $201/293$ and $149/207$, retaining non-recovered trajectories in the denominators.

The last three rows of Table 1 concern only the recovered trajectories. For each trajectory, confidence change and visual-reliance change are late-minus-middle differences between window means, followed by averaging within the outcome group. The final row is the fraction of recovered trajectories with non-positive visual-reliance change. These comparisons characterize visual-attention changes conditional on confidence recovery within the selected confidence-declining cohort.

Figure 2 shows the images and confidence and visual-attention trajectories for two image-description cases. These cases illustrate local changes around confidence valleys and are separate from the binary-question cohort used for the recovery table.

Table 9 extends this analysis to R1-OneVision, Vision-R1, and ThinkLite-VL on HallusionBench and MathVision, using the same trajectory measures, windows, and recovery criteria. Each model–dataset pair includes 500 selected confidence-declining Vanilla trajectories, grouped by final-answer correctness. The Count columns give the size of each correctness group, which serves as the denominator for its recovery rate. Both Δ measures are averaged over recovered trajectories within the corresponding group, using per-trajectory late-minus-middle differences. Across all six pairs, recovered incorrect trajectories show a mean decrease in visual reliance, whereas recovered correct trajectories show a mean increase.

B.2 Final-Answer Confidence and Supplementary Comparisons

The negative control in Section 3.2 compares Vanilla, confidence-only exploration, and full VALR on the same 951 image-based HallusionBench questions in `hallusion.json`. The "+ Vis. cond." column in Table 2 corresponds to the full Visually-Anchored Latent Rectification (VALR) configuration described in Section 4. The analysis includes all 951 questions across confidence trajectories

Table 9: **Confidence recovery across models and tasks.** Each row summarizes 500 selected confidence-declining Vanilla trajectories, grouped by final-answer correctness. Changes are averaged over recovered trajectories.

Dataset	Backbone	Count		Recovery		△ confidence		△ visual reliance	
		Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect
HallusionBench	R1-OneVision	293	207	68.6%	72.0%	+0.039	+0.047	+0.010	-0.003
HallusionBench	Vision-R1	320	180	67.5%	74.4%	+0.036	+0.051	+0.008	-0.005
HallusionBench	ThinkLite-VL	306	194	69.0%	71.6%	+0.041	+0.044	+0.006	-0.002
MathVision	R1-OneVision	142	358	63.4%	70.4%	+0.031	+0.043	+0.007	-0.004
MathVision	Vision-R1	171	329	66.1%	73.6%	+0.034	+0.049	+0.009	-0.006
MathVision	ThinkLite-VL	157	343	64.3%	71.1%	+0.029	+0.046	+0.005	-0.003

and final-answer outcomes. Confidence-only exploration retains the image input, initializes the K latent positions from $E(\tau_{\text{lat}})$, and omits candidate visual fusion. It uses Gaussian proposals, a fixed 15-candidate budget, and the absolute reward $1 - K^{-1} \sum_i \mathrm{H}(q_i)$, where each q_i is renormalized on that branch's own top-16 predictions. The reward guides center updates and best-candidate selection under the same final decoding settings. Full VALR instead uses visual initialization and paired evaluation as specified in Section 4.

Equation 2 scores the model's actual predicted Yes/No answer in that method's complete context/state immediately before answering. For a candidate string $a = (a_1, \ldots, a_L)$, its probability is $P(a \mid x_i) = \prod_{\ell=1}^{L} P(a_\ell \mid x_i, a_{<\ell})$, where L is the number of tokens in that candidate. The candidate probabilities are normalized over Yes and No, and the shared high-confidence threshold is $c_i \geq 0.90$. We report the percentage of all 951 questions whose final predictions are both incorrect and high-confidence.

Prediction transitions. Table 10 compares prediction transitions and high-confidence errors across the exploration and visual-conditioning variants. C2W denotes a change from a correct Vanilla answer to a wrong answer, whereas W2C denotes the reverse. Both rates use all 951 evaluation questions as the denominator, so W2C minus C2W gives the accuracy gain over Vanilla in percentage points. Confidence-only exploration corrects more answers than it disrupts, yet increases high-confidence errors on all three backbones. Full VALR yields more corrections and fewer correct-to-wrong transitions than confidence-only exploration, improving accuracy while reducing high-confidence errors.

Table 10: Supplementary HallusionBench transition and high-confidence error results (%). C2W denotes correct-to-wrong, and W2C denotes wrong-to-correct. A dash denotes no transition comparison for Vanilla.

Backbone	Variant	C2W	W2C	Hi-conf wrong
R1-OneVision	Vanilla	—	—	18.5
	Confidence-only	8.20	9.88	22.3
	Visual conditioning only	6.94	8.31	16.8
	Full VALR	6.73	10.09	14.2
Vision-R1	Vanilla	—	—	16.8
	Confidence-only	4.31	6.52	20.7
	Visual conditioning only	3.15	4.94	15.4
	Full VALR	2.84	7.68	12.4
ThinkLite-VL	Vanilla	—	—	18.3
	Confidence-only	5.15	6.62	22.8
	Visual conditioning only	4.10	5.26	16.9
	Full VALR	3.36	6.83	14.6

Object-existence responses. The POPE-Adversarial comparison separates the frequency of Yes responses from false positives (FP) and high-confidence false positives (Table 11). Yes-bias is the fraction of all questions answered Yes, while FP and high-confidence FP use ground-truth No

questions as their denominator. Relative to Vanilla, confidence-only exploration lowers the overall FP rate but increases high-confidence FP on all three backbones. Full VALR lowers both rates, showing that fewer object-existence errors can be accompanied by fewer high-confidence errors when exploration incorporates visual evidence.

Table 11: Supplementary POPE-Adversarial results (%). FP denotes false positives.

Backbone	Variant	Yes-bias	FP	Hi-conf FP
R1-OneVision	Vanilla	52.30	19.67	12.80
	Confidence-only	53.10	18.80	14.33
	Visual conditioning only	51.80	17.80	10.93
	Full VALR	50.70	14.87	8.67
Vision-R1	Vanilla	50.90	17.47	11.60
	Confidence-only	51.80	15.87	13.13
	Visual conditioning only	50.40	14.80	10.20
	Full VALR	49.30	11.07	7.40
ThinkLite-VL	Vanilla	52.70	19.27	13.00
	Confidence-only	53.40	18.47	14.80
	Visual conditioning only	52.00	17.33	11.33
	Full VALR	51.20	14.40	9.20

B.3 Controlled Visual Interventions

Population and intervention order. The R1-OneVision experiment uses the 584 samples that Vanilla answers correctly and that have valid reasoning spans within the 951 image-based Hallusion-Bench questions. Real and shuffled restoration both follow selected-step visual suppression on this same cohort, including samples that remain correct after suppression. For intervention condition b, the induced hallucination rate is

$$\mathrm{IHR}(b) = \frac{N_{\mathrm{corr}\to\mathrm{hall}}(b)}{N_{\mathrm{corr}}}, \tag{13}$$

where $N_{\mathrm{corr}} = 584$ and the numerator counts members of this cohort whose final predictions become hallucinated under b. Table 3 reports this fraction in percent. The induced-error counts are 132 for selected-step visual suppression, 22 for random-step visual suppression, 12 for non-visual suppression, and 27 for norm-matched noise. After suppression, shuffled and real restoration leave 43 and 20 errors, respectively, corresponding to 7.4% and 3.4% of the cohort after rounding. Both restoration rates use the complete 584-sample cohort as their denominator.

Intervention timing and shared context. Interventions use the same final-quarter decoder layers and all query heads as the trajectory analysis in Appendix B.1. We restrict candidate steps to the middle 20–80% of the original reasoning span and select the earliest step t^* attaining the minimum score

$$S_t = \mathrm{Conf}_t \, \mathrm{VAR}_t. \tag{14}$$

All selected-step conditions branch from the same generated prefix and KV-cache state immediately before the query computation at t^*. The branch cache contains no results from an unmodified computation of that query. Each intervention affects only this single decoding computation, after which the hook is removed and generation proceeds freely from the intervened state. The original image, prompt, decoding configuration, and random seed are shared across conditions.

Suppression and matched controls. For an intervened attention row, let A_j denote its original post-softmax weight on key j. Visual suppression scales the visual weights by $\alpha = 0.1$ and renormalizes over valid keys:

$$A'_j = \frac{w_j A_j}{\sum_k w_k A_k}, \qquad w_j = \begin{cases} \alpha, & j \text{ is a visual key}, \\ 1, & \text{otherwise}. \end{cases} \tag{15}$$

The resulting attention output is rescaled to the norm of the unmodified output at the same position. For the random-position control, we randomly select a different step within the same 20–80% range

and apply the identical operator and strength, branching from the original prefix immediately before that step. Non-visual suppression matches the attention mass being modified, while the noise control matches the norm of the output perturbation caused by visual suppression.

Real and shuffled restoration. Both restoration branches begin with the same selected-step visual suppression. Real restoration recomputes a visual-only context from the current query and the original visual keys and values. We match this context's norm to the suppressed output, mix the two with restoration weight $\beta = 0.5$, and rescale the mixed output to the original output norm. This restoration weight is separate from the method's fusion weight λ. Shuffled restoration keeps visual keys, positions, and attention weights fixed and permutes only the assignment of visual values to keys. Its restoration vector undergoes the same norm matching and mixing, so the comparison changes the restored content while controlling its magnitude. Under the tested suppression-and-restoration protocol, predictions depend on which visual content is restored.

Paired restoration statistics. Table 12 compares original and shuffled restoration on the same 584 originally correct questions. Of these, 531 remain correct and 10 become incorrect under both conditions. Original restoration is correct while shuffled restoration is incorrect on 33 questions, with the reverse occurring on 10. Define d_i as the shuffled-restoration error indicator minus the original-restoration error indicator on question i. The IHR reduction is $100\bar{d}$, with its two-sided 95% interval computed using the question-level paired normal approximation in Equation 24 and $n = 584$. We also apply a two-sided exact McNemar test to the 33 versus 10 discordant outcomes.

