RETHINKING LATENT VISUAL REASONING: GROUNDING LATENT REASONING IN VISUAL EVIDENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Latent visual reasoning (LVR) enables multimodal large language models (MLLMs) to perform intermediate computation in continuous latent tokens rather than expressing every reasoning step in words. However, unlike textual CoT, latent reasoning is not directly observable, making it difficult to supervise what latent tokens learn. In this work, we first conduct a thorough analysis of latent-token behavior and identify a *latent evidence-credit gap*: latent tokens respond only weakly to image perturbations that alter the correct answer. We hypothesize that this issue stems from the lack of explicit supervision during GRPO training. These findings suggest that a final-answer reward provides too little guidance on what visual evidence to preserve or how credit should be assigned across latent tokens. To bridge this gap, we propose ReaLVR, which brings visual-evidence supervision to the model's own free-running latent trajectories. ReaLVR contrasts correct and model-generated wrong answers to determine where stronger supervision is needed, and relevant and mismatched visual evidence to specify what to preserve. Across three model families, ReaLVR consistently outperforms evaluated LVR baselines, achieving the highest five-task average of 63.7% on Qwen2.5-VL-7B. Crucially, we are the first to scale visual reasoning in latent space, showing that our framework continues to deliver robust improvements at frontier model scales up to 235B. Further analyses show more question-sensitive latent-token positions, stronger alignment with relevant visual regions, and greater fixed-context dependence on the most attended latent tokens.

1 INTRODUCTION

Latent visual reasoning (LVR) has emerged as an alternative to textual chain-of-thought reasoning in multimodal large language models (MLLMs), performing intermediate computation through continuous latent tokens rather than expressing every reasoning step in words (Li et al., 2026a; Wang et al., 2026b; Yang et al., 2026b; Dong et al., 2026; Hu et al., 2026; Jeon et al., 2026; Li et al., 2026b). This approach is especially appealing for problems involving spatial relationships and fine-grained visual details that are difficult to describe step by step. However, the underlying mechanisms of LVR remain unclear: what information the generated latent tokens encode, whether they respond to visual evidence that changes the correct answer, and how they contribute to the final answer? Assessing only final-answer accuracy is insufficient to address these questions.

Our controlled behavioral and representation tests suggest that vanilla LVR does not reliably preserve answer-relevant visual evidence in its generated trajectory. Motivated by these observations, we examine how the generated trajectory is supervised. In the SFT stage, latent visual states are supervised to match target visual features. During RL and inference, the model instead generates a *free-running latent trajectory* without these targets. Standard GRPO (Shao et al., 2024b) only optimizes the generated text rather than the latent trajectory itself. Consequently, answer-level feedback provides no direct signal indicating which positions need stronger visual supervision or what evidence they should preserve. We term this missing connection the *latent evidence-credit gap*.

To bridge this gap, we propose ReaLVR, which directly trains free-running latent tokens to preserve answer-relevant visual evidence. ReaLVR regenerates the trajectory with the current model and retains gradients through its generation. To decide where stronger visual supervision is needed, it compares how the ground-truth answer and wrong answers sampled from the behavior policy attend to each latent token. To determine what the supervised tokens should preserve, ReaLVR contrasts answer-relevant visual evidence with mismatched evidence. Figure 1 provides an overview.

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

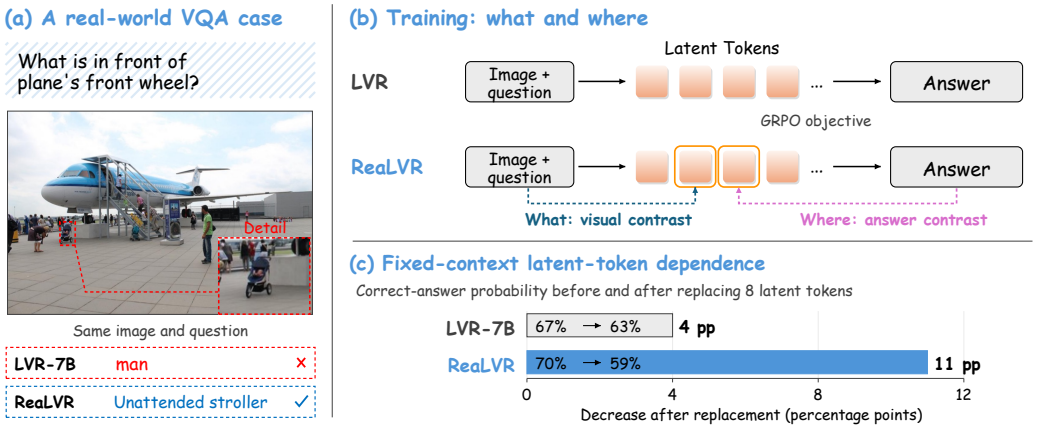


Figure 1: **ReaLVR connects visual evidence to latent reasoning.** (a) ReaLVR identifies the stroller missed by LVR-7B in a complex scene; dashed boxes mark image details. (b) Visual and answer contrast determine *what to preserve* and *where to supervise*, during training only. (c) Replacing the top-8 tokens ranked by answer-to-token attention, with other states fixed, decreases correct-answer probability by 4 and 11 percentage points. The larger ReaLVR drop indicates greater local answer dependence.

We evaluate ReaLVR on five benchmarks with six backbones spanning three model families (`Qwen2.5-VL`/`Qwen3-VL`, `InternVL3`, and `Gemma-3`). ReaLVR achieves the highest five-task average among the evaluated latent-reasoning methods on Qwen2.5-VL-7B, reaching 63.7%. On Qwen3-VL-8B, ReaLVR reaches an average of 61.2%, exceeding the evaluated latent-reasoning baselines. The gains extend to Qwen3-VL-30B, where ReaLVR reaches 65.2%, 1.1 points above LVR. On `Qwen3-VL-235B`, ReaLVR also improves over LVR-SFT on all three evaluated benchmarks. To our knowledge, this is the first demonstration of visual reasoning in latent space trained on a 235B-parameter multimodal backbone.

The contributions of this work are threefold. **First**, we identify the *latent evidence-credit gap* and show that vanilla LVR does not reliably preserve answer-relevant visual evidence in its generated trajectory. **Second**, we introduce ReaLVR, which uses answer comparison to determine where stronger visual supervision is needed and visual evidence comparison to determine what the supervised tokens should preserve. This additional supervision requires no change to the model architecture or inference procedure. **Third**, we evaluate six backbones up to 235B parameters. To the best of our knowledge, we provide the first demonstration that continuous latent visual reasoning can be trained at frontier scale. The resulting latent states remain input-dependent and carry answer-relevant information rather than collapsing to a fixed trajectory. Through this work, we call for more attention toward demystifying the internal dynamics of continuous latent reasoning beyond benchmark accuracy, laying a grounded foundation for robust and faithful multimodal systems.

2 Related Work

Visual Reasoning in Latent Space. Latent visual reasoning (LVR) performs intermediate computation through continuous latent tokens rather than explicit textual rationales (Li et al., 2026a; Wang et al., 2026b; Yang et al., 2026b; Dong et al., 2026). Recent work has explored a range of latent trajectory designs. Some methods switch or interleave textual and visual reasoning, adapt the number of latent states, or combine text and image representations within a shared latent workspace (Tong et al., 2025; Chen et al., 2026b; Tong et al., 2026; Chen et al., 2026a; Jiang et al., 2026). Others introduce structured or coarse-to-fine trajectories (Viveiros et al., 2026a; Wang et al., 2026d), or support long, parallel, decomposed, progressive, and multi-hypothesis reasoning (Wang et al., 2026a; Lu et al., 2026; Zhu et al., 2026b; Li et al., 2026c; Huang & Shan, 2026; Tang et al.). These methods expand the form and flexibility of latent computation. However, they leave open how to ensure that a free-running latent trajectory preserves the visual evidence required for its answer.

Visually Grounded Multimodal Reasoning. A broad line of work grounds multimodal reasoning in observable image evidence. VisCoT selects relevant regions (Shao et al., 2024a); PixelReasoner and DeepEyes revisit images through pixel-space operations or visual tools (Su et al., 2025; Zheng et al., 2026); and Argus and grounded chain-of-thought methods make regions or coordinates ex-

plicit during reasoning (Man et al., 2025; Wu et al., 2026b; Xia et al., 2025). Other methods learn multi-turn grounding from final-answer rewards or guide policy updates with verifiable perception questions (Huang et al., 2026b; Zhang et al., 2026a). For continuous latent reasoning, methods use semantic or attention-trajectory targets (Xu et al., 2026; Wu et al., 2026a), align states with visual features, regions, relations, or contrastive objectives (Miao et al., 2026; Cui et al., 2026; Wang et al., 2026e; Ding et al., 2026), or develop latent-specific policy objectives (Cheng et al., 2026; Zhu et al., 2026a). RoT instead uses rendered textual CoT rather than targets from the input image (Wang et al., 2026c). Diagnostic studies go beyond accuracy and representation similarity to probe what latent states encode, how they respond to image evidence, and whether final answers depend on them (Li et al., 2026d; Viveiros et al., 2026b; Zhang et al., 2026b;c; Guo et al., 2026; Yang et al., 2026a; Park et al., 2026; Kang et al., 2026). Monet directly optimizes sampled latent trajectories (Wang et al., 2026b), CoLVR contrasts latent trajectories (Ding et al., 2026), and RIS supervises region evidence (Cui et al., 2026). ReaLVR uses correct-versus-wrong answer readout to weight positions within one differentiable current-model trajectory, then applies relevant-versus-mismatched visual supervision at those positions while retaining the LVR inference procedure.

3 PRELIMINARIES

3.1 LATENT VISUAL REASONING

Each example contains an image–question input x, a ground-truth answer $y^\star$, and optionally an evidence annotation a, such as a region of interest (ROI); $a = \emptyset$ means that no region annotation is available. Let π_θ denote the autoregressive policy, and let $\mathcal{T}_\theta(c)$ denote the decoder hidden state produced from a causal prefix c. The vision stack represents the image in x with N visual tokens $\mathbf{V} = \{v_n\}_{n=1}^{N}$, where $v_n \in \mathbb{R}^d$ and d is the shared visual-token and decoder hidden-state dimension. Latent Visual Reasoning (LVR) inserts continuous decoder states between the input and the textual answer (Li et al., 2026a). At each latent position, the decoder passes its hidden state directly to the next decoding step instead of mapping it to a vocabulary token. With K latent tokens, the generation order is

$$x \to \texttt{<|lvr_start|>}, z_1, \ldots, z_K, \texttt{<|lvr_end|>}, \text{answer},$$

where $z_t \in \mathbb{R}^d$. The block $z_{1:K}$ is the *latent span*. Let o denote the discrete control markers and answer text. A free-running rollout is $\tau = (z_{1:K}, o) \sim \pi_\theta(\cdot \mid x)$. If $c_t(\tau)$ is the causal prefix before latent position t, then $z_t = \mathcal{T}_\theta(c_t(\tau))$. Thus, a generated latent token can depend on the image, question, and earlier latent tokens, but never on future answer tokens.

3.2 TWO-STAGE LVR TRAINING

LVR first initializes its latent states with visual supervision and then post-trains the model using outcome feedback (Li et al., 2026a).