Table 12: **Paired comparison of visual restoration.** Both conditions follow the same selected-step suppression. The IHR reduction and its 95% CI are in percentage points.

Comparison	N	**IHR reduction**	**95% CI**	**McNemar** p
Original vs. shuffled	584	3.94	[1.76, 6.12]	0.00061

B.4 Supplementary Position and Attention Analyses

Position-dependent correction. To examine how the effect of an intervention varies along a trajectory, we measure the hallucination correction rate at normalized reasoning position $p \in [0, 1]$:

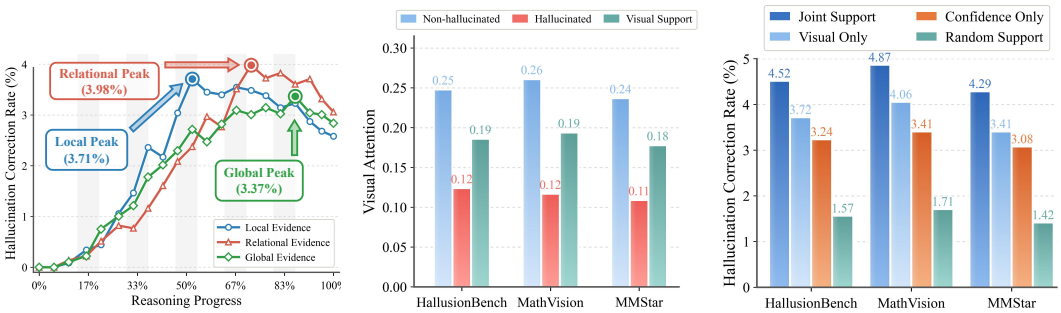
$$\mathrm{HCR@}p = \frac{N_{\mathrm{corr}}(p)}{N_{\mathrm{hall}}}, \tag{16}$$

where N_{hall} counts baseline-hallucinated samples and $N_{\mathrm{corr}}(p)$ counts those corrected by intervention at p. Each reasoning chain is normalized to relative positions, and the same intervention policy is evaluated at each position bin. Correction sensitivity peaks at specific reasoning positions under the evaluated intervention policy (Figure 5(a)). This baseline-hallucinated population is separate from the baseline-correct intervention cohort in Section 3.3.

Attention allocation and intervention ranking. Figure 5(b) compares group-average visual-token attention. "Joint Support" in Figure 5(c) denotes the combined score $S_t = \mathrm{Conf}_t \, \mathrm{VAR}_t$ from Equation 14, with lower scores used to select intervention positions. A high score requires both high confidence and high visual attention. The combined-score selection policy yields a 4.87% correction rate on MathVision.

C Complete VALR Procedure

VALR distinguishes the search center $\mathcal{H}^{(t)}$, the sampled proposal $\widetilde{\mathcal{H}}^{(t)}$, and its real/control states $\overline{\mathcal{H}}^{(t),\pm}$. The paired score updates the center, while $\mathcal{H}^*$ stores the real branch of the highest-scoring evaluated proposal. The original image remains in context throughout initialization, exploration, and generation.



(a) Correction by position (b) Visual-token attention (c) Intervention ranking

Figure 5: Supplementary position, attention, and intervention-ranking analyses. These descriptive comparisons complement the controlled intervention results in Table 3.

C.1 Model and Evaluation Configuration

Checkpoints and token identities. We use the following 7B checkpoints, loading the tokenizer and processor associated with each model. All three use the Qwen2.5-VL architecture, with 28 decoder layers and hidden dimension 3584.

Backbone	Checkpoint identifier
R1-OneVision	`Fancy-MLLM/R1-Onevision-7B-RL`
Vision-R1	`Osilly/Vision-R1-7B`
ThinkLite-VL	`russwang/ThinkLite-VL-7B`

For all three tokenizers, τ_{lat} is <|endoftext|> (ID 151643) and τ_{vis} is <|image_pad|> (ID 151655). We use the corresponding entries directly from each model's frozen input embedding table. The latent positions remain visible in the attention mask even though the placeholder token also serves as a padding token. The visual anchor functions solely as an embedding-space anchor and bypasses image-placeholder expansion by the processor. The final-quarter attention range used for initialization and diagnostics is layers 21–27 under zero-based indexing, with all query heads included.

Input processing and inference. The native processor resizes images while preserving their aspect ratio, and preprocessing uses the uncropped result. The minimum and maximum pixel budgets are $256 \times 28^2 = 200704$ and $1280 \times 28^2 = 1003520$, respectively. The models remain in evaluation mode with dropout disabled.

Final answers use greedy decoding (`do_sample=False`, `num_beams=1`), a repetition penalty of 1.0, and KV caching. A single 4096-token generation budget covers reasoning and the final answer. Decoding terminates at a checkpoint-defined EOS token or when this budget is exhausted. The EOS IDs are 151645 and 151643 for R1-OneVision, and 151645 for Vision-R1 and ThinkLite-VL. Temperature, top-p, and generation top-k sampling filters are inactive. The reward's $k = 16$ shared-support construction is independent of these decoding settings. Within each backbone, methods share image preprocessing, task prompts, final-decoding settings, and answer evaluation, while retaining their method-specific operations.

Answer extraction and scoring. Evaluation uses the final-answer portion and omits the preceding reasoning. POPE and HallusionBench outputs are normalized to Yes/No, with POPE option labels mapped according to the supplied A/B choices. MMStar outputs are mapped to a single valid option for the corresponding question. MathVision uses the official MATH-V answer-extraction and equivalence-checking routines for boxed answers, choices, and mathematical expressions (Wang et al., 2024). All methods use the same deterministic, task-specific evaluation procedure. Unparseable outputs count as errors in the accuracy denominator and are recorded separately. Binary F1 treats Yes as the positive class.

Hyperparameter selection. We select one fusion weight per backbone on a fixed set of 500 image-based questions from the ScienceQA validation split (Lu et al., 2022). This set is separated from all reported evaluation data by source identifiers and images, keeping related questions together when checking overlap. All other parameters remain at the shared defaults in Section 5.1. The coarse search uses $\lambda \in \{0.50, 0.60, 0.70, 0.80, 0.90\}$, followed by refinement within ± 0.05 of the best coarse value in increments of 0.01. Selection maximizes validation answer accuracy, breaking ties in favor of the smaller λ. All candidate settings use the same validation questions, fixed random seed, and exploration and decoding budgets. The selected weights, 0.70, 0.73, and 0.75, are then fixed across evaluation benchmarks and the three runs for R1-OneVision, Vision-R1, and ThinkLite-VL, respectively. Baselines retain their published method-specific parameter settings. Fusion weights are selected on ScienceQA validation, while Table 7(c) and Table 17 report post-selection sensitivity around these fixed defaults.

C.2 Question-Guided Visual Initialization

We use projected image embeddings in the language model's input space, excluding padding and image boundary markers. The frozen model's embedding table E supplies the latent placeholder $e_{\mathrm{lat}} = E(\tau_{\mathrm{lat}})$ and visual anchor $v_{\mathrm{pad}} = E(\tau_{\mathrm{vis}})$. We adapt question-attention ranking for visual-latent construction Zhang et al. (2026) to a single-pass initialization. One prefill on the original image and prompt provides question-to-image attention. For a decoder with L layers, let $\mathcal{L} = \{\lceil 0.75L \rceil, \ldots, L-1\}$ under zero-based indexing and let H_ℓ be the number of query heads in layer ℓ. Question positions Q include the question and answer choices, but exclude system instructions and special tokens. Post-softmax attention $A_{s,j}^{\ell,h}$ gives the relevance score

$$a_j = \frac{1}{|\mathcal{L}|\,|Q|} \sum_{\ell \in \mathcal{L}} \frac{1}{H_\ell} \sum_{h=1}^{H_\ell} \sum_{s \in Q} A_{s,j}^{\ell,h}. \tag{17}$$

This uses all query heads with equal weights within each layer and averages equally over layers and question positions. Initialization aggregates attention at input-question positions, whereas the trajectory measurement in Appendix B.1 uses generated-reasoning positions.

Sort the image tokens by decreasing a_j, breaking ties by original token index, to obtain the permutation ρ. For $N \geq K$, set $R_0 = \min(R, \lfloor N/K \rfloor)$ and allocate

$$S_i^0 = \{\rho(i + mK) : m = 0, \ldots, R_0 - 1\}, \qquad i = 1, \ldots, K. \tag{18}$$

Interleaving ranks distributes the highest-ranked evidence across positions and makes the initialization sets disjoint. For $N < K$, each position receives one token, cycling through the ranked list. Attention-weighted context is then

$$v_{\mathrm{ctx},i}^{q} = \sum_{j \in S_i^0} \frac{a_j}{\sum_{u \in S_i^0} a_u}\, v_j. \tag{19}$$

If the denominator is zero, we use uniform weights on S_i^0. Equation 3 combines this context with the model-native latent placeholder, using the same anchor and fusion weight as candidate conditioning. Subsequent retrieval uses the entire valid image pool and permits overlapping sets across latent positions.

C.3 Derivation of the Latent Update

Let $G^+(\widetilde{\mathcal{H}}; \mathcal{V})$ and $G^-(\widetilde{\mathcal{H}}; \mathcal{V})$ denote the real and control constructions, with the image and prompt fixed. Their shared-support entropy difference defines $\mathcal{R}(\widetilde{\mathcal{H}})$ in Equation 11. For fixed σ, the center enters the expected reward through the Gaussian proposal density, giving

$$\begin{aligned} \nabla_{\mathcal{H}} \mathcal{J}(\mathcal{H}) &= \int \mathcal{R}(\widetilde{\mathcal{H}})\, \nabla_{\mathcal{H}} \pi(\widetilde{\mathcal{H}} \mid \mathcal{H})\, d\widetilde{\mathcal{H}} \\ &= \mathbb{E}_\pi \Big[\mathcal{R}(\widetilde{\mathcal{H}}) \nabla_{\mathcal{H}} \log \pi(\widetilde{\mathcal{H}} \mid \mathcal{H}) \Big]. \end{aligned} \tag{20}$$

For the isotropic Gaussian over Kd coordinates,

$$\log \pi(\widetilde{\mathcal{H}} \mid \mathcal{H}) = -\frac{Kd}{2}\log(2\pi\sigma^2) - \frac{\|\widetilde{\mathcal{H}} - \mathcal{H}\|_F^2}{2\sigma^2},$$
$$\nabla_{\mathcal{H}} \log \pi(\widetilde{\mathcal{H}} \mid \mathcal{H}) = \frac{\widetilde{\mathcal{H}} - \mathcal{H}}{\sigma^2} = \frac{\epsilon}{\sigma^2}. \tag{21}$$

One proposal yields Equation 12. The estimator differentiates only the Gaussian log density, while retrieval, branch construction, and frozen-backbone evaluation enter through the scalar reward. The resulting score-function direction is the raw perturbation ϵ.