Stage 1: target-conditioned visual supervision. An ROI annotation is mapped to an ordered sequence of T_v visual targets $v^\star_{1:T_v}$. Under teacher forcing (TF), these target visual embeddings are supplied along the latent span instead of autoregressively feeding back the model's own generated latent states. The decoder hidden states $z^{\mathrm{TF}}_{1:T_v}$ are trained to reconstruct this target sequence. We therefore call them *target-conditioned*, to distinguish them from the free-running states used in Stage 2. Their visual reconstruction loss is

$$\mathcal{L}_{\mathrm{rec}}(\theta) = \frac{1}{T_v} \sum_{t=1}^{T_v} \left\| z_t^{\mathrm{TF}} - v_t^\star \right\|_2^2 . \tag{1}$$

Together with the standard next-token prediction loss, this stage produces parameters θ_0, which are held fixed as the reference policy during Stage 2. The target length T_v can differ from the free-running length K.

Stage 2: free-running outcome optimization. The frozen behavior policy $\pi_{\theta_{\mathrm{old}}}$ samples $\{\tau_i = (z^{\mathrm{roll}}_{i,1:K}, o_i)\}_{i=1}^{G}$. Here G is the group size, $z^{\mathrm{roll}}_{i,1:K}$ are the saved latent states, and $\widehat{y}_i$ is the canonical answer parsed from o_i, with $\widehat{y}_i = \perp$ for a parse failure. We write $\mathrm{Correct}(\widehat{y}_i, y^\star) \in \{0, 1\}$ for answer correctness.

The standard reward combines answer correctness and output format. Group Relative Policy Optimization (GRPO) converts the G rewards into fixed group-relative advantages $\widehat{\mathbf{A}} = (\widehat{A}_1, \ldots, \widehat{A}_G)$.

Let $\mathcal{J}_{\text{clip}}$ denote the clipped text-token GRPO objective, $\widehat{\mathcal{L}}_{\text{KL}}^{\text{text}}$ the sampled text-token Kullback–Leibler (KL) penalty to π_{θ_0}, and $\beta \geq 0$ its weight. Vanilla Stage 2 minimizes

$$\mathcal{L}_{\text{S2}}^{\text{LVR}}(\theta) = -\mathcal{J}_{\text{clip}}(\theta; \theta_{\text{old}}, \widehat{\mathbf{A}}) + \beta \widehat{\mathcal{L}}_{\text{KL}}^{\text{text}}(\theta; \theta_0). \tag{2}$$

Both terms score generated text positions. During policy replay, sampled latent vectors are treated as fixed context, so the policy loss does not backpropagate through the process that generated those vectors. Appendix B explains why this setup leaves free-running latent states without direct visual-evidence supervision and distinguishes readout, grounding, and intervention utility.

4 REALVR: OUTCOME-CONTRASTIVE EVIDENCE CREDIT

ReaLVR couples answer-contrastive position weighting with relevant-versus-mismatched visual supervision on one differentiable trajectory generated by the current model. Correct-versus-wrong answer readout assigns stronger supervision to selected latent positions; visual contrast defines the evidence those positions should preserve. The visual loss backpropagates through latent generation, updating the process used at inference without changing the architecture or inference procedure.

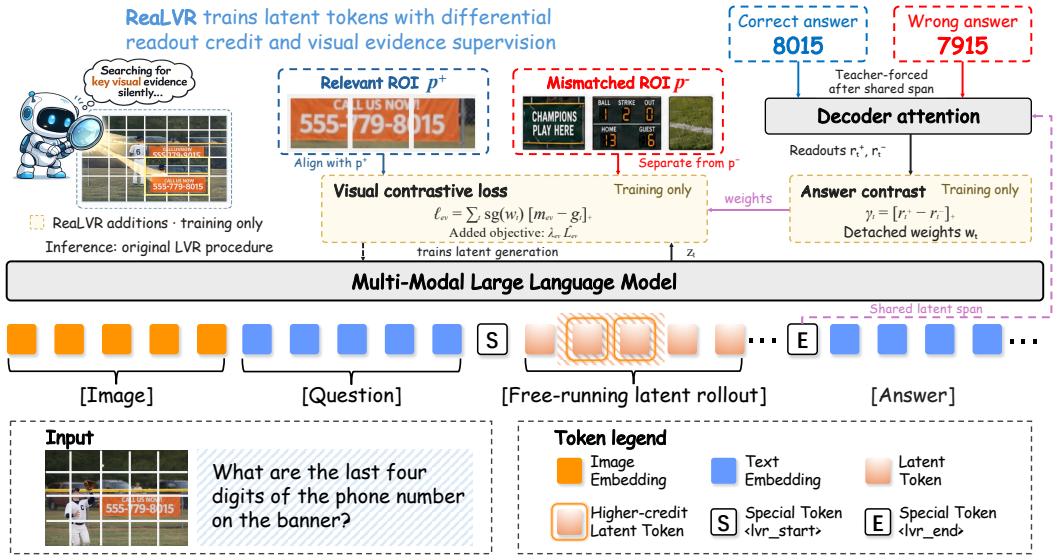


Figure 2: **ReaLVR learns visual grounding on its own latent trajectory.** The model first generates continuous latent states from the image and question. Relevant and mismatched visual prototypes specify what these states should preserve. Correct and wrong answers are separately teacher-forced after the shared latent span; their attention contrast determines the supervision weights. The weights are detached, and the visual loss trains the latent-generation process. Both supervision branches are used only during training; inference follows the original LVR procedure.

4.1 SUPERVISE THE MODEL'S OWN LATENT TRAJECTORY

Stage 1 teaches latent states under supplied visual prefixes, while inference requires the model to generate those prefixes itself. This distinction motivates the central principle of on-policy distillation: provide supervision on trajectories produced by the learner (Agarwal et al., 2024). ReaLVR applies this principle to visual grounding, using image evidence to supervise the current model's own latent computation. For each input x, the G behavior-policy completions from Section 3.2 supply the GRPO outcomes and candidate wrong answers. GRPO replays these completions with their saved latent inputs fixed. Alongside this policy update, ReaLVR regenerates one current-model trajectory by recursively feeding back its own hidden states:

$$z_t^\theta = \mathcal{T}_\theta\big(x, <|\texttt{lvr_start}|>, z_{1:t-1}^\theta\big), \qquad t = 1, \ldots, K.$$

We retain gradients through this recurrence, allowing the evidence loss to update the process that produces the latent states. The entire latent span is generated before any answer token is supplied. Correct and wrong answers are then teacher-forced in separate branches after this shared span to compute the supervision weights. Thus, one differentiable trajectory supports both visual alignment and answer-conditioned routing. Figure 2 illustrates these two sources of supervision.

4.2 SPECIFY WHAT TO PRESERVE WITH VISUAL CONTRAST

We anchor the generated states to visual evidence from the input image. The annotation a defines an evidence mask $a^+ = (a_1^+, \ldots, a_N^+) \in [0,1]^N$ over the visual tokens $\mathbf{V}$. The positive prototype is the masked mean, $p^+ = \mathrm{Pool}(\mathbf{V}; a^+) = \frac{\sum_n a_n^+ v_n}{\sum_n a_n^+ + \varepsilon}$, with $\varepsilon > 0$. If the ROI is unavailable or its mask is empty, we set $a_n^+ = 1$ and obtain a whole-image target. All latent positions share this visual target; their supervision strengths will be determined by answer contrast. To make the target discriminative, we form a set $\mathcal{N} = \{p_s^-\}_{s=1}^{N_-}$ of nonzero visual prototypes pooled from mismatched examples using the same construction, with $N_- \geq 1$. The margin at latent position t compares the relevant prototype with the most similar negative:

$$g_t = \mathrm{sim}(z_t^\theta, p^+) - \max_{p^- \in \mathcal{N}} \mathrm{sim}(z_t^\theta, p^-),$$

where sim is cosine similarity. Increasing this margin trains the latent state to distinguish the supporting visual content from competing image features. The vision encoder and connector are frozen, keeping the prototypes fixed as the language model learns to preserve their content.

4.3 LOCATE SUPERVISION WITH ANSWER CONTRAST

The model's own wrong answers provide a reference for identifying which latent positions are preferentially read under the correct answer. Subtracting this reference discounts attention shared across competing outcomes and concentrates supervision on positions with a stronger correct-answer readout. From the G behavior-policy completions, we construct $\mathcal{Y}_x^-$ by retaining distinct, parseable wrong answers with nonempty answer content. These answer candidates guide position selection, while the visual negatives in $\mathcal{N}$ define the content to distinguish.

For a canonical answer y, let $\mathrm{Fmt}(y) = (\widetilde{y}_1, \ldots, \widetilde{y}_{M(y)})$ be its formatted sequence, and let $\mathcal{J}(y)$ index the content tokens after excluding control and format markers. Each candidate is teacher-forced after $(x, \texttt{<|lvr_start|>}, z_{1:K}^\theta, \texttt{<|lvr_end|>})$. We extract attention at the input position of each content token $\widetilde{y}_j$: its query conditions on $\widetilde{y}_{1:j}$, including $\widetilde{y}_j$ itself. For a single-token multiple-choice answer, this is the query after `A` or `B` has been supplied as input.

Let $A_{j,t}^{(\ell,h)}(y)$ denote the resulting post-softmax attention to latent position t. Averaging over decoder layers $\ell \in \mathcal{L}_{\mathrm{dec}}$, heads $h \in \mathcal{H}$, and content positions $j \in \mathcal{J}(y)$ gives the readout $r_t(y) = \mathrm{mean}_{\ell,h,j} A_{j,t}^{(\ell,h)}(y)$. We retain the raw attention mass without renormalizing it within the latent span. All candidate branches use the current model and the same regenerated latent states; their role is to compute routing weights. Write $r_t^+ = r_t(y^\star)$ for the correct-answer readout and $r_t^- = |\mathcal{Y}_x^-|^{-1} \sum_{y^- \in \mathcal{Y}_x^-} r_t(y^-)$ for the mean wrong-answer readout. The selective credit is their positive difference, $\gamma_t = [r_t^+ - r_t^-]_+$, where $[u]_+ = \max(u, 0)$. When no valid wrong answer is available, we set $r_t^- = r_t^+$, giving $\gamma_t = 0$. We keep the magnitude of this contrast so that it expresses both the preferred positions and the strength of the routing signal.

4.4 TRAIN WITH READOUT-WEIGHTED VISUAL EVIDENCE

We combine selective credit with a uniform baseline, $w_t = \eta/K + (1-\eta)\gamma_t$, where $\eta \in [0,1]$ controls the baseline strength. The total weight is $\sum_t w_t = \eta + (1-\eta)\sum_t \gamma_t$: stronger answer contrast increases the evidence supervision assigned to the example. With no selective signal, the weights reduce to $w_t = \eta/K$, retaining uniform supervision whenever $\eta > 0$. Hyperparameters are listed in Appendix A. We detach w_t when optimizing the visual margin. This makes the weights allocate supervision while the gradient improves the evidence representation, preventing a shortcut through reducing the weight itself. Let sg denote stop-gradient and let $m_{\mathrm{ev}} \in (0, 2]$ be the target margin. The per-example evidence loss and the Stage 2 objective are

$$\ell_{\mathrm{ev}} = \sum_{t=1}^{K} \mathrm{sg}(w_t)[m_{\mathrm{ev}} - g_t]_+, \tag{3}$$

$$\mathcal{L}_{\mathrm{S2}}^{\mathrm{ReaLVR}}(\theta) = \mathcal{L}_{\mathrm{S2}}^{\mathrm{LVR}}(\theta) + \lambda_{\mathrm{ev}} \widehat{\mathcal{L}}_{\mathrm{ev}}(\theta), \tag{4}$$

where $\widehat{\mathcal{L}}_{\mathrm{ev}} = B^{-1} \sum_{b=1}^{B} \ell_{\mathrm{ev}}^{(b)}$ averages over a minibatch of B examples and $\lambda_{\mathrm{ev}} \geq 0$ controls the added objective. Gradients of Equation 3 flow through the visual margins and recurrent latent

generation, with answer strings, prototypes, and weights fixed. Equation 4 preserves GRPO rewards and advantages while training latent states on discriminative visual evidence. Appendices D and E detail the weight allocation and gradient decomposition. At inference, ReaLVR generates K latent tokens and decodes the answer with the original LVR architecture and procedure; both supervision branches are used only during training.

5 Experiments

5.1 Experimental Setup

We evaluate ReaLVR on five benchmarks of visual discrimination, spatial reasoning, and high-resolution perception: `MMVP` (Tong et al., 2024), `BLINK` (Fu et al., 2024), `HRBench-4K/8K` (Wang et al., 2025), and `MME-RealWorld` (Zhang et al., 2025). Six backbones span Qwen2.5-VL and Qwen3-VL (Bai et al., 2025b;a), InternVL3 (Zhu et al., 2025), and Gemma 3 (Team et al., 2025). We compare with Pixel Reasoner (Su et al., 2025), Vision-R1 (Huang et al., 2026a), LVR (Li et al., 2026a), ILVR (Dong et al., 2026), and Monet (Wang et al., 2026b). We report task accuracy and the unweighted five-benchmark mean when available. Training used 800 AMD MI250X GPUs (128 GB each), please see Appendix A for detailed settings. Appendix C covers evaluation protocols and aggregation; Appendix G defines the analysis estimators; Appendix H.3 gives benchmark cases.

5.2 Main Results

Table 1: **Comparison with SOTA methods.** Five-benchmark accuracy on `Qwen2.5-VL-7B`. We report mean ± standard deviation over three random seeds. Purple method cells mark non-latent baselines, green method cells mark latent-reasoning baselines.

Model	Training recipe		Visual reasoning		High-res. / real-world			**Average↑** (5 tasks)
	Stage	Method	MMVP↑	BLINK↑	HR-4K↑	HR-8K↑	MME-RW↑	
Qwen2.5-VL-7B	RL	Pixel Reasoner [NeurIPS'25]	66.8$_{\pm 0.4}$	53.3$_{\pm 0.3}$	69.8$_{\pm 0.3}$	64.1$_{\pm 0.3}$	49.7$_{\pm 0.2}$	60.7$_{\pm 0.2}$
	RL	Vision-R1 [ICLR'26]	51.5$_{\pm 0.5}$	52.7$_{\pm 0.4}$	62.9$_{\pm 0.4}$	58.6$_{\pm 0.4}$	44.2$_{\pm 0.3}$	54.0$_{\pm 0.3}$
	SFT	LVR-SFT [ICLR'26]	63.6$_{\pm 0.3}$	53.2$_{\pm 0.2}$	69.0$_{\pm 0.3}$	63.3$_{\pm 0.2}$	49.5$_{\pm 0.2}$	59.7$_{\pm 0.2}$
	RL	LVR-RL [ICLR'26]	64.2$_{\pm 0.4}$	53.6$_{\pm 0.3}$	69.6$_{\pm 0.3}$	64.4$_{\pm 0.3}$	50.1$_{\pm 0.3}$	60.4$_{\pm 0.2}$
	S1	ILVR-Stage1 [ACL'26]	68.1$_{\pm 0.3}$	55.4$_{\pm 0.3}$	70.2$_{\pm 0.3}$	66.0$_{\pm 0.3}$	49.1$_{\pm 0.2}$	61.8$_{\pm 0.2}$
	S2	ILVR-Stage2 [ACL'26]	69.4$_{\pm 0.4}$	**56.8**$_{\pm 0.3}$	71.0$_{\pm 0.2}$	**66.9**$_{\pm 0.3}$	50.3$_{\pm 0.3}$	62.9$_{\pm 0.2}$
	SFT	Monet-SFT [CVPR'26]	68.4$_{\pm 0.3}$	51.2$_{\pm 0.4}$	68.9$_{\pm 0.3}$	64.8$_{\pm 0.3}$	50.7$_{\pm 0.2}$	60.8$_{\pm 0.2}$
	RL	Monet-RL [CVPR'26]	69.9$_{\pm 0.4}$	52.4$_{\pm 0.4}$	71.3$_{\pm 0.3}$	66.0$_{\pm 0.3}$	51.5$_{\pm 0.2}$	62.2$_{\pm 0.3}$
	RL	**ReaLVR**	**72.0**$_{\pm 0.4}$	55.8$_{\pm 0.3}$	**71.8**$_{\pm 0.3}$	66.6$_{\pm 0.3}$	**52.2**$_{\pm 0.2}$	**63.7**$_{\pm 0.2}$

Table 2: **Applicability across model sizes and families.** The 235B evaluation covers MMVP, BLINK, and MME-RealWorld; a five-task average is reported only when all five scores are available.

Model	Training recipe		Visual reasoning		High-res. / real-world			**Average↑** (5 tasks)
	Stage	Method	MMVP↑	BLINK↑	HR-4K↑	HR-8K↑	MME-RW↑	
Across Qwen3-VL model sizes								
Qwen3-VL-8B	SFT	LVR-SFT [ICLR'26]	65.5	52.9	68.4	61.0	47.7	59.1
	RL	LVR-RL [ICLR'26]	66.9	54.2	69.9	62.2	49.7	60.6
	RL	Monet [CVPR'26]	67.4	55.1	68.5	61.9	48.4	60.3
	RL	**ReaLVR**	68.7	54.9	70.4	62.9	49.3	**61.2**
Qwen3-VL-30B	SFT	LVR-SFT [ICLR'26]	77.0	47.5	73.8	66.8	53.4	63.7
	RL	LVR-RL [ICLR'26]	77.6	47.8	73.9	67.2	53.9	64.1
	RL	**ReaLVR**	77.7	51.4	74.1	67.9	54.7	**65.2**
Qwen3-VL-235B	–	Direct	80.0	73.5	–	–	69.8	–
	SFT	LVR-SFT [ICLR'26]	80.9	74.2	–	–	70.4	–
	RL	**ReaLVR**	81.9	75.4	–	–	71.0	–
Across model families								
InternVL3-8B	SFT	LVR-SFT [ICLR'26]	71.0	46.1	60.0	53.6	39.0	53.9
	RL	LVR-RL [ICLR'26]	71.4	46.6	61.2	54.0	41.6	55.0
	RL	**ReaLVR**	72.3	52.4	64.4	55.4	45.9	**58.1**
Gemma-3-12B	SFT	LVR-SFT [ICLR'26]	58.3	32.9	36.1	34.6	28.1	38.0
	RL	LVR-RL [ICLR'26]	59.1	33.3	36.7	36.3	29.6	39.0
	RL	**ReaLVR**	60.0	40.6	37.4	38.0	31.8	**41.6**

ReaLVR reaches the highest five-task average on `Qwen2.5-VL-7B`, 63.7 (Table 1): +4.0 points over LVR-SFT, +3.3 over LVR-RL, +2.9 over Monet-SFT, +1.5 over Monet-RL, and +0.8 over

ILVR, the strongest competing latent-reasoning row by average accuracy. Gains over LVR-SFT cover all five tasks, led by MMVP (+8.4), HR-8K (+3.3), and HR-4K (+2.8), spanning subtle visual discrimination and high-resolution evidence. Across model sizes (Table 2), applying the same objective and inference procedure to Qwen3-VL-8B, Qwen3-VL-30B, and Qwen3-VL-235B-A22B (235B total parameters) yields five-task averages of 61.2 and 65.2 at 8B and 30B. At 235B, ReaLVR scores 81.9 on MMVP, 75.4 on BLINK, and 71.0 on MME-RealWorld. HRBench was not evaluated, so no five-task mean is reported. Across model families, ReaLVR yields five-task means of 58.1 on InternVL3-8B and 41.6 on Gemma-3-12B without architecture or inference changes. On these backbones, ReaLVR exceeds LVR-RL by 3.1 and 2.6 points in five-task mean accuracy, respectively. These families use different vision encoders and language backbones; Gemma uses a fixed 256-token, single-tile image representation. Together, these results show applicability across three model families and up to 235B total parameters. Appendix F.3 isolates the components, and Appendix F sweeps inference budgets $K = 0$–20; a short span captures much of the benefit, and $K = 8$ gives the highest mean accuracy. Appendix J.3 reports a further supervision-mass-matched 2×2 test of answer-based position allocation and positive-versus-negative visual evidence.

5.3 Mechanism Analysis: Variation, Grounding, and Use

Visual attention can link generated words to image regions (Xu et al., 2015). We examine visual-evidence readout into latent states and answer readout from them (Figure 3; Appendix I), using variation, grounding, and fixed-context replacement to test answer dependence.Figure 3 shows weak

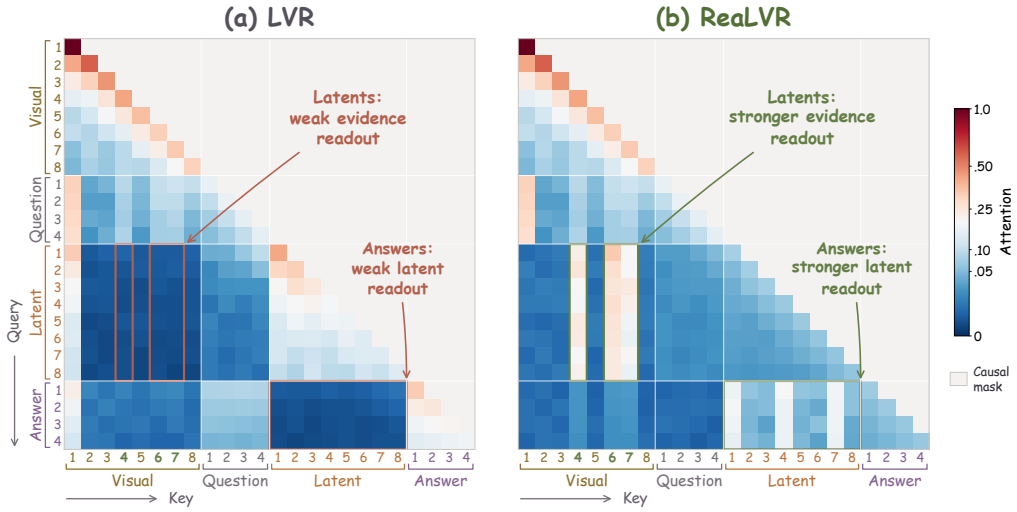


Figure 3: **Visual evidence and latent-token readout.** Attention of LVR (a) and ReaLVR (b) on Qwen2.5-VL-7B, averaged over all layers and heads and 200 HR-Bench-4K examples, with image, question, and answer tokens averaged into 8/4/4 bins and the $K = 8$ latent positions shown individually. The gray upper triangle is the causal mask: each query can attend only to its own and earlier positions (Vaswani et al., 2017). Each row sums to one over allowed keys. Green tick labels mark the same visual-evidence keys (4, 6, and 7), the image bins overlapping the annotated ROI. The narrow frames show latent queries reading these visual keys; the bottom frames show answer queries reading latent states. Panel (a) shows weak readout along both links; panel (b) shows stronger readout along both. Please see Appendix I for further attention analyses.