Because both branch entropies depend on the sampled proposal, their difference defines a proposal-dependent optimization objective. The objective contains only this paired entropy difference, with its signed value left unclipped. Each branch entropy lies in $[0, \log|\mathcal{U}_i|]$, and $|\mathcal{U}_i| \le 2k$, giving the bound $|\mathcal{R}| \le \log(2k)$. Positive and negative scores move the center along and against the perturbation, respectively. This single-proposal update can fluctuate, so candidate retention uses the best observed paired score.

C.4 Norm Matching and Matched Branches

Norm matching compares retrieved-content effects at equal final embedding norms for each latent position. The real branch first matches $v_{\text{pad}} + v_{\text{ctx},i}$ to the raw candidate norm and then fuses it with that candidate. The control uses the same candidate, anchor, and λ, omitting only $v_{\text{ctx},i}$ and matching its final norm to the real branch.

The mixture norm depends on both the input norms and the angle between them. Let $a = \|\widetilde{h}_i\|_2 = \|v_{\text{lat},i}\|_2$ and let ϕ be the angle between these inputs. Then

$$\|\overline{h}_i^{+}\|_2^2 = a^2\left[\lambda^2 + (1-\lambda)^2 + 2\lambda(1-\lambda)\cos\phi\right]. \tag{22}$$

The final control normalization in Equation 9 therefore matters even though both branches already use norm-matched anchors. The real mixture itself is left unchanged. For $\mathrm{NM}(x; r)$ with $\|x\|_2 < 10^{-12}$, we use the unit direction of e_{lat} scaled to r.

Both branches retain the original visual input and the visual content already carried by the sampled state. Their entropy difference measures the relative confidence change associated with newly retrieved content. Reward differences across candidates depend on both conditioned and control entropy. We evaluate this relative-confidence signal through the accuracy of the candidates it selects (Table 16).

C.5 Candidate Evaluation and Generation

The Gaussian perturbation scale is proportional to the initial state's root-mean-square magnitude:

$$\sigma = 0.10\frac{\|\mathcal{H}^{(0)}\|_F}{\sqrt{Kd}}. \tag{23}$$

This scale is computed once after initialization and remains fixed throughout exploration.

Three counts govern different parts of the procedure: K latent positions, R retrieved image tokens per position, and k vocabulary entries selected from each branch. The defaults are $K = 8$, $R = 16$, and $k = 16$. The union of the two top-k sets contains at most 32 entries at each position, with ties resolved by vocabulary index. Both branches renormalize probabilities on this same union. Entropy uses natural logs and retains its support-dependent scale. Scoring uses the next-token distributions at the K latent query positions, and answer decoding begins after candidate selection.

Inference mode disables dropout, and both branches use identical positions, masks, and the same multimodal prefix. The cache preceding the latent block is shared. Latent-block computation is rebuilt independently for each candidate and branch, keeping real-branch fusion out of the control keys and values. For each proposal, cosine retrieval takes $O(KNd)$ arithmetic before top-R selection and aggregation. A fixed budget of 15 proposals entails 30 branch evaluations plus the initialization prefill, with prefix reuse where applicable.

Algorithm 1 initializes the best score to $-\infty$ and replaces its stored real state only on a strict improvement. Ties retain the earlier candidate. If every score is negative, the highest-scoring real branch is still retained. Generation uses the stored real state with the highest evaluated score. The optional stopping rule changes the number of evaluated pairs while preserving this selection rule.

Algorithm 1 Visually-Anchored Latent Rectification

Require: Frozen model f_θ, embedding table E, context $(\mathcal{T}, \mathcal{V})$
Require: K, R, $k = 16$, N_{opt}, λ, η, token IDs $\tau_{\text{lat}}, \tau_{\text{vis}}$
Ensure: Generated answer A

$e_{\text{lat}} \leftarrow E(\tau_{\text{lat}})$, $v_{\text{pad}} \leftarrow E(\tau_{\text{vis}})$
Prefill $(\mathcal{T}, \mathcal{V})$ and compute a_j using Equation 17
Construct $\{S_i^0\}$ using Equation 18 and its small-pool rule
for $i = 1, \ldots, K$ **do**
    Compute $v^q_{\text{ctx},i}$ using Equation 19
    $h_i^{(0)} \leftarrow \lambda e_{\text{lat}} + (1-\lambda)\,\text{NM}(v_{\text{pad}} + v^q_{\text{ctx},i}; \|e_{\text{lat}}\|_2)$
end for
$\mathcal{H}^{(0)} \leftarrow [h_i^{(0)}]_{i=1}^K$, $\sigma \leftarrow 0.10\|\mathcal{H}^{(0)}\|_F/\sqrt{Kd}$
$r^* \leftarrow -\infty$
for $t = 0, \ldots, N_{\text{opt}} - 1$ **do**
    Sample $\epsilon^{(t)} \sim \mathcal{N}(0, \sigma^2 I)$
    $\widetilde{\mathcal{H}}^{(t)} \leftarrow \mathcal{H}^{(t)} + \epsilon^{(t)}$
    for $i = 1, \ldots, K$ **do**
        $s_{ij}^{(t)} \leftarrow \cos(\widetilde{h}_i^{(t)}, v_j)$ for every valid image token j
        $\mathcal{N}_i^{(t)} \leftarrow \text{Top}_{\min(R,N)}(s_{ij}^{(t)})$
        $w_{ij}^{(t)} \leftarrow \exp(s_{ij}^{(t)}) / \sum_{\ell \in \mathcal{N}_i^{(t)}} \exp(s_{i\ell}^{(t)})$
        $v^{(t)}_{\text{ctx},i} \leftarrow \sum_{j \in \mathcal{N}_i^{(t)}} w_{ij}^{(t)} v_j$
        $v^{(t)}_{\text{lat},i} \leftarrow \text{NM}(v_{\text{pad}} + v^{(t)}_{\text{ctx},i}; \|\widetilde{h}_i^{(t)}\|_2)$
        $\overline{h}_i^{(t),+} \leftarrow \lambda \widetilde{h}_i^{(t)} + (1-\lambda) v^{(t)}_{\text{lat},i}$
        $b_i^{(t)} \leftarrow \lambda \widetilde{h}_i^{(t)} + (1-\lambda)\,\text{NM}(v_{\text{pad}}; \|\widetilde{h}_i^{(t)}\|_2)$
        $\overline{h}_i^{(t),-} \leftarrow \text{NM}(b_i^{(t)}; \|\overline{h}_i^{(t),+}\|_2)$
    end for
    Compute $\{P_i^{(t),+}\}$ from $f_\theta(\mathcal{T}, \mathcal{V}, \overline{\mathcal{H}}^{(t),+})$
    Independently compute $\{P_i^{(t),-}\}$ from $f_\theta(\mathcal{T}, \mathcal{V}, \overline{\mathcal{H}}^{(t),-})$
    for $i = 1, \ldots, K$ **do**
        $\mathcal{U}_i^{(t)} \leftarrow \text{Top}_k(P_i^{(t),+}) \cup \text{Top}_k(P_i^{(t),-})$
        $q_i^{(t),\pm}(u) \leftarrow P_i^{(t),\pm}(u) / \sum_{v \in \mathcal{U}_i^{(t)}} P_i^{(t),\pm}(v)$
        $E_i^{\pm} \leftarrow -\sum_{u \in \mathcal{U}_i^{(t)}} q_i^{(t),\pm}(u) \log q_i^{(t),\pm}(u)$
    end for
    $r^{(t)} \leftarrow K^{-1} \sum_{i=1}^K (E_i^- - E_i^+)$
    if $r^{(t)} > r^*$ **then**
        $(r^*, \mathcal{H}^*) \leftarrow (r^{(t)}, \overline{\mathcal{H}}^{(t),+})$
    end if
    $\mathcal{H}^{(t+1)} \leftarrow \mathcal{H}^{(t)} + \eta r^{(t)} \epsilon^{(t)} / \sigma^2$
end for
$A \leftarrow \text{Decode}(\mathcal{T}, \mathcal{V}, \mathcal{H}^*)$
return A

C.6 Relation to Latent-Space Reasoning Methods

Latent-space methods differ in how they construct states, evaluate proposals, and incorporate visual information. VALR shares Gaussian proposals and score-function updates with prior test-time optimization methods Ye et al. (2026); Liu et al. (2026). Its question-guided visual initialization determines the search starting point, while its within-candidate paired reward determines which states are favored. Candidate-dependent retrieval and norm-matched fusion supply the content tested by that reward. Table 13 compares these operations with DMLR, followed by discussion of other latent-space approaches.

Table 13: **Mechanism comparison between DMLR and VALR.** The seven dimensions separate shared latent optimization from differences in state construction, visual integration, and evaluation.

Dimension	DMLR	VALR (Ours)
Initialization	Latent input tokens and an attention-initialized patch pool	Question-attention-weighted visual content fused into latent positions
Candidate sampling	Gaussian latent perturbations	Gaussian perturbations of the search center
Reward	Complement of mean top-k truncated entropy	Real/control entropy difference on a shared vocabulary support
Policy update	Confidence-guided REINFORCE	REINFORCE using the signed paired reward
Visual retrieval	Latent attention with a reward-retained patch pool	Per-position cosine retrieval recomputed for each proposal
Visual integration	Selected patches interleaved as sequence positions	Aggregated evidence fused into fixed latent positions
Candidate selection	Reward-gated patch retention and decoding from the updated interleaved state	Highest paired-score real latent state retained for generation

DMLR. DMLR Liu et al. (2026) combines Gaussian latent policy optimization with dynamic visual injection. Its latent-token attention selects image patches, which are interleaved with the current state and a retained patch pool. Absolute-confidence rewards guide optimization and determine whether newly selected patches enter that pool. VALR evaluates the same proposal with retrieved evidence and with a norm-matched anchor-only control, retaining the original image in both branches. Their entropy difference guides both the center update and selection of the real conditioned state. This paired objective complements the change from an interleaved patch sequence to evidence fused into fixed latent positions.

LTPO. LTPO Ye et al. (2026) initializes latent thoughts from placeholder embeddings and refines them with Gaussian exploration and a confidence-based policy gradient. Its reward is obtained from the frozen model's output distributions, allowing each optimization step to be evaluated before complete-answer decoding. VALR reuses this score-function update to optimize the entropy difference between matched versions of each candidate. The comparison holds the sampled state fixed while varying its newly retrieved visual content, making evidence-induced confidence gain the feedback for optimization.

LatentSeek. LatentSeek Li et al. (2025) starts from hidden representations of a generated reasoning sequence and optimizes a fraction of its token-wise states before the language-model head. Updated states are decoded into tokens, completed into answers, and evaluated through self-reward prompting. VALR instead optimizes appended input embeddings and obtains paired feedback directly from next-token distributions at matched latent positions. The two methods share instance-level latent refinement, but operate at different representation locations and obtain feedback from different stages of inference.

LEAD. LEAD Xu et al. (2026) acts during autoregressive reasoning, using entropy to switch between discrete decoding and probability-weighted continuous embeddings. At entry into a high-entropy latent phase, it injects an anchor formed from pretrained visual-special-token embeddings. VALR shares the use of a model-native visual anchor, but combines it with image content retrieved for each sampled candidate. Here entropy scores the effect of this content during pre-generation optimization, while in LEAD it governs transitions between reasoning modes.

Coconut. Coconut Hao et al. (2025) feeds the model's last hidden state back as the next input embedding and learns continuous reasoning through a curriculum that progressively replaces textual reasoning steps. Its latent trajectory is extended through successive forward passes. VALR optimizes a fixed-size block of input embeddings within the frozen pretrained model. Successive updates revise this block in place, and the selected block then conditions answer generation.

SoftCoT. SoftCoT Xu et al. (2025a) uses a frozen assistant model to construct soft thoughts and trains a projection into the target model's embedding space. These projected representations then guide textual reasoning. VALR works directly in the frozen target model's input space, adapting its latent block to each image–question pair through inference-time exploration.

SoftCoT++. SoftCoT++ Xu et al. (2025b) extends soft-thought generation with distinct initialization tokens and contrastive learning to encourage multiple latent reasoning paths. Its test-time scaling uses this learned diversity to produce parallel paths and aggregate their answers. VALR samples and updates one shared search center, then generates from the candidate with the highest paired reward. Its search budget thus refines candidate states sequentially before a single answer is generated.

Unsilencing visual latents. Zhang et al. (2026) rank visual patches using question attention and assign disjoint positive and negative sets to latent positions for contrastive warm-up. A second stage uses positive entropy reductions across adjacent latent positions to reinforce confidence progression. VALR adapts question-ranked visual allocation to initialize latent states directly with the existing fusion operator. Its paired reward measures the effect of retrieved content within one candidate, while the progression reward measures changes across latent positions.

LatentStream. LatentStream Qu et al. (2026) contextualizes latent memory with a query and hierarchical video memory, then retrieves evidence over progressively broader temporal scopes. It jointly optimizes latent memory and injected evidence with absolute-confidence and hierarchical-progression rewards. For a static image–question pair, VALR refreshes retrieval over fixed projected image embeddings and optimizes the latent block. Its paired reward compares the effect of current retrieved content within a candidate, while LatentStream's progression term relates successive memory scopes.

Connection to the empirical analysis. Section 3 motivates visual conditioning by showing that confidence recovery can coexist with weak visual reliance and that confidence-only exploration increases high-confidence errors. The component and matched controls in Section 5.3 distinguish state initialization, paired evaluation, adaptive search, and selective retrieval. The benchmark comparison with DMLR evaluates the complete methods, while the within-VALR controls address individual design choices.

Table 14: **Paired comparisons with the strongest listed baseline.** VALR uses the three-run mean from Table 15. Accuracy is in percent, and gains and intervals are in percentage points.

Backbone	Benchmark	Baseline	VALR	Δ	95% CI
R1-OneVision	POPE-A	83.80	85.83	+2.03	[+1.46, +2.60]
R1-OneVision	HallusionBench	62.57	64.77	+2.21	[+0.77, +3.64]
Vision-R1	POPE-A	86.23	88.23	+2.00	[+1.40, +2.60]
Vision-R1	HallusionBench	66.77	68.98	+2.21	[+0.83, +3.58]
ThinkLite-VL	POPE-A	84.63	86.77	+2.13	[+1.59, +2.67]
ThinkLite-VL	HallusionBench	64.04	66.25	+2.21	[+1.00, +3.42]

Table 15: **Three-seed stability.** Individual-run accuracies and mean ± sample standard deviation across three runs.

Backbone	Benchmark	Vanilla	Run 1	Run 2	Run 3	VALR
R1-OneVision	POPE-A	82.67	85.67	85.80	86.03	85.83 ± 0.19
R1-OneVision	HallusionBench	61.41	64.67	64.98	64.67	64.77 ± 0.18
Vision-R1	POPE-A	83.47	88.00	88.30	88.40	88.23 ± 0.21
Vision-R1	HallusionBench	64.14	68.98	68.87	69.09	68.98 ± 0.11
ThinkLite-VL	POPE-A	83.40	86.90	86.60	86.80	86.77 ± 0.15
ThinkLite-VL	HallusionBench	62.78	66.46	66.04	66.25	66.25 ± 0.21

D More Experimental Results and Analysis

D.1 Statistical Robustness

The paired comparisons in Table 14 use the highest-accuracy listed baseline for each backbone–benchmark pair. For question i, let $c_{i,s} \in \{0,1\}$ indicate VALR correctness under seed s, and let $b_i \in \{0,1\}$ indicate correctness of the fixed baseline prediction. Using the same $S = 3$ runs reported in Table 15, we first average correctness within each question as $\bar{c}_i = S^{-1} \sum_{s=1}^{S} c_{i,s}$, then form the paired difference $d_i = \bar{c}_i - b_i$. Let $\bar{d} = n^{-1} \sum_{i=1}^{n} d_i$ over the n evaluated questions. Using the sample standard deviation $s_d = \sqrt{\sum_{i=1}^{n} (d_i - \bar{d})^2/(n-1)}$, we compute the accuracy gain and its two-sided 95% normal-approximation interval as

$$\widehat{\Delta} = 100\bar{d}, \qquad \mathrm{CI}_{95\%} = 100 \left[\bar{d} - 1.96 \frac{s_d}{\sqrt{n}},\ \bar{d} + 1.96 \frac{s_d}{\sqrt{n}} \right]. \tag{24}$$

The statistical units are the n questions, with seed averaging performed before estimating s_d. We report individual, unadjusted question-level normal intervals under independence, using the analytical standard error above. All calculations use full precision before rounding the reported gains and interval endpoints to two decimal places.

Accuracy gains range from 2.00 to 2.21 percentage points across POPE-Adversarial and HallusionBench, with all 95% confidence intervals excluding zero. Table 15 reports individual-run accuracies and their mean and sample standard deviation across three seeds. The means match both Table 14 and the main results, with standard deviations ranging from 0.11 to 0.21 percentage points across the six settings.

The paired intervals and the seed variation describe different sources of uncertainty. Each question contributes its seed-averaged correctness difference, so d_i can be fractional. The intervals describe question-level variation conditional on these runs and the fixed baseline predictions. The three-seed standard deviation instead measures variation across repeated VALR runs on the same benchmark. Its small magnitude relative to the reported gains indicates that the improvement is consistent across these runs, complementing the question-level paired comparison.

D.2 Component and Control Protocols

The comparisons in Table 7(a,b) use R1-OneVision on the same POPE-Adversarial and HallusionBench evaluation sets. Latent variants retain the original image and prompt, $K = 8$, and the same

final decoding settings. Initialization and scoring are varied only where indicated by the component labels.

Visual initialization and paired reward. Table 7(a) compares full VALR with separate removals of visual initialization and paired reward, alongside Vanilla. All three latent configurations preserve dynamic retrieval, norm-matched fusion, and iterative exploration. Visual initialization uses Appendix C.2, while its removal sets every initial position to $E(\tau_{\text{lat}})$. Paired reward uses Equation 11, while absolute-confidence scoring evaluates the predictive concentration of the real conditioned candidate. The two checked components identify full VALR. Vanilla generates directly from the original image and prompt.

Diagnostic variants. The confidence-only and visual-conditioning-only variants in Tables 10 and 11 serve as complementary diagnostics. Confidence-only exploration follows Appendix B.2. Visual conditioning only applies retrieval, aggregation, norm matching, and fusion once to the placeholder-initialized state and then generates directly. These diagnostics vary visual fusion and iterative search, whereas the component ablation removes visual initialization or paired scoring individually from full VALR.

Random search. This control draws 15 independent candidates $\widetilde{\mathcal{H}}^{(t)} = \mathcal{H}^{(0)} + \epsilon^{(t)}$, with $\epsilon^{(t)} \sim \mathcal{N}(0, \sigma^2 I)$ and the same fixed 10% perturbation rule. The center remains at the question-guided visual initialization. Each proposal undergoes aligned retrieval, real/control construction, and paired evaluation exactly as in full VALR. The highest-scoring real branch is retained for generation. This design isolates adaptive center updates from best-of-budget selection.