LVR attention from latent queries to target-overlapping visual bins and from answer queries to latent states; ReaLVR strengthens both links. The latter link motivates the dependence test below.

Vanilla LVR has weak counterfactual sensitivity. We test whether generated latent states respond to edits that change answer-relevant evidence. Each of four edit types contains 512 original–edited pairs with a fixed question. The correct answer changes in 81.45%–86.33% of pairs, but LVR changes its prediction in only 5.66%–13.09% (Table G.5). These edits preserve much of the scene while altering a decisive cue, as in counterfactual VQA evaluations (Agarwal et al., 2020; Dancette et al., 2021). Figure 4(c) illustrates the failure: LVR answers the original views correctly but retains its answers after changes to mug color, ball presence, object shape, or left–right relation. ReaLVR updates its answer in each case. Across the full sets, the gap between ground-truth and LVR prediction-change rates is 68.36–80.67 percentage points. In panel (b), ReaLVR's mean-pooled la-

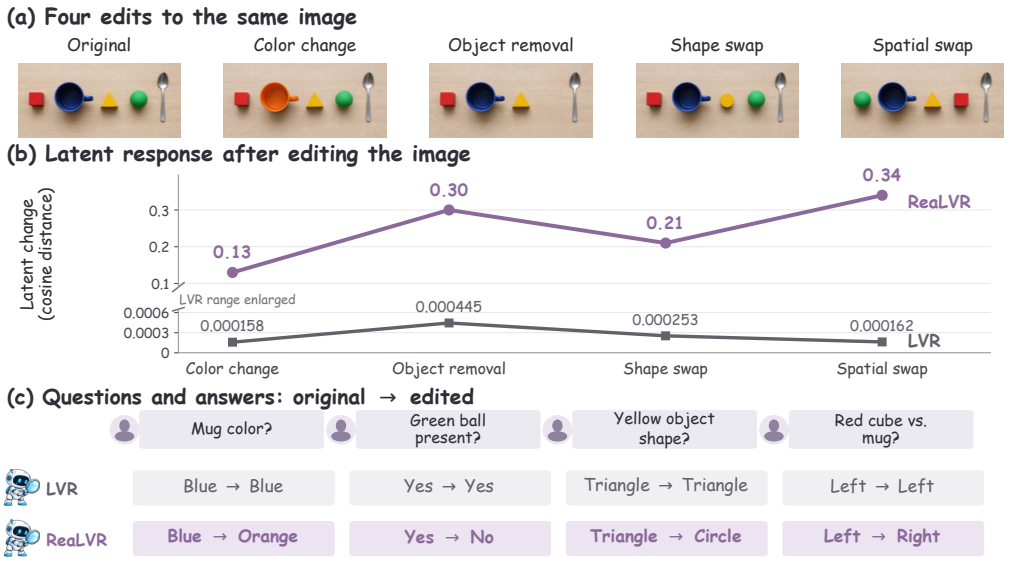


Figure 4: **Latent changes and answer updates after image edits.** (a) Four edits of the same reference image. (b) Cosine distance between the mean-pooled latent trajectories for the original and edited inputs; zero denotes no change. Each point is the average over 512 original–edited pairs per edit type with the question held fixed, measured for both LVR and ReaLVR on `Qwen2.5-VL-7B`. The broken vertical axis enlarges the LVR range to make its small fluctuations visible; values are unchanged. (c) Each user question is followed by the two model replies, read from the original image to the edited image. See Appendix H for more analysis.

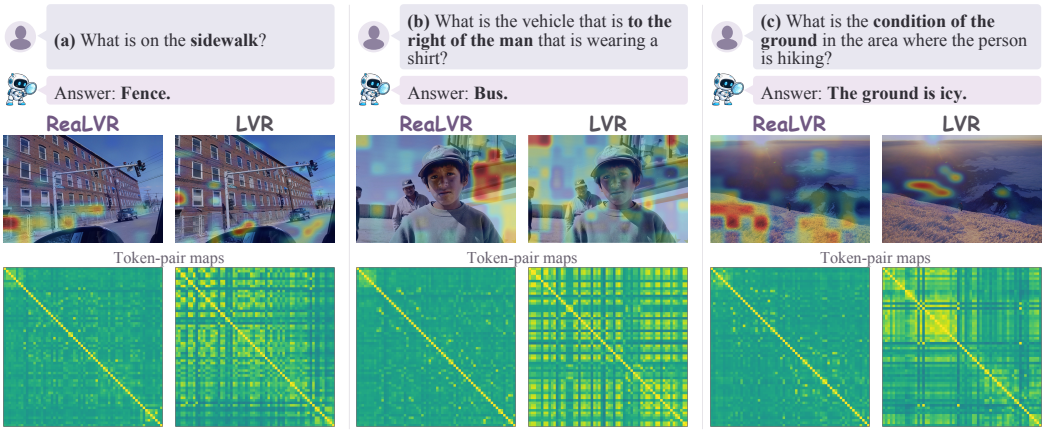


Figure 5: **Image and token views of the same three visual questions.** ReaLVR and LVR on the same fence, bus, and icy-ground questions: image saliency overlays (top) and token-pair maps (bottom). Each pair is labeled with its question and answer.

tent distance between original and edited inputs is 0.13–0.34, versus below 0.0005 for LVR. *LVR's trajectory may retain shared scene information while responding weakly to the cue needed to revise the answer.* Together with its weak free-running alignment to visual targets (Table G.3), this motivates supervising generated latent states on answer-relevant evidence. Appendices J.1 and J.2 report paired Direct/LVR/ReaLVR answer-change tests and latent interventions to separate correct answer revision from local dependence on the latent span.

Latent positions respond differently to input changes. Figure H.6 compares latent-position variation across image–question examples (a) and across questions about one fixed image (b). ReaLVR's variation is more concentrated at particular positions than Monet's or the LVR variants'. Panel (c) reports a top-token variation gap of 0.07 for ReaLVR, 0.02 for Monet, and 0.01 for each LVR variant. Mean pooling can obscure changes concentrated in a few latent states (ENNADIR et al., 2025). This pattern is consistent with ReaLVR's position-specific training signal: *positions share a visual*

target within an example, while correct-versus-wrong-answer readout assigns them different supervision weights. Figure 5 adds saliency views for the fence, bus, and icy-ground questions. ReaLVR focuses more on the relevant regions, whereas LVR's saliency is more diffuse. Because each overlay is normalized within its example, the comparison concerns spatial focus. Together, the variation map, saliency cases, and fixed-context replacement examine *how latent states change, where visual evidence is read, and whether selected states affect the answer.*

Useful latent computation need not be verbal. ReaLVR trains continuous states to preserve visual evidence, rather than produce intermediate sentences. We inspect their textual readout by applying the `Gemma-3-12B` vocabulary head to actual latent vectors from 47 questions with 8 states each, with special tokens masked. All 376 projections have < as their top-1 token with probability 1.0, while the same head gives correct next-token predictions for the answer "27B" in Figure 6. We attribute this readout to the closing-tag prior learned in Stage 1: < begins the literal `<|lvr_end|>` that follows every latent span, and the latent states align with its unembedding direction (cosine 0.41 versus 0.02 for a random vocabulary row). Top-1 vocabulary readout therefore does not expose a language rationale, yet the states are not empty: a linear probe recovers the BLINK task label from the mean latent with 99.9% accuracy (Appendix G.6). Continuous states can thus support answers without expressing a rationale (Hao et al., 2025).

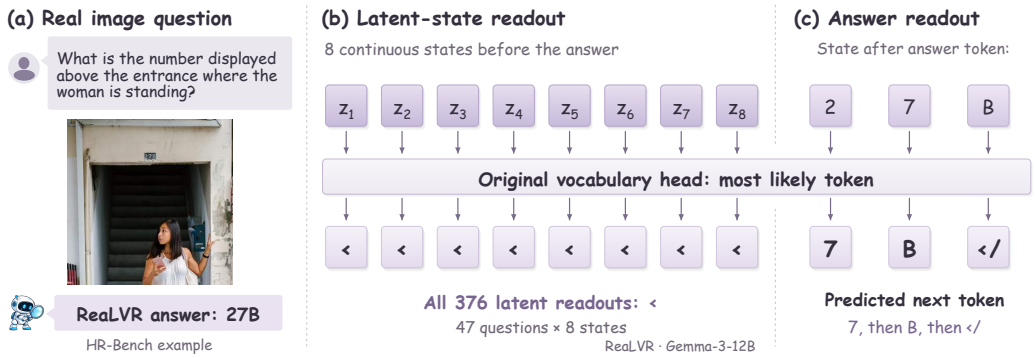


Figure 6: **A real visual answer and its latent-state text readout.** ReaLVR (`Gemma-3-12B`) answers "27B" from an HR-Bench image (a). The original vocabulary head reads each of its eight latent states as <, as it does all 376 states from 47 questions (b), yet predicts `7` after `2`, `B` after `7`, and `</` after `B` during answer generation (c). The latent readouts do not form an intermediate explanation.

A fixed-context replacement test probes whether the answer uses these states: replacing the eight most answer-attended tokens while holding other states fixed lowers ReaLVR's correct-answer probability from 0.70 to 0.59 (Figure G.1; Appendix G.1). Thus, nonverbal latent states affect answer likelihood locally; regenerating later states could introduce further effects.

ReaLVR attends more strongly to target regions. Target-region enrichment divides answer attention on an annotated region by that on same-area background windows, reducing region-size effects. A value of 1 means equal attention; 2 means twice as much on the target. ReaLVR reaches about 2 in middle layers, versus Monet near 1.6 and LVR at or below 1.3 (Figure G.2). Correct and incorrect ReaLVR responses have similar curves, so spatial alignment alone cannot explain correctness, consistent with prior VLM observations (Liu et al., 2026). Together with fixed-context replacement, this probes *where answers attend and whether selected latent states affect answer likelihood under controlled replacement with other states held fixed* (Reich et al., 2023).

6 Conclusion

In this work, we study a fundamental challenge in latent visual reasoning: a correct final answer does not guarantee that the preceding latent tokens have learned to preserve the visual evidence necessary to produce it. We identify this missing link as the *latent evidence-credit gap*. To address it, we introduce ReaLVR, which contrasts visual evidence to teach latent tokens *what* to preserve, while comparing how the correct and wrong answers attend to these tokens to decide *where* supervision is most critical. Across the tested backbones up to 235B, evidence supervision improves the available LVR baselines without changing model architectures or inference procedures. These results support visual-evidence supervision as a way to improve the grounding and use of continuous latent states.