Random visual tokens. Initialization remains question-guided. During iterative retrieval, each latent position independently draws $\min(R, N)$ tokens uniformly without replacement from the valid image pool. Sets may overlap across positions and are resampled at each iteration. Within each set, aggregation retains the cosine-similarity softmax weights in Equation 6. The anchor, norm matching, fusion, paired reward, center update, and candidate retention follow full VALR. A separate random-number stream for token selection preserves the Gaussian perturbation sequence used in the matched comparison.

Budget matching. Full VALR and both matched controls evaluate 15 proposals, each with a real/control pair, with early stopping disabled. They share visual initialization and the two-branch scoring procedure. The comparison matches proposal and branch-evaluation counts, while measured latency additionally includes retrieval and cache handling. The confidence-only diagnostic evaluates one scoring branch per proposal, whereas Vanilla generates directly and the single-pass visual-conditioning diagnostic performs one fusion before generation.

D.3 Complete Ablation Results

Tables 17 and 18 extend the main-text ablations to the learning rate, perturbation scale, and visual-token aggregation rule.

Candidate selection. Table 16 compares absolute confidence and paired reward on the same fixed diagnostic candidate set for each question. Each set contains 15 proposal pairs generated with seed 2718, independently of full VALR's adaptive search path. The comparison uses 951 HallusionBench questions and a fixed 304-question diagnostic subset sampled from the official MathVision evaluation pool. This comparison tests the scoring criterion used to retain the highest-scoring candidate, with candidate generation held fixed.

We compute two-sided 95% confidence intervals for the accuracy difference using the paired normal approximation in Equation 24. Here, d_i is the paired selector's correctness indicator minus the absolute-confidence selector's indicator on question i, taking values in $\{-1, 0, 1\}$. The interval uses the sample standard deviation s_d and standard error $s_d/\sqrt{N}$, with critical value 1.96. The paired-only/absolute-only correct counts are 44/23, 39/23, 42/23, 23/14, 19/12, and 18/12 in table row order.

Table 16: **Candidate selection with absolute confidence and paired reward.** Both selectors use identical candidate sets. N counts questions. Accuracy is in percent, and Δ and its 95% CI give the paired-minus-absolute difference in percentage points.

Dataset	Backbone	N	Absolute confidence Acc.↑	Paired reward Acc.↑	Δ ↑	95% CI
HallusionBench	R1-OneVision	951	61.93	64.14	+2.21	[+0.53, +3.89]
HallusionBench	Vision-R1	951	64.56	66.25	+1.68	[+0.06, +3.30]
HallusionBench	ThinkLite-VL	951	62.46	64.46	+2.00	[+0.34, +3.66]
MathVision	R1-OneVision	304	28.29	31.25	+2.96	[-0.95, +6.87]
MathVision	Vision-R1	304	34.21	36.51	+2.30	[-1.28, +5.89]
MathVision	ThinkLite-VL	304	31.58	33.55	+1.97	[-1.56, +5.50]

Table 17: **Sensitivity to learning rate and perturbation scale on R1-OneVision.** Results are accuracy (%).

Parameter	Value	POPE-A	HallusionBench
η	5×10^{-4}	84.23	63.09
	$\mathbf{1\times10^{-3}}$	**85.83**	**64.77**
	2×10^{-3}	84.40	63.51
	5×10^{-3}	83.90	62.67
Perturbation scale	5%	84.27	63.30
	10%	**85.83**	**64.77**
	20%	84.20	63.20
	30%	83.83	62.67

On HallusionBench, paired reward improves selection accuracy by 1.68–2.21 percentage points across the three backbones, with all three 95% confidence intervals above zero. MathVision also shows positive gains across all three backbones, although the intervals include zero.

Update size and proposal scale. The learning rate η scales the change in the search center after a candidate is evaluated. The perturbation scale σ sets the spread of the proposal distribution and also appears in the score-function factor ϵ/σ^2. These parameters therefore affect different parts of exploration. The default $\eta = 10^{-3}$ and 10% perturbation scale achieve the highest accuracy in their respective sweeps on both benchmarks. Moving either below or above its default reduces accuracy, although every tested setting remains above Vanilla. The non-monotonic pattern favors a moderate local search over simply increasing the update size or sampling more distant candidates.

Visual-token aggregation. Table 18 compares soft and hard aggregation on the complete Count, Position, Color, and OCR subtasks of MME (Fu et al., 2023) and the 951 image-based HallusionBench questions. For each MME subtask, the official score adds question-level accuracy (Acc) and the proportion of images for which both paired questions are answered correctly (Acc+), with both quantities expressed in percent and a maximum combined score of 200. HallusionBench reports accuracy in percent.

Soft aggregation combines the top-16 tokens retrieved for each latent position using the cosine-similarity softmax weights in Equation 6. Hard aggregation instead uses the single highest-similarity token, setting $v^{(t)}_{\mathrm{ctx},i} = v_{j_i^*}$ with $j_i^* = \arg\max_j s^{(t)}_{ij}$. Both recompute retrieval for each candidate and use the same visual-placeholder anchor, norm matching, and fusion rule. The comparison fixes $K = 8$, $N_{\mathrm{opt}} = 15$, $\eta = 10^{-3}$, the 10% perturbation scale, $\lambda = 0.70$, and paired scoring with $k = 16$. Both variants share question-guided initialization, control construction, and decoding settings, and hard aggregation inherits the soft-aggregation hyperparameters.

Both soft and hard aggregation improve over Vanilla in every reported column. Their relative performance nevertheless varies across tasks: soft aggregation performs better on HallusionBench, Count, and Color, whereas hard aggregation performs better on Position and OCR.

Table 18: **Soft and hard visual-token aggregation on R1-OneVision.**

Method	MME score↑				HallusionBench
	Count	Position	Color	OCR	Acc. (%)↑
Vanilla	143.33	68.33	173.33	123.33	61.41
Soft aggregation	**148.33**	71.67	**176.67**	126.67	**64.77**
Hard aggregation	145.00	**73.33**	175.00	**128.33**	63.20

D.4 Visual Evidence Dependence

The two analyses in Table 8(a,b) evaluate R1-OneVision on all 951 image-based HallusionBench questions. Token deletion tests sensitivity to the representations selected by VALR, while the visual margin compares answer support under original and spatially shuffled images.

Selected-token deletion. We first run VALR on the original image and record the retrieval sets associated with its retained, highest-reward conditioned candidate. Their union across the K latent positions, $S = \bigcup_{i=1}^{K} \mathcal{N}_i^{(t^*)}$, defines the deletion set, with duplicate token indices removed. Holding S fixed, we mask its visual keys in the initialization prefill, both scoring branches, and subsequent decoding, and exclude its tokens from initialization allocation and iterative retrieval. We rebuild question-guided initialization from the remaining pool, then rerun exploration and generation with the same random seed and rebuilt KV caches. The perturbation scale is recomputed from this deletion-conditioned initialization using Equation 23. The random-deletion control samples $|S|$ valid visual tokens without replacement, using a fixed, recorded seed and the same rerun procedure. This control may overlap S. The original pixels remain unchanged, and other visual tokens may still carry related information, so the experiment measures sensitivity to encoded representations under re-exploration.

Because the retained latent state has already incorporated the selected visual content, we rebuild it under the deletion condition. The selected representations are removed from initialization, retrieval, and attention before re-exploration. Selected-token deletion produces a larger accuracy drop than equal-count random deletion and changes more predictions (Table 8(a)).

Clean–counterfactual visual margin. For each preprocessed image I, we construct $\widetilde{I}$ by dividing it into an 8×8 grid and shuffling the grid cells with a fixed random seed. Vanilla and VALR receive the same original/shuffled image pair and score the same ground-truth Yes/No answer $a^\star$. For each input, the scoring context comprises the image, the same question q, and a fixed answer prefix. VALR independently initializes and explores a latent state for each image input, rebuilding both scoring branches and their caches for the shuffled image. For method m, the length-normalized visual margin is

$$M_{\text{vis}}^{(m)} = \frac{\log P_m(a^\star \mid I, q) - \log P_m(a^\star \mid \widetilde{I}, q)}{|a^\star|}. \tag{25}$$

The shared answer prefix is implicit in the notation, and P_m includes the input-specific retained real latent state for VALR. We teacher-force the complete answer string, sum its token log-probabilities using natural logarithms, and divide by the number of answer tokens, excluding EOS. Answer whitespace, capitalization, and tokenization are held fixed across branches, giving a margin in nats per token. This score measures the same reference answer's dependence on the image under a controlled prompt. This protocol scores a fixed reference answer across two images. Section 3.2 scores the predicted Yes/No answer, while VALR's paired reward compares latent branches that share the original image. Groups are determined by final predictions on the original images and remain fixed for the shuffled-image comparison. The corrected group contains questions answered incorrectly by Vanilla and correctly by VALR, and each row reports the mean margin within its group.

The Vanilla-correct and Vanilla-wrong groups partition the 951 questions into 584 and 367 samples. The 96 VALR-corrected questions form a subset of the latter group. Their mean margin increases from 0.23 to 0.55 nats per token, the largest gain among the reported groups. This comparison links successful corrections to greater reference-answer support from the original image relative to its shuffled counterpart. This measurement perturbs spatial arrangement and tracks the resulting answer likelihood, whereas token deletion removes selected representations.

Table 19: **Inference time by reasoning length on R1-OneVision.**

Length	N	Avg. tokens	Vanilla	VALR	Overhead
≤ 300	465	218	09:42	12:36	+2:54
301–600	332	463	16:18	19:12	+2:54
> 600	154	782	27:35	30:29	+2:54

Table 20: **Optional early stopping on R1-OneVision and HallusionBench.**

ϵ	Avg. steps	Acc.	Time
fixed	15.0	**64.77**	17:48
0.005	11.2	64.67	17:01
0.010	8.3	64.46	16:25
0.020	5.7	64.04	16:00

D.5 Inference-Cost Details

Timing includes question-guided initialization, retrieval, both candidate-scoring branches, cache handling, and final generation. In Table 8(d), one optimization step denotes one proposal evaluated as a pair, and the zero-step row reports direct Vanilla generation from the original image and prompt. VALR adds 2:54 over Vanilla in each response-length group (Table 19). The sample-weighted times are 14:54 for Vanilla and 17:48 for VALR, matching Table 8(c). The constant overhead across these groups is consistent with latent exploration preceding generation.

As responses become longer, this exploration overhead accounts for a smaller fraction of total inference time. The absolute cost of exploration and its relative contribution should therefore be distinguished when comparing workloads with different reasoning lengths. The fixed-budget comparison in Table 8(d) addresses the other side of this trade-off by reducing candidate evaluations directly. Eight steps retain about 93% and 94% of the 15-step accuracy gains over Vanilla on POPE-A and HallusionBench, respectively, indicating that much of the measured benefit is already obtained before the full budget is exhausted.

Optional early stopping. The fixed 15-step budget remains the default used in the main results. For the optional early-stopping variant, we monitor the best paired candidate reward obtained so far:

$$B_t = \max_{0 \leq j \leq t} r^{(j)}. \tag{26}$$

After the first candidate evaluation, we initialize a reference reward $b_{\text{ref}} = B_0$ and a patience counter $c = 0$. At each subsequent evaluation, if $B_t - b_{\text{ref}} > \epsilon$, we set $b_{\text{ref}} = B_t$ and reset c to zero. Otherwise, we increment c. The threshold therefore tests the accumulated absolute improvement since the last reference update. The counter starts accumulating at the second evaluation. We stop once at least five candidates have been evaluated and $c \geq 3$. Exploration ends after at most 15 candidate evaluations. Independently of this threshold, we retain the real branch of the highest-scoring evaluated pair for generation. Table 20 uses thresholds $\epsilon \in \{0.005, 0.010, 0.020\}$. At $\epsilon = 0.010$, early stopping reduces the average exploration budget to 8.3 steps and inference time to 16:25, with an accuracy decrease of 0.32 percentage points.

The threshold measures the absolute paired-reward improvement in nats since the last reference update. Small improvements can accumulate until they exceed ϵ, because the reference reward changes only when that threshold is crossed. A stricter requirement for improvement consequently triggers stopping sooner when the best reward plateaus. The three thresholds in Table 20 show the resulting trade-off: larger thresholds reduce the average number of evaluations but also lower accuracy.

D.6 Radar Chart Results

Across the three evaluated backbones, VALR achieves higher scores than the displayed baselines on the plotted hallucination and reasoning benchmarks (Figure 6).

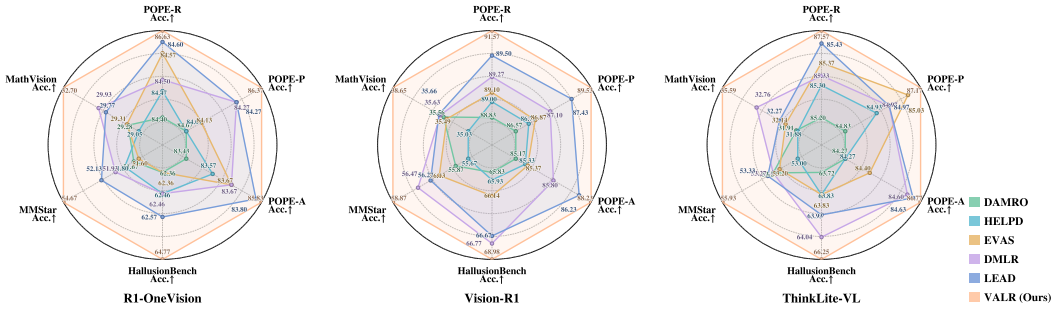


Figure 6: Radar chart comparison of inference-time methods across hallucination and broader reasoning benchmarks with R1-OneVision Yang et al. (2025), Vision-R1 Huang et al. (2026), and ThinkLite-VL Wang et al. (2025).

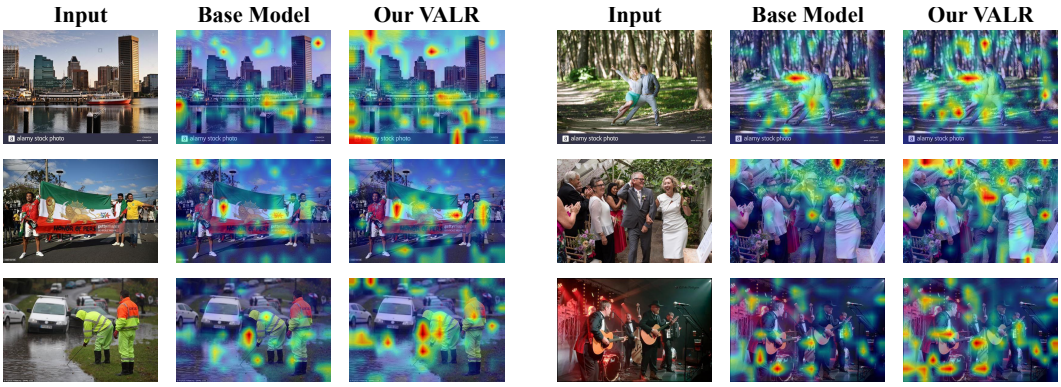


Figure 7: **Visual-attention comparison.** VALR places more attention on task-relevant regions in the displayed examples.

D.7 Heatmap Comparison

Figure 7 compares the visual-attention proxy produced by the base model and VALR. In the displayed scenes, VALR assigns more attention to objects and relations relevant to the query, whereas the base model's attention is more diffuse.

D.8 Additional Qualitative Comparisons

The reasoning examples in Figures 8 and 9 show how an unsupported visual interpretation can alter an otherwise coherent calculation. In the upper example of Figure 8, Vanilla first obtains a shaded area of 1, then revises it to $3/2$ after introducing an additional geometric claim. VALR retains the decomposition into a triangle of area 2 and a unit square. The lower example follows a similar pattern: Vanilla correctly derives a triangle area of 54 from the similarity ratio, but doubles it because the larger triangle appears vertically stretched. VALR follows the stated similarity relation through to the final answer and retains the original scale.

Figure 9 shifts the emphasis from geometric calculation to local visual relations. In the hand-holding example, VALR distinguishes a child's left-hand contact from the right-hand contacts being counted, changing the answer from seven to six. In the grid example, Vanilla identifies two repaintings before adding an extra one for a purported white-square adjacency. VALR limits its count to the depicted conflicts and answers two. Across these examples, the difference lies in which visual premises enter the reasoning, including premises introduced after a plausible intermediate answer has already been reached.

E Qualitative Case Studies

The image descriptions in Figures 10–16 complement the reasoning examples above. Here, the main subjects are often recognized by both methods, but their descriptions diverge as they add background objects, infer activities, or name fine-grained details. We examine these points of divergence to make the qualitative changes more concrete.

E.1 From Visible Objects to an Invented Scene

In the guitar-playing scene (Figure 10), Vanilla expands the visible person and guitar into a studio-session narrative, adding a recording microphone setup and a drum amplifier stack. These additions fit the musical setting but are not established by the visible equipment. VALR preserves the person, guitar, and posters while referring more broadly to nearby audio or electronic equipment. The response remains descriptive and uses broader labels for ambiguous background items.

The kayaking scene (Figure 11) illustrates a related shift from visible action to an inferred event. Vanilla adds a waterfall entrance, a professional racing suit, and a river competition with photographers. VALR limits its description to the helmet, kayak, paddle, splashing water, and surrounding vegetation. Both responses recognize the kayaker but differ in the inferred setting of the activity.

E.2 Specificity in Cluttered and Distant Regions

Cluttered interiors leave some object identities and room boundaries ambiguous. In Figure 12, Vanilla describes a large television, musical instrument cases, and a doorway leading into a kitchen, then interprets the scene as a musician moving into a studio. VALR uses broader descriptions of boxes, bags, boards, and an adjacent darker room. It still suggests moving or reorganization, but avoids several specific object and room identities that drive Vanilla's account.

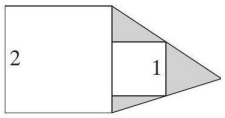
The balcony and harbor scenes show similar differences in the treatment of background content. In Figure 13, Vanilla adds a tennis court and a wine bottle and identifies the location as a high-end resort. VALR reports the table, pool, and mountain setting, leaving the venue type and additional objects unspecified. In Figure 14, Vanilla names a sports stadium and a luxury hotel and describes tourist-cruise departure activity. VALR describes the visible buildings and vessels, including their appearance, while leaving their exact functions and activities unspecified. In both cases, the descriptions differ most where the image provides less detail for a precise identification.

E.3 Interpreting Activities and Fine-Grained Details

People's clothing and nearby objects can suggest several plausible activities. In the flooded-road scene (Figure 15), Vanilla interprets the activity as a response to a traffic accident, adding rescue coordination and road-closure barriers. VALR describes the rain gear, long tool, standing water, and vehicles, presenting maintenance or flood monitoring as possible activities.

The wedding scene (Figure 16) involves a different kind of over-specificity. Vanilla suggests a live band and entry into a glass-roofed reception hall, and names the bouquet flowers as lilies and orchids. VALR retains the wedding participants and floral details, leaving the venue and flower species unspecified. Across these examples, VALR retains the main subjects and visible details while making fewer unsupported claims about object identities and activities.

The diagram shows two squares: one has a side with a length of 2 and the other (abut on the first square) has a side with a length of 1. What is the area of the shaded zone?