AI USE STATEMENT

We used generative AI tools (ChatGPT and Claude Code) for implementing methods: parts of our training and evaluation code were drafted with AI coding assistants. No other required-disclosure task involved generative AI, and those concerning mathematical proofs and qualitative data analysis do not apply to this work. We additionally used AI tools to identify and summarize related literature and to edit the manuscript for readability. All AI-assisted code was read and tested by the authors, all AI-suggested references were retrieved and checked against the originals. We take responsibility for the full content of this work.

REPRODUCIBILITY STATEMENT

Section 4 specifies the ReaLVR objective, visual targets, and gradient flow. Appendix A documents the two-stage training recipe, hardware, distributed setup, and optimization hyperparameters. Appendix C gives the evaluated backbones and benchmarks, score aggregation, and rules for constructing visual prototypes and wrong-answer candidates. Appendix G defines the grounding, saliency, counterfactual, and latent-replacement diagnostics. Task-level results and ablations are reported in the main paper and appendices.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

Vedika Agarwal, Rakshith Shetty, and Mario Fritz. Towards causal vqa: Revealing and reducing spurious correlations by invariant and covariant semantic editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

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

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025b. URL https://arxiv.org/abs/2502.13923.

Chao Chen, Zhixin Ma, Yongqi Li, Yupeng Hu, Yinwei Wei, Wenjie Li, and Liqiang Nie. Reasoning in the dark: Interleaved vision-text reasoning in latent space. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 39117–39129, 2026a.

Xiuwei Chen, Wentao Hu, Yongxin Wang, Zisheng Chen, Likui Zhang, Kun Xiang, Jianhua Han, Hui-Ling Zhen, Jingyuan Zou, Hang Xu, and Xiaodan Liang. Latent visual states for efficient multimodal reasoning, 2026b. URL https://arxiv.org/abs/2606.24233.

Tao Cheng, Shi-Zhe Chen, Hao Zhang, Yixin Qin, Jinwen Luo, and Zheng Wei. Hybrid latent reasoning with decoupled policy optimization. *arXiv preprint arXiv:2604.20328*, 2026.

Jin Cui, Xinyue Long, Xunyong Zhang, Yadong Zhang, Chuanchang Su, Jingye Gan, Boran Zhao, and Pengju Ren. Retrieve, integrate, and synthesize: Spatial-semantic grounded latent visual reasoning, 2026. URL https://arxiv.org/abs/2605.07106.

Corentin Dancette, Rémi Cadène, Damien Teney, and Matthieu Cord. Beyond question-based biases: Assessing multimodal shortcut learning in visual question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1574–1583, October 2021.

Ziyang Ding, Linjian Meng, Yiming Wu, Yuhan Li, Yuhao Liu, and Zhen Zhao. Colvr: Enhancing exploratory latent visual reasoning via contrastive optimization, 2026. URL https://arxiv.org/abs/2605.08802.

Shuai Dong, Siyuan Wang, Xingyu Liu, Chenglin Li, Haowen Hou, and Zhongyu Wei. Interleaved latent visual reasoning with selective perceptual modeling. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 29316–29335, 2026.

Sofiane ENNADIR, Levente Zólyomi, Oleg Smirnov, Tianze Wang, John Pertoft, Filip Cornell, and Lele Cao. Pool me wisely: On the effect of pooling in transformer-based models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=8uhXfdSJmA.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. Blink: Multimodal large language models can see but not perceive. In *European Conference on Computer Vision*, pp. 148–166. Springer, 2024.

Jiawei Guo, Yu Chen, Xiang Wang, Shuai Li, Xinpei Zhao, Huaxing Liu, Shuai Dong, Feifei Zhai, and Yu Zhou. Beyond visual memory: Mechanistic diagnostics of latent visual reasoning. *arXiv preprint arXiv:2606.01287*, 2026.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason E Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=Itxz7S4Ip3.

Lianyu Hu, Shengqian Qin, Zeqin Liao, Qing Guo, Liang Wan, Wei Feng, and Yang Liu. Colt: Teaching multi-modal models to think with chain of latent thoughts. In *European Conference on Computer Vision*, pp. 526–544. Springer, 2026.

David Huang and Lianlei Shan. Dlwm: Diverse latent world models for efficient multimodal reasoning, 2026. URL https://arxiv.org/abs/2606.15160.

Wenxuan Huang, Bohan Jia, Shaosheng Cao, Zheyu Ye, Zhe Xu, Yao Hu, Shaohui Lin, et al. Vision-r1: Incentivizing reasoning capability in multimodal large language models. In *International Conference on Learning Representations*, volume 2026, pp. 63794–63812, 2026a.

Xinyu Huang, Yuhao Dong, Weiwei Tian, Bo Li, Rui Feng, and Ziwei Liu. Mgpo: Thinking with images via multi-turn grounding-based reinforcement learning. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 383–399, 2026b.

Byungwoo Jeon, Yoonwoo Jeong, Hyunseok Lee, Minsu Cho, and Jinwoo Shin. Vision-aligned latent reasoning for multi-modal large language model. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=qmvoTiXSfG.

Houcheng Jiang, Jiajun Fu, Junfeng Fang, Chen Gao, Xiang Wang, Xiangnan He, and Yong Li. Univlr: Unifying text and vision in visual latent reasoning for multimodal llms, 2026. URL https://arxiv.org/abs/2605.11856.

Jiaxuan Kang, Siyu Chen, Mingda Li, Mingjie Liu, Tianyue Wang, Zhaoyang Wei, Yongheng Zhang, Yanchao Hao, and Zheng Wei. Lut: Latent utility training for visual reasoning, 2026. URL https://arxiv.org/abs/2608.00743.

Bangzheng Li, Ximeng Sun, Jiang Liu, Ze Wang, Jialian Wu, Xiaodong Yu, Emad Barsoum, Muhao Chen, and Zicheng Liu. Latent visual reasoning. In *International Conference on Learning Representations*, volume 2026, pp. 148076–148090, 2026a.

Kelvin Li, Chuyi Shang, Leonid Karlinsky, Rogerio Feris, Trevor Darrell, and Roei Herzig. Latent implicit visual reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 33457–33466, 2026b.

Peiming Li, Yifan Wang, Xiaotian Zhang, Zhiyuan Hu, Shiyu Li, Zheng Wei, and Yang Tang. Prolavit: Learning progressive latent visual thoughts in structured latent space. In *European Conference on Computer Vision*, pp. 355–372. Springer, 2026c.

You Li, Chi Chen, Yanghao Li, Fanhu Zeng, Kaiyu Huang, Jinan Xu, and Maosong Sun. Imagination helps visual reasoning, but not yet in latent space. In *Forty-third International Conference on Machine Learning*, 2026d. URL https://openreview.net/forum?id=l1cMErXg1P.

Zhining Liu, Ziyi Chen, Hui Liu, Chen Luo, Xianfeng Tang, Suhang Wang, Jingying Zeng, Zhenwei Dai, Zhan Shi, Tianxin Wei, Hanqing Lu, Benoit Dumoulin, and Hanghang Tong. Seeing but not believing: Probing the disconnect between visual attention and answer correctness in VLMs. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=JAI7afWA9e.

Dongchen Lu, Zhimo Li, Mao Shu, and Huo Cao. Deeplatent: Think with images via parallel latent visual reasoning, 2026. URL https://arxiv.org/abs/2606.00562.

Yunze Man, De-An Huang, Guilin Liu, Shiwei Sheng, Shilong Liu, Liang-Yan Gui, Jan Kautz, Yu-Xiong Wang, and Zhiding Yu. Argus: Vision-centric reasoning with grounded chain-of-thought. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14268–14280. IEEE, 2025.

Yanting Miao, Yutao Sun, Dexin Wang, Mengyu Zhou, Pascal Poupart, Lei Lv, Qi Zhao, Li Wang, Hao Li, Xiaoxi Jiang, and Guanjun Jiang. Fill the gap: A granular alignment paradigm for visual reasoning in multimodal large language models, 2026. URL https://arxiv.org/abs/2605.12374.

Suhyeong Park, Junha Jung, and Jaewoo Kang. Reason through the latent! making latent visual reasoning necessary, 2026. URL https://arxiv.org/abs/2609.06746.

Daniel Reich, Felix Putze, and Tanja Schultz. Measuring faithful and plausible visual grounding in VQA. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023. URL https://openreview.net/forum?id=pvEkYbUPVW.

Hao Shao, Shengju Qian, Han Xiao, Guanglu Song, Zhuofan Zong, Letian Wang, Yu Liu, and Hongsheng Li. Visual cot: Advancing multi-modal language models with a comprehensive dataset and benchmark for chain-of-thought reasoning. *Advances in Neural Information Processing Systems*, 37:8612–8642, 2024a.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024b. URL https://arxiv.org/abs/2402.03300.

Alex Su, Haozhe Wang, Weiming Ren, Fangzhen Lin, and Wenhu Chen. Pixel reasoner: Incentivizing pixel space reasoning via curiosity-driven reinforcement learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=VeZkY3JjWV.

Yuesen Tang, Yiming Yang, Tengfei Bao, and Yu Tong. Thinking in latent space: Progressive multimodal simplification for visual reasoning. In *Forty-third International Conference on Machine Learning*.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, Gaël Liu, Francesco Visin, Kathleen Kenealy, Lucas Beyer, Xiaohai Zhai, Anton Tsitsulin, Robert Busa-Fekete, Alex Feng, Noveen Sachdeva, Benjamin Coleman, Yi Gao, Basil Mustafa, Iain Barr, Emilio Parisotto, David Tian, Matan Eyal, Colin Cherry, Jan-Thorsten Peter, Danila Sinopalnikov, Surya Bhupatiraju, Rishabh Agarwal, Mehran Kazemi, Dan Malkin, Ravin Kumar, David Vilar, Idan Brusilovsky, Jiaming Luo, Andreas Steiner, Abe Friesen, Abhanshu Sharma, Abheesht Sharma, Adi Mayrav Gilady, Adrian Goedeckemeyer, Alaa Saade, Alex Feng, Alexander Kolesnikov, Alexei Bendebury, Alvin Abdagic, Amit Vadi, András György, André Susano Pinto, Anil Das, Ankur Bapna, Antoine Miech, Antoine Yang, Antonia Paterson, Ashish Shenoy, Ayan Chakrabarti, Bilal Piot, Bo Wu, Bobak Shahriari, Bryce Petrini, Charlie Chen, Charline Le Lan, Christopher A. Choquette-Choo, CJ Carey, Cormac Brick, Daniel Deutsch, Danielle Eisenbud, Dee Cattle, Derek Cheng, Dimitris Paparas, Divyashree Shivakumar Sreepathihalli, Doug Reid, Dustin Tran, Dustin Zelle, Eric Noland, Erwin Huizenga, Eugene Kharitonov, Frederick Liu, Gagik Amirkhanyan, Glenn Cameron, Hadi Hashemi, Hanna Klimczak-Plucińska, Harman Singh, Harsh Mehta, Harshal Tushar Lehri, Hussein Hazimeh, Ian

Ballantyne, Idan Szpektor, Ivan Nardini, Jean Pouget-Abadie, Jetha Chan, Joe Stanton, John Wieting, Jonathan Lai, Jordi Orbay, Joseph Fernandez, Josh Newlan, Ju yeong Ji, Jyotinder Singh, Kat Black, Kathy Yu, Kevin Hui, Kiran Vodrahalli, Klaus Greff, Linhai Qiu, Marcella Valentine, Marina Coelho, Marvin Ritter, Matt Hoffman, Matthew Watson, Mayank Chaturvedi, Michael Moynihan, Min Ma, Nabila Babar, Natasha Noy, Nathan Byrd, Nick Roy, Nikola Momchev, Nilay Chauhan, Noveen Sachdeva, Oskar Bunyan, Pankil Botarda, Paul Caron, Paul Kishan Rubenstein, Phil Culliton, Philipp Schmid, Pier Giuseppe Sessa, Pingmei Xu, Piotr Stanczyk, Pouya Tafti, Rakesh Shivanna, Renjie Wu, Renke Pan, Reza Rokni, Rob Willoughby, Rohith Vallu, Ryan Mullins, Sammy Jerome, Sara Smoot, Sertan Girgin, Shariq Iqbal, Shashir Reddy, Shruti Sheth, Siim Põder, Sijal Bhatnagar, Sindhu Raghuram Panyam, Sivan Eiger, Susan Zhang, Tianqi Liu, Trevor Yacovone, Tyler Liechty, Uday Kalra, Utku Evci, Vedant Misra, Vincent Roseberry, Vlad Feinberg, Vlad Kolesnikov, Woohyun Han, Woosuk Kwon, Xi Chen, Yinlam Chow, Yuvein Zhu, Zichuan Wei, Zoltan Egyed, Victor Cotruta, Minh Giang, Phoebe Kirk, Anand Rao, Kat Black, Nabila Babar, Jessica Lo, Erica Moreira, Luiz Gustavo Martins, Omar Sanseviero, Lucas Gonzalez, Zach Gleicher, Tris Warkentin, Vahab Mirrokni, Evan Senter, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, Yossi Matias, D. Sculley, Slav Petrov, Noah Fiedel, Noam Shazeer, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Jean-Baptiste Alayrac, Rohan Anil, Dmitry, Lepikhin, Sebastian Borgeaud, Olivier Bachem, Armand Joulin, Alek Andreev, Cassidy Hardin, Robert Dadashi, and Léonard Hussenot. Gemma 3 technical report, 2025. URL https://arxiv.org/abs/2503.19786.

Jintao Tong, Jiaqi Gu, Yujing Lou, Lubin Fan, Yixiong Zou, Yue Wu, Jieping Ye, and Ruixuan Li. Sketch-in-latents: Eliciting unified reasoning in mllms, 2025. URL https://arxiv.org/abs/2512.16584.

Jintao Tong, Shilin Yan, Hongwei Xue, Xiaojun Tang, Kunyu Shi, Guannan Zhang, Ruixuan Li, and Yixiong Zou. Swimbird: Eliciting switchable reasoning mode in hybrid autoregressive mllms. *arXiv preprint arXiv:2602.06040*, 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 *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9568–9578. IEEE, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

André G Viveiros, Nuno Gonçalves, Matthias Lindemann, and André Martins. Lantern: Latent visual structured reasoning. *arXiv preprint arXiv:2603.25629*, 2026a.

André G. Viveiros, Nuno Gonçalves, André F. T. Martins, and Matthias Lindemann. What's holding back latent visual reasoning?, 2026b. URL https://arxiv.org/abs/2605.18445.

Chenfeng Wang, Wei He, Xuhan Zhu, Chunpeng Zhou, Qizhen Li, Song Yan, Yufei Zheng, Chengjun Yu, Fan Lu, Wei Zhai, Yang Cao, Pengfei Yu, and Zheng-Jun Zha. Self-consistent latent reasoning: Long latent sequence reasoning for vision-language model, 2026a. URL https://arxiv.org/abs/2605.12163.

Qixun Wang, Yang Shi, Yifei Wang, Yuanxing Zhang, Pengfei Wan, Kun Gai, Xianghua Ying, and Yisen Wang. Monet: Reasoning in latent visual space beyond image and language. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12030–12040, 2026b.

Wenbin Wang, Liang Ding, Minyan Zeng, Xiabin Zhou, Li Shen, Yong Luo, Wei Yu, and Dacheng Tao. Divide, conquer and combine: A training-free framework for high-resolution image perception in multimodal large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 7907–7915, 2025.

Yifan Wang, Shiyu Li, Peiming Li, Xiaochen Yang, Zheng Wei, and Yang Tang. Render-of-thought: Rendering textual chain-of-thought as images for visual latent reasoning. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 45236–45253, 2026c.

Yubo Wang, Juntian Zhang, Yichen Wu, Yankai Lin, Nils Lukas, and Yuhan Liu. Forest before trees: Latent superposition for efficient visual reasoning. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11272–11288, 2026d.

Zihu Wang, Karthik Somayaji N. S, and Peng Li. Regular: Relation-grounded latent reasoning for large vision-language models, 2026e. URL https://arxiv.org/abs/2605.30587.

Linquan Wu, Tianxiang Jiang, Yifei Dong, Haoyu Yang, Fengji Zhang, Shichang Meng, Ai Xuan, Linqi Song, and Jacky Keung. Lavit: Aligning latent visual thoughts for multi-modal reasoning. In *European Conference on Computer Vision*, pp. 345–363. Springer, 2026a.

Qiong Wu, Xiangcong Yang, Yiyi Zhou, Chenxin Fang, Baiyang Song, Xiaoshuai Sun, and Rongrong Ji. Grounded chain-of-thought for multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 33577–33587, 2026b.

Jiaer Xia, Bingkui Tong, Yuhang Zang, Rui Shao, and Kaiyang Zhou. Bootstrapping grounded chain-of-thought in multimodal llms for data-efficient model adaptation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 208–217. IEEE, 2025.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pp. 2048–2057. PMLR, 2015.

Tianrun Xu, Yue Sun, Qixun Wang, Jingyi Lu, Yuan Wang, Tianren Zhang, Longteng Guo, Fengyun Rao, Jing LYU, Feng Chen, and Jing Liu. Semantic-enriched latent visual reasoning. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=DTuBIEhSF3.

Zesheng Yang, Lingling Zhang, Xinyu Zhang, Cheng Zhang, Pengyu Li, Heng Wang, and Lin Wu. Glaq: Grounding latent queries in visual evidence for multimodal reasoning, 2026a. URL https://arxiv.org/abs/2608.15517.

Zeyuan Yang, Xueyang Yu, Delin Chen, Maohao Shen, and Chuang Gan. Machine mental imagery: Empower multimodal reasoning with latent visual tokens. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 33510–33520, 2026b.

Chi Zhang, Haibo Qiu, Qiming Zhang, Yufei Xu, Zhixiong Zeng, Siqi Yang, Peng Shi, Lin Ma, and Jing Zhang. Perceptual-evidence anchored reinforced learning for multimodal reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 41111–41120, 2026a.

Xin Zhang, Qiqi Tao, Jiawei Du, Moyun Liu, and Joey Tianyi Zhou. Visual latents know more than they say: Unsilencing latent reasoning in mllms, 2026b. URL https://arxiv.org/abs/2605.02735.

XiuYu Zhang, Junfeng Fang, and Zhenkai Liang. Cosine misleads: Auxiliary losses reshape vision language models, not their latents. *arXiv preprint arXiv:2606.05753*, 2026c.

YiFan Zhang, Huanyu Zhang, Haochen Tian, Chaoyou Fu, Shuangqing Zhang, Junfei Wu, Feng Li, Kun Wang, Qingsong Wen, Zhang Zhang, et al. Mme-realworld: Could your multimodal llm challenge high-resolution real-world scenarios that are difficult for humans? In *International Conference on Learning Representations*, volume 2025, pp. 89655–89701, 2025.

Ziwei Zheng, Minghao Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, and Chao Shen. Deepeyes: Incentivizing" thinking with images" via reinforcement learning. In *International Conference on Learning Representations*, volume 2026, pp. 126775–126798, 2026.

Dongyao Zhu, Zhen Wang, Xi Xiao, Han Jiang, Saeed Vahidian, Wei-Lun Chao, Tanya Berger-Wolf, Yu Su, Raju Vatsavai, and Jianyang Gu. Leveraging latent visual reasoning in silence, 2026a. URL https://arxiv.org/abs/2605.18641.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, Zhangwei Gao, Erfei Cui, Xuehui Wang, Yue Cao, Yangzhou Liu, Xingguang Wei, Hongjie Zhang, Haomin Wang, Weiye Xu, Hao Li, Jiahao Wang, Nianchen

Deng, Songze Li, Yinan He, Tan Jiang, Jiapeng Luo, Yi Wang, Conghui He, Botian Shi, Xingcheng Zhang, Wenqi Shao, Junjun He, Yingtong Xiong, Wenwen Qu, Peng Sun, Penglong Jiao, Han Lv, Lijun Wu, Kaipeng Zhang, Huipeng Deng, Jiaye Ge, Kai Chen, Limin Wang, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models, 2025. URL https://arxiv.org/abs/2504.10479.

Mengdan Zhu, Senhao Cheng, and Liang Zhao. Decompose, look, and reason: Reinforced latent reasoning for vlms. *arXiv preprint arXiv:2604.07518*, 2026b.

Technical Appendices

Table of Contents

A Training Details and Hyperparameters **17**

B The Latent Evidence-Credit Gap **18**

C Experimental and Training Details **19**
C.1 Benchmarks and Model Coverage 19
C.2 Evaluation Protocol and Metrics 19
C.3 Visual Prototypes 19
C.4 Wrong-Answer Construction 20
C.5 Dependencies and Gradient Flow 20

D Uniform Bootstrap and Unassigned Mass **20**
D.1 Selective Credit and Its Remainder 20
D.2 Uniform Allocation and Boundary Cases 20

E Detached-Credit Gradient Decomposition **20**
E.1 Local Derivative of the Undetached Objective 20
E.2 The Detached Update 21

F Latent-Length Ablation **21**
F.1 Evaluation Setup 21
F.2 Task-Dependent Budgets 21
F.3 Components Ablations 22

G Additional Diagnostic Results **23**
G.1 Fixed-Context Latent-Token Dependence 23
G.2 Target-Region Attention Enrichment 23
G.3 Answer-Conditioned Saliency 25
G.4 Similarity and Representation Geometry 25
G.5 Target Alignment and Answer Readout 26
G.6 Task Structure and Counterfactual Sensitivity 27

H Additional Qualitative Results **27**
H.1 Image Saliency and Token Views 28
H.2 Attention and Saliency Cases 28
H.3 Task Examples Across Five Benchmarks 33
H.4 Position-wise Latent Variation 34

I Attention Analysis Designs **35**
I.1 Visual Evidence and Answer Readout 35
I.2 Question-Dependent Evidence Selection 36
I.3 Latent Selection and Fixed-Context Intervention 37

J Paired Counterfactual and Mass-Matched Mechanism Tests **37**
J.1 Paired counterfactual behavior 37

J.2 Latent exchange and position selection 38
J.3 Mass-matched where-by-what ablation 38

A Training Details and Hyperparameters

All backbones are trained with the two-stage recipe of Section 3.1: Stage 1 (target-conditioned visual supervision) followed by Stage 2 (free-running outcome optimization with the ReaLVR evidence loss). The recipe and all optimization hyperparameters are shared across backbones; only the number of GPUs, and hence the global batch size, changes with model scale.