Base Model

<think> The figure contains a square of side length 2 attached to a triangular region, inside which there is a smaller square of side length 1. The shaded region consists of the remaining triangular parts outside the smaller square. Since the large triangle appears symmetric, its total area can be estimated as \(\frac{1}{2}\times 2 \times 2 = 2\). Subtracting the inner square area \(1^2=1\), the shaded area becomes 2 \(- 1 = 1\). However, because the upper and lower shaded triangles appear congruent and there is also a right shaded tip, the total shaded region should instead sum to \frac{3}{2}}.<...></think>

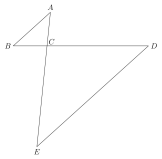
<answer> \boxed{\frac{3}{2}} </answer>

Our VALR

<think> The shaded region is the area inside the large triangle-like wedge but outside the small square of side length \(1\). The left side of the wedge spans the full side of the larger square, so its vertical height is \(2\). The wedge narrows linearly to a point on the right. Since the inner square has side length \(1\), its top and bottom corners lie halfway along the wedge height, implying the horizontal distance from the left boundary to the tip is also \(2\). Thus the enclosing triangular region has area \[\frac{1}{2}\times 2 \times 2 = 2.\]
The small square contributes area \[1^2 = 1.\]
Therefore, the shaded area is\[2 - 1 = 1.\]</think>

<answer> \boxed{1} </answer>

The area of $\\triangle ABC$ is 6 square centimeters. $\\overline{AB}\\|\\overline{DE}$. $BD=4BC$. What is the number of square centimeters in the area of $\\triangle CDE$?



Base Model

<think> Since \(\overline{AB}\parallel\overline{DE}\), triangles \(ABC\) and \(CDE\) are similar. We are told that \(BD=4BC\). Because \(BD\) includes both \(BC\) and \(CD\), we have\[CD = BD-BC = 4BC-BC = 3BC.\]
Thus the similarity ratio from \(\triangle ABC\) to \(\triangle CDE\) is\[\frac{CD}{BC}=3.\]
Areas scale by the square of the similarity ratio, so \[[\triangle CDE]=6\cdot 3^2=54.\]
However, since point \(C\) lies between the two triangles and the larger triangle appears stretched vertically, an additional scaling factor of 2 should be considered, giving \[54\times 2 = 108.\]</think>

<answer> \boxed{108} </answer>

Our VALR

<think> Because \(\overline{AB}\parallel\overline{DE}\), triangles \(ABC\) and \(CDE\) are similar. We are given \[BD=4BC.\]
Since \(BD=BC+CD\), it follows that \[CD=4BC-BC=3BC.\]
Therefore, the similarity ratio from \(\triangle ABC\) to \(\triangle CDE\) is \[\frac{CD}{BC}=3.\]
Areas of similar triangles scale by the square of the side ratio, so \[[\triangle CDE]=[\triangle ABC]\times 3^2=6\times 9=54.\]</think>

<answer> \boxed{54} </answer>

Figure 8: **Geometric reasoning comparisons.** The responses differ in whether additional geometric assumptions alter an initially valid calculation.

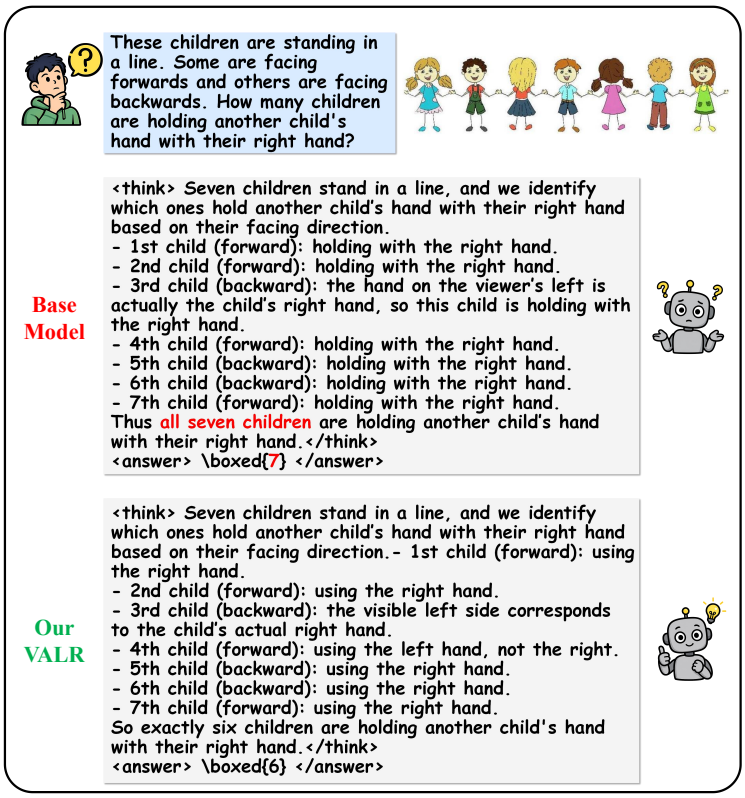


Figure 9: **Visual-relation reasoning comparisons.** Hand assignment and square adjacency change the counts used in the final answers.

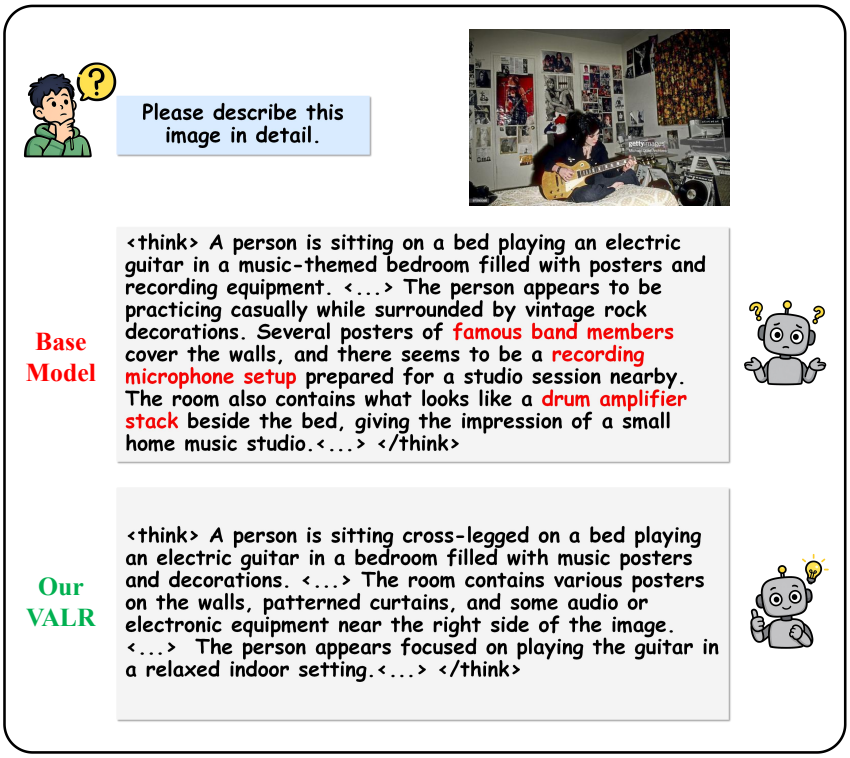


Figure 10: **Guitar-playing scene.** Comparison of visible objects and additional claims about equipment and setting.

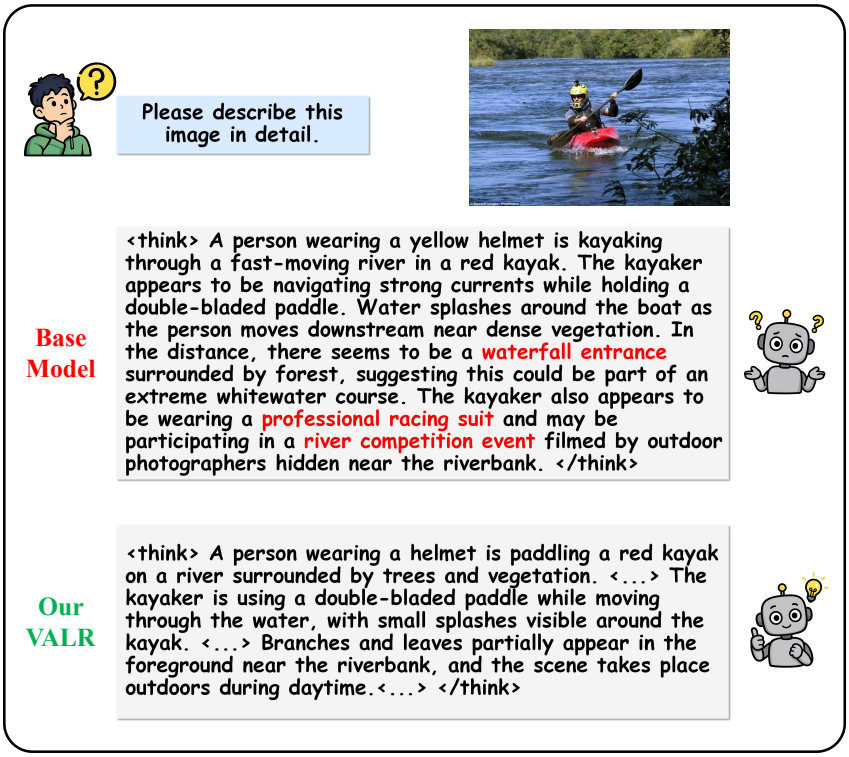


Figure 11: **Kayaking scene.** Comparison of the observed action and claims about a surrounding competitive event.

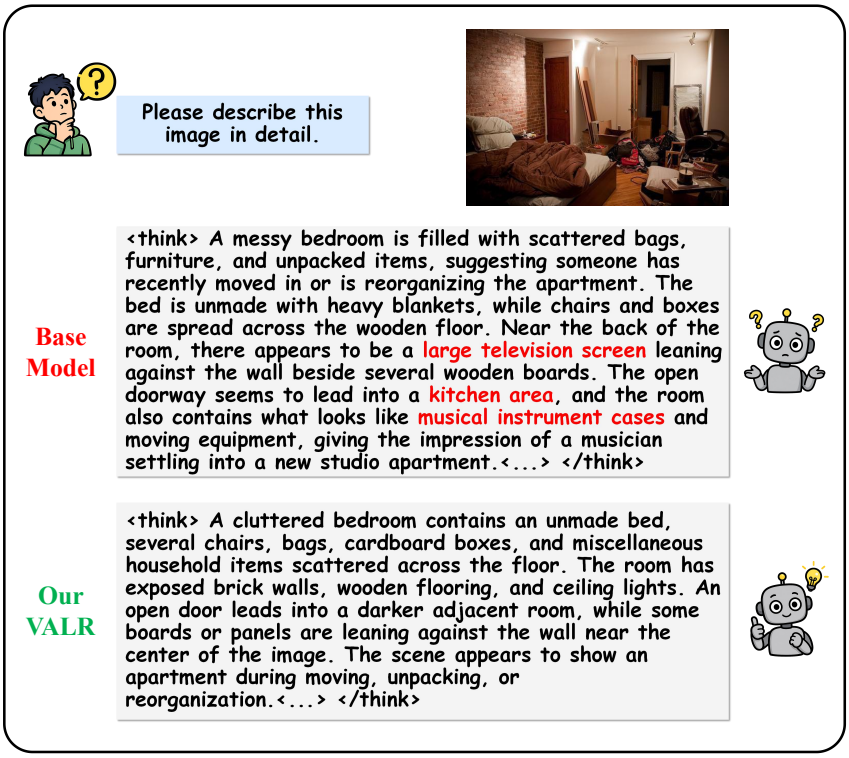


Figure 12: **Cluttered interior.** Comparison of object descriptions, room identities, and interpretations of the scene.