Hardware. All models are trained on a cluster of AMD Instinct MI250X accelerators. An MI250X is a dual-die package: it holds two Graphics Compute Dies (GCDs), each with its own 64 GB of HBM2e memory, and the ROCm runtime exposes every GCD as a separate device. Each node in our cluster contains four MI250X accelerators and therefore provides eight independently addressable GPUs with 64 GB each. Following common practice on MI250X systems, we refer to a GCD as a GPU when referring to ROCm-visible devices. The 235B model is trained on 200 nodes, i.e. 800 MI250X accelerators exposed as 1,600 GCDs; the 7B, 8B, and 30B backbones use 64 nodes (256 MI250X accelerators, 512 GCDs). World sizes and per-device batch counts below refer to GCDs.

Distributed training. We use DeepSpeed ZeRO-3 with full parameter, gradient, and optimizer-state partitioning across all GPUs and no CPU offload; communication is overlapped with computation. Table A.1 summarizes the configuration.

Table A.1: **Hardware and distributed training configuration.** Each MI250X accelerator contributes two ROCm-visible 64 GB GCDs.

Item	Value
Accelerator	AMD Instinct MI250X (2 GCDs per accelerator)
Per node	4 MI250X = 8 GPUs (GCDs), 64 GB HBM2e per GPU
Accelerators, 235B	200 nodes $\times$ 4 = 800 MI250X (1,600 GCDs)
Accelerators, $\leq$30B	64 nodes $\times$ 4 = 256 MI250X (512 GCDs)
Parallelism	DeepSpeed ZeRO-3 (parameters, gradients, optimizer states)
Offload	None
`overlap_comm`	True
`stage3_max_live_parameters`	1×10^9
Precision	bf16
Gradient checkpointing	Enabled
Attention kernel	PyTorch SDPA
Launcher	Multi-node DeepSpeed launcher (hostfile)

Stage 1: target-conditioned visual supervision. Stage 1 trains the language model on the next-token loss plus the visual reconstruction loss, with the vision encoder and the vision–language merger frozen. Latent states are fed back as continuous hidden states (Section 3.1); no separate latent projection head is used. Sequences are packed to 4,096 tokens, and each image contributes between 128 and 5,120 visual tokens depending on its resolution. Each GPU processes one packed sequence per step without gradient accumulation, so the global batch equals the number of GPUs.

Stage 2: free-running outcome optimization with evidence credit. Stage 2 optimizes the objective. For every prompt the behavior policy samples $G = 8$ completions at temperature 0.6, which supply both the GRPO advantages and the wrong-answer set $\mathcal{Y}_x^-$. We set the KL weight β to 0, so the Stage-1 model serves only as the initialization. Images are capped at 2,560 visual tokens ($\approx$2.0 MP). Training prompts are drawn from a mixture of ViRL39K and Visual-CoT. Stage 2 runs for 100 optimizer steps with a checkpoint every 25 steps. And all baselines such as LVR and Monet use the same data and supervision for training. The evidence loss uses $K = 8$ latent tokens, weight $\lambda_{\text{ev}} = 0.2$, margin $m_{\text{ev}} = 0.5$, uniform baseline $\eta = 0.3$, and $N^- = 16$ negative prototypes pooled from other examples in the same global batch; the same K is used at inference.

Table A.2: **Stage-1 hyperparameters.**

Hyperparameter	Value
Per-GPU batch $\times$ grad. accumulation	1×1
Global batch (sequences)	1,600 (235B) / 512 ($\leq$30B)
Optimizer	AdamW
Peak learning rate	1×10^{-5}
LR schedule / warmup ratio / weight decay	cosine / 0.03 / 0.1
Reconstruction loss	MSE, weight 0.1
Latent feedback	continuous hidden state
Latent projection head	none
Trainable modules	language model
Frozen modules	vision tower, merger
Sequence packing / max packed tokens	yes / 4,096
Max sequence length	4,096
Visual tokens per image	128–5,120

Table A.3: **Stage-2 hyperparameters (GRPO with ReaLVR evidence supervision).**

Hyperparameter	Value
Objective	GRPO + evidence loss
Group size G (rollouts per prompt)	8
Sampling temperature / top-p / top-k	0.6 / 1.0 / off
KL weight β	0
Per-GPU batch $\times$ grad. accumulation	1×1
Prompts per step	1,600 (235B) / 512 ($\leq$30B)
Optimizer	AdamW
Peak learning rate	5×10^{-7}
LR schedule / warmup ratio / weight decay	cosine / 0.03 / 0.1
Max prompt length	4,096
Max completion length	192
Max visual tokens per image	2,560
Training steps	100
Checkpoint interval	25 steps
Latent length K (training and inference)	8
Evidence loss weight λ_{ev}	0.2
Target margin m_{ev}	0.5
Uniform baseline η	0.3
Negative prototypes per example N^-	16 (pooled from other examples in the global batch)
Training data	ViRL39K + Visual-CoT mixture

B THE LATENT EVIDENCE-CREDIT GAP

Equation 2 assigns an outcome to an entire completion. The reward identifies a successful answer, while leaving open which latent tokens supported it and what visual evidence they preserved. Vanilla LVR supplies visual targets only to target-conditioned Stage 1 states, leaving the free-running states used in Stage 2 and at inference without direct visual-evidence supervision.

We separate three questions about a generated latent token. *Readout* asks whether the answer decoder attends to it. *Grounding* asks whether it represents the evidence relevant to the image–question pair rather than mismatched evidence. *Utility* asks whether intervening on the token changes the answer. These questions require distinct measurements: a token can receive attention without representing relevant evidence or affecting the answer. Readout shared by correct and behavior-policy wrong answers can also reflect formatting or transition behavior common to both outcomes.

ReaLVR addresses the training gap with outcome-contrastive readout credit. It supervises one regenerated current-model trajectory with visual evidence and gives greater weight to positions read more strongly under the correct answer than under sampled wrong answers. Both readouts use the current model and the same regenerated latent span. Intervention utility remains a separate evaluation criterion.

Figure B.1 separates the roles of outcome reward, visual evidence supervision, and intervention-based evaluation.

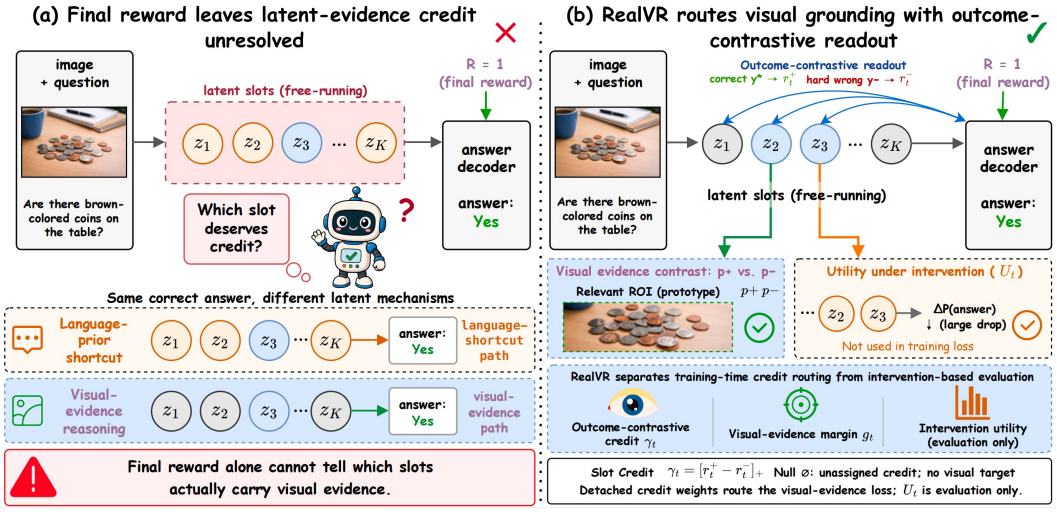


Figure B.1: **From final reward to latent evidence credit.** The output reward indicates whether the answer is correct. ReaLVR uses relevant and mismatched visual evidence to determine what the latent tokens should preserve, then contrasts ground-truth and wrong-answer readouts to determine where supervision should be stronger. Intervention separately evaluates whether the answer depends on the latent tokens and is not part of the training objective.

C Experimental and Training Details

C.1 Benchmarks and Model Coverage

`MMVP` (Tong et al., 2024) tests subtle visual discrimination, while `BLINK` (Fu et al., 2024) evaluates visual perception tasks, including counting, jigsaw, spatial relations, and depth. `HRBench-4K` and `HRBench-8K` (Wang et al., 2025) test high-resolution perception, and `MME-RealWorld` (Zhang et al., 2025) tests understanding of complex real-world scenes. Together, these benchmarks cover visual reasoning at different image resolutions.

Five backbones have complete five-benchmark results: `Qwen2.5-VL-7B` (Bai et al., 2025b), `Qwen3-VL-8B` and `Qwen3-VL-30B` (Bai et al., 2025a), `InternVL3-8B` (Zhu et al., 2025), and `Gemma-3-12B` (Team et al., 2025). Here, `Qwen3-VL-30B` abbreviates `Qwen3-VL-30B-A3B`. We additionally evaluate `Qwen3-VL-235B-A22B`, abbreviated as `Qwen3-VL-235B`, on `MMVP`, `BLINK`, and `MME-RealWorld`, comparing direct decoding, LVR-SFT, and ReaLVR.

C.2 Evaluation Protocol and Metrics

Table 1 compares Pixel Reasoner (Su et al., 2025), Vision-R1 (Huang et al., 2026a), LVR (Li et al., 2026a), ILVR (Dong et al., 2026), and Monet (Wang et al., 2026b) at the training stages indicated in each row. Table 2 reports the Qwen3-VL size series and the InternVL3 and Gemma-3 family comparisons. The latter use a matched evaluation protocol within each family.

We report task accuracy and compute an unweighted mean only for rows with all five benchmark scores. Average gains are differences between the displayed one-decimal means. The three-benchmark 235B evaluation is reported task by task and has no five-benchmark average. Mechanism analyses examine latent variation, visual grounding, and answer dependence; Appendix G defines the region, saliency, and fixed-context token-replacement estimators.

The training-time construction rules below complete the derivation in Section 4, using the notation of Section 3.

C.3 Visual Prototypes

If the ROI annotation is absent or its visual-token mask is empty, we set $a_n^+ = 1$ for all n, giving a whole-image target p^+. We stabilize the pooling denominator and require nonzero prototypes

for the cosine margin. The negative set $\mathcal{N}$ contains at least one valid prototype; each p_s^- uses the same pooling rule on a designated mismatched example. Frozen vision and connector weights keep prototypes fixed while the evidence loss updates the language model through the regenerated trajectory.

C.4 Wrong-Answer Construction

We require $\mathcal{J}(y^\star) \neq \emptyset$. After parsing and canonicalization, the retained set is

$$\mathcal{Y}_x^- = \mathrm{Unique}\{\widehat{y}_i : i \in \{1, \dots, G\}, \widehat{y}_i \neq \perp, \mathrm{Correct}(\widehat{y}_i, y^\star) = 0, \mathcal{J}(\widehat{y}_i) \neq \emptyset\}\,.$$

This rule removes parse failures, correct answers, candidates without answer-content positions, and duplicates. The retained strings come from the behavior policy and supply comparison outcomes, regardless of their probability under the updated model.