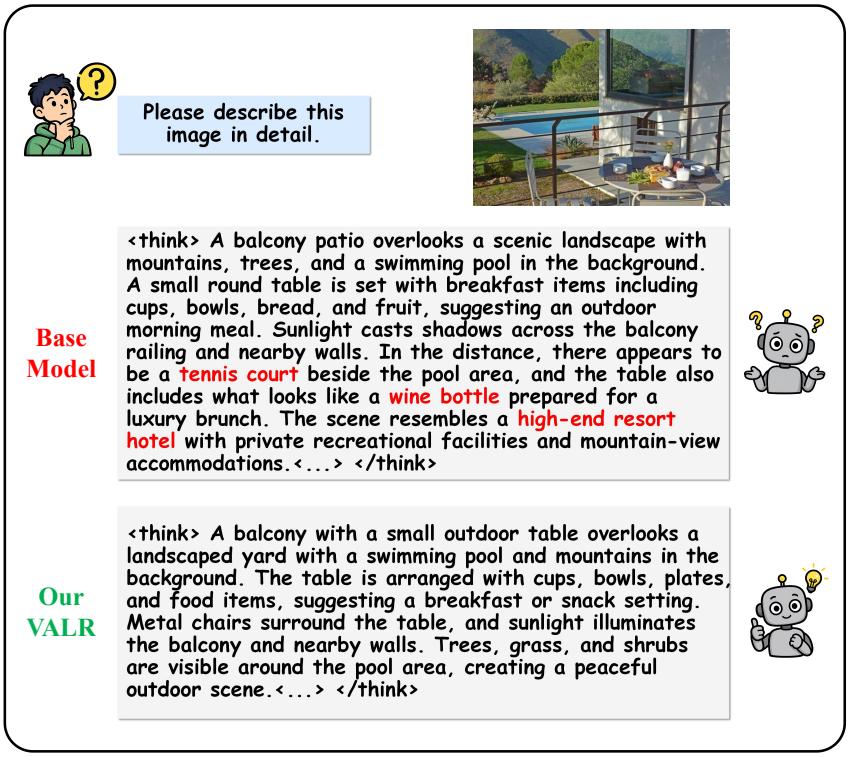


Figure 13: **Balcony and pool scene.** Comparison of background objects and claims about the type of venue.

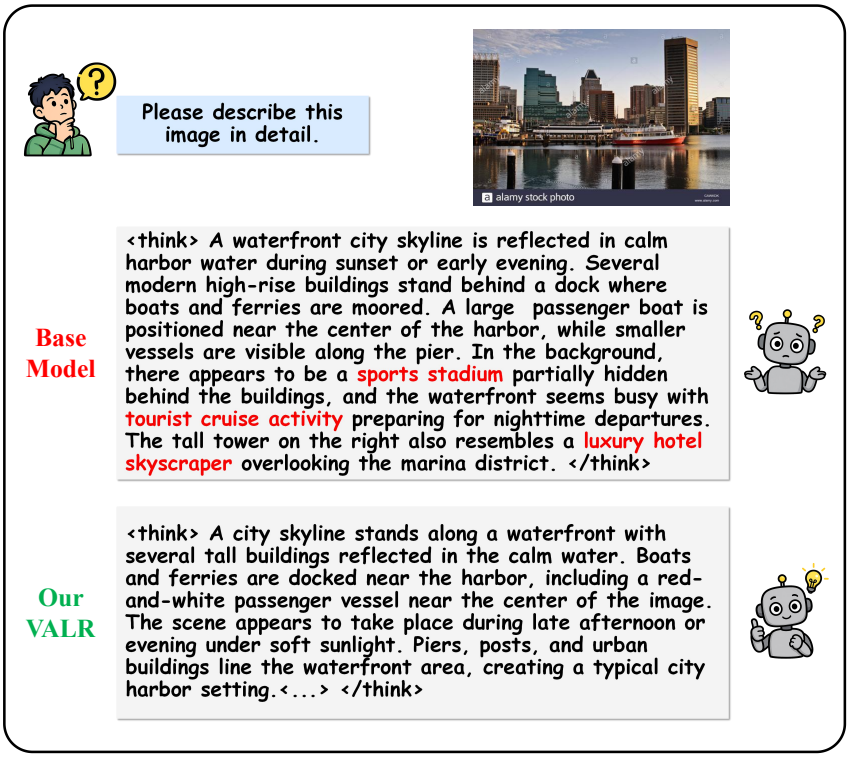


Figure 14: **Harbor scene.** Comparison of distant buildings and vessels with claims about their identities and activities.

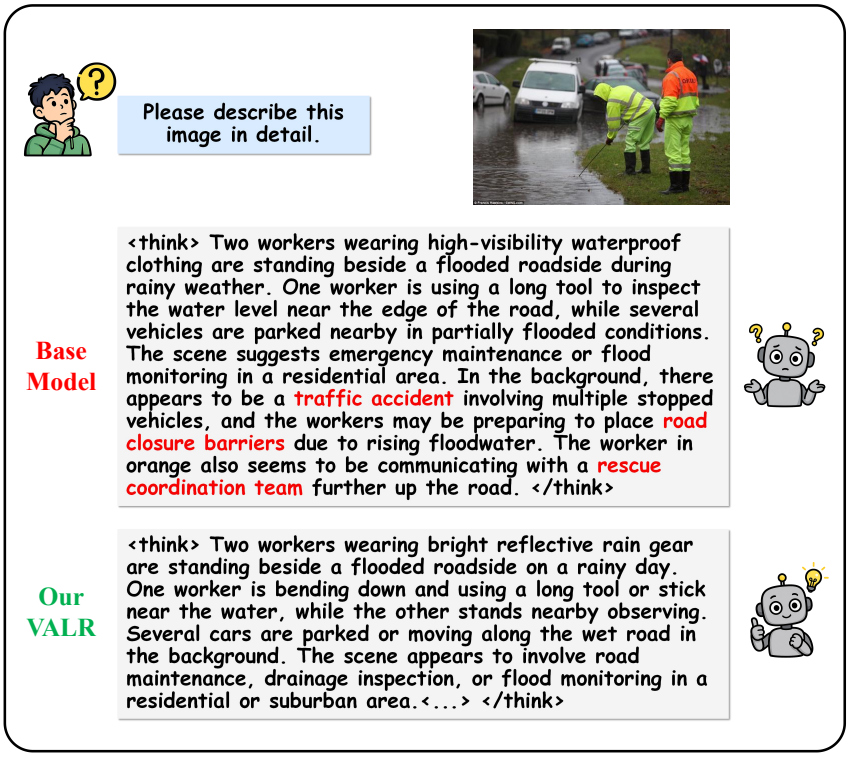


Figure 15: **Flooded-road scene.** Comparison of visible activity cues and interpretations of the event.

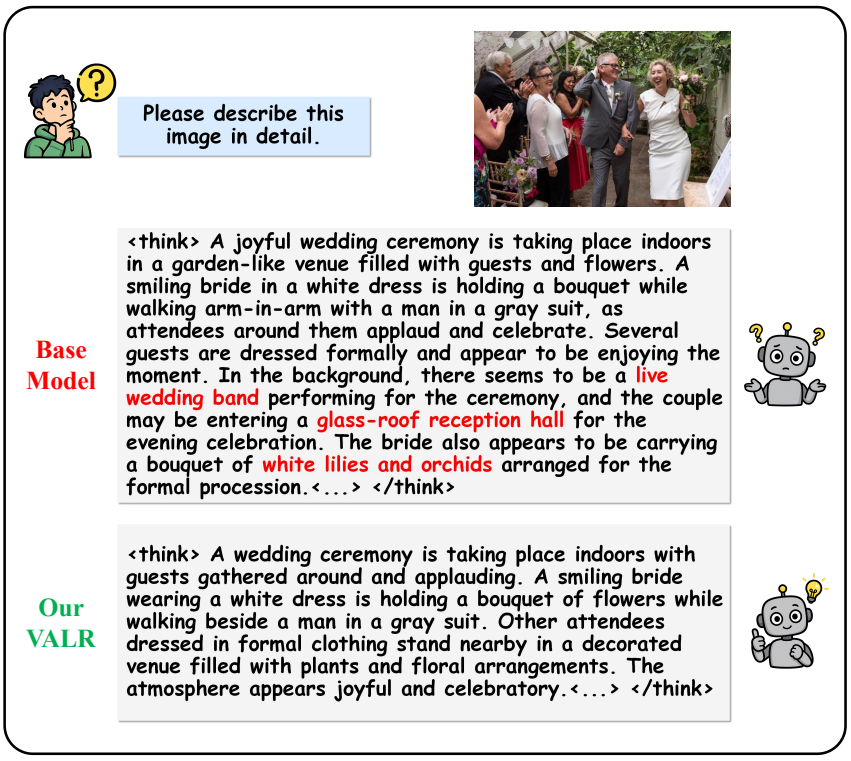


Figure 16: **Wedding scene.** Comparison of scene details and additional claims about background structures and flower species.