Every formatted candidate is teacher-forced after the same regenerated latent span. The readout in Section 4.3 averages the selected decoder layers, heads, and answer-content positions.

C.5 Dependencies and Gradient Flow

For minibatch example b, the group outputs $\{o_{b,i}\}_{i=1}^G$ construct $\mathcal{Y}_{x_b}^-$. The loss also depends on $(x_b, a_b, y_b^\star)$ and the supplied visual negative set $\mathcal{N}_b$; we leave these dependencies implicit in $w_{b,t}$ and $g_{b,t}$. Gradients pass through the autoregressive latent-generation process, so a loss term at position t can update earlier generation steps and shared parameters. A token's weight specifies its contribution to the loss, not an update restricted to that position.

D Uniform Bootstrap and Unassigned Mass

D.1 Selective Credit and Its Remainder

Section 4.3 defines $\gamma_t = [r_t^+ - r_t^-]_+$. Because $\gamma_t \leq r_t^+$ and the raw attention mass over the latent span is at most one, $\sum_t \gamma_t \leq 1$. We therefore define the nonnegative bookkeeping remainder

$$\gamma_\emptyset = 1 - \sum_{t=1}^{K} \gamma_t.$$

Here $\emptyset$ labels unassigned mass; it is distinct from $a = \emptyset$, the notation for a missing ROI annotation.

D.2 Uniform Allocation and Boundary Cases

Section 4.4 mixes selective credit with a uniform component. The full allocation is

$$w_t = \frac{\eta}{K} + (1 - \eta)\gamma_t, \qquad w_\emptyset = (1 - \eta)\gamma_\emptyset.$$

For $0 < \eta < 1$, the first term gives every latent token a nonzero routing weight, while the second term preserves selective routing; $\eta = 0$ and $\eta = 1$ recover the selective-only and uniform-only endpoints. Because $\gamma_\emptyset + \sum_t \gamma_t = 1$, the complete allocation satisfies $w_\emptyset + \sum_t w_t = 1$. The remainder $w_\emptyset$ is bookkeeping only: it introduces no token, module, or loss term, and the token weights retain their original mass without renormalization.

If $\mathcal{Y}_x^- = \emptyset$, then $\gamma_t = 0$ and $w_t = \eta/K$. Thus, $\eta > 0$ retains uniform supervision when no valid wrong-answer comparison is available, while $\eta = 0$ assigns zero evidence weight to that example.

E Detached-Credit Gradient Decomposition

Detaching the readout-derived weights lets them allocate supervision while the evidence gradient improves the visual margin. The following decomposition shows which gradient path detachment removes.

E.1 Local Derivative of the Undetached Objective

At a given optimization update, we condition on the sampled candidate-answer set, its tokenized answer sequences, the selected decoder layers and heads, the answer-position masks, the visual prototypes, and η. These quantities are fixed for the local gradient calculation; the dependence on

θ below comes from the regenerated trajectory and its current-model readout. For one regenerated trajectory, define the per-token evidence violation

$$h_t(\theta) = [m_{\text{ev}} - g_t(\theta)]_+,$$

and temporarily view $w_t(\theta) = \eta/K + (1-\eta)\gamma_t(\theta)$ as an ordinary differentiable function. The corresponding hypothetical undetached objective is

$$\ell_{\text{ev}}^{\text{undet}}(\theta) = \sum_{t=1}^{K} w_t(\theta) h_t(\theta).$$

Away from the kink points of the positive-part operators, the product rule gives

$$\nabla_\theta \ell_{\text{ev}}^{\text{undet}} = \sum_{t=1}^{K} \underbrace{w_t \nabla_\theta h_t}_{\text{update the visual evidence margin}} + \sum_{t=1}^{K} \underbrace{h_t \nabla_\theta w_t}_{\text{update the credit router}} .$$

At a hinge or ReLU kink, or when multiple visual negatives attain the same maximum, automatic differentiation selects a subgradient and the same two computational-graph paths remain. The first term is the intended weighted evidence update. The second changes the router in proportion to the current evidence violation. In particular, $\partial \ell_{\text{ev}}^{\text{undet}} / \partial w_t = h_t \geq 0$: when the hinge is active, gradient descent has a local path to reduce the loss by lowering the token's weight. More explicitly, at smooth points,

$$\nabla_\theta h_t = -\mathbf{1}\{g_t < m_{\text{ev}}\} \nabla_\theta g_t, \qquad \nabla_\theta w_t = (1-\eta)\mathbf{1}\{r_t^+ > r_t^-\}(\nabla_\theta r_t^+ - \nabla_\theta r_t^-).$$

Thus, the routing term can lower r_t^+ or raise r_t^- where the readout difference is active, reducing the mass assigned to the token and increasing the bookkeeping remainder without improving the token's visual evidence margin. The decomposition therefore identifies a local optimization shortcut through the router.

E.2 The Detached Update

The stop-gradient operator preserves the forward value, $\text{sg}(w_t) = w_t$, but sets its derivative to zero. At optimization step k, it is equivalent for this branch to differentiating the local surrogate

$$\widetilde{\ell}_{\text{ev},k}(\theta) = \sum_{t=1}^{K} w_t(\theta_k) h_t(\theta).$$

Its gradient at the current parameters is

$$\nabla_\theta \widetilde{\ell}_{\text{ev},k}(\theta)\Big|_{\theta=\theta_k} = \sum_{t=1}^{K} w_t(\theta_k)\ \nabla_\theta h_t(\theta)|_{\theta=\theta_k}\,.$$

The weights are recomputed at each update, so this surrogate describes the local gradient at step k. The evidence branch uses the current readout to allocate supervision and improves the visual margin through the latent trajectory. Readout still evolves across updates through GRPO, KL regularization, and shared-parameter changes. Detachment removes only the evidence loss's direct gradient through the routing weights.

F Latent-Length Ablation

F.1 Evaluation Setup

We hold the `Qwen2.5-VL-7B` ReaLVR checkpoint fixed within this ablation and vary the prescribed inference-time latent length K. The $K = 0$ setting skips the latent span and decodes the answer directly.

F.2 Task-Dependent Budgets

Additional steps help different tasks to different degrees. Table F.1 shows that the best observed length is task dependent. `HR-8K` first reaches its maximum at $K = 8$, while `BLINK` peaks at $K = 16$, improving by 6.3 points over $K = 0$. In contrast, `MME-RealWorld` peaks at $K = 4$

Table F.1: **Sensitivity to the inference-time latent budget.** The ReaLVR (`Qwen2.5-VL-7B`) checkpoint is held fixed while the latent-token budget K is varied at inference; $K = 0$ decodes the answer without a latent span. $K = 8$ is the budget used during training and serves as the reference (†): entries are differences in accuracy (percentage points) from that column, which is therefore 0.0 by construction. The sweep is run on a fixed evaluation subset so that all budgets are scored identically; its absolute scores are consequently not on the same scale as Table 1, and the differences reported here are meaningful only within this sweep. Under the full evaluation protocol of Table 1, the $K = 8$ setting scores 72.0 on `MMVP`, 55.8 on `BLINK`, 66.6 on `HR-8K` and 52.2 on `MME-RealWorld`. Bold marks each row's largest value, including ties. The mean is the unweighted average across the four listed benchmarks.

Benchmark	**Latent-token budget** K *(difference from* $K = 8$*)*						
	0	2	4	$8^{\dagger}$	12	16	20
`HR-8K`	−3.0	−2.0	−1.5	**0.0**	−1.0	−0.5	**0.0**
`MMVP`	−0.7	−1.0	−0.3	**0.0**	−1.0	−1.0	**0.0**
`BLINK`	−4.0	0.0	−1.1	0.0	+1.2	**+2.3**	+0.9
`MME-RealWorld`	−4.4	+1.9	**+2.1**	0.0	−0.2	−1.4	−2.1
Mean	−3.0	−0.3	−0.2	**0.0**	−0.2	−0.1	−0.3

and then declines by 4.2 points by $K = 20$. `MMVP` varies by only 1.0 point across all tested lengths and shows no consistent trend. Together, these results favor task-dependent budgets over uniformly longer trajectories.

A short latent span captures much of the benefit. Across the four tasks in this ablation, $K = 8$ gives the highest mean accuracy; the closest alternative, $K = 16$, is 0.1 points below it, and removing the latent span entirely ($K = 0$) costs 3.0 points. Selecting the best tested K separately for each benchmark gives a post-hoc task-level oracle 1.1 points above $K = 8$, which motivates adaptive latent budgets. Even $K = 2$ captures 4.0 of the 6.3-point maximum gain on `BLINK` and 6.3 of the 6.5-point maximum gain on `MME-RealWorld`.

F.3 Components Ablations

Table F.2 removes each ingredient of ReaLVR on `Qwen2.5-VL-7B`. Every variant stays above LVR-RL, so the evidence loss helps in any form, but the full method is best on all five benchmarks.

Answer-contrast routing matters. Replacing the routing weights with a uniform $1/K$ ($\eta = 1$) costs 1.3 points on average and 2.5 on `MMVP`: the same visual supervision, spread evenly over the latent span, is markedly less effective than supervision concentrated where the correct answer reads. Subtracting the wrong-answer readout is part of this effect; routing with raw correct-answer attention alone recovers only 62.9, because attention shared by correct and wrong answers (formatting, transitions) then receives credit. Removing the uniform floor entirely ($\eta = 0$) is slightly worse than the full model (63.2 vs. 63.7), consistent with its role of keeping supervision alive on examples without a valid wrong answer.

Negatives make the target discriminative. Aligning latents to p^{+} without mismatched prototypes drops 1.6 points, the largest loss among the "what" and "where" components; a plain alignment objective pulls latents toward generic image content rather than toward what distinguishes this image from others.

On-policy regeneration and detachment are both necessary. Supervising the saved rollout latents instead of a regenerated trajectory loses 1.8 points, the largest drop in the table, confirming that the loss must reach the process that produces the latents at inference. Removing the stop-gradient on w_t loses 1.1 points, matching the shortcut identified in Appendix E: the router lowers weights on hard positions instead of improving their evidence margin.

Table F.2: **Component ablation on `Qwen2.5-VL-7B`.** Each row removes or replaces one ingredient of ReaLVR; all other settings follow Appendix A. "Uniform routing" sets $\eta = 1$ so every latent position receives weight $1/K$ and the answer contrast is unused. "Selective only" sets $\eta = 0$. "No negatives" replaces the margin with a plain cosine alignment to p^+. "Raw attention" routes with r_t^+ instead of $[r_t^+ - r_t^-]_+$. "Undetached" removes the stop-gradient on w_t. "Off-policy" applies the evidence loss to the saved rollout latents instead of a regenerated trajectory. LVR-RL is the $\lambda_{\text{ev}} = 0$ endpoint.

Variant	MMVP	BLINK	HR-4K	HR-8K	MME-RW	**Avg.**
LVR-RL (no evidence loss)	64.2	53.6	69.6	64.4	50.1	60.4
Where to supervise (answer contrast)						
Uniform routing ($\eta = 1$)	69.5	54.6	71.0	65.6	51.3	62.4
Raw attention ($\gamma_t = r_t^+$)	70.6	55.1	71.3	66.0	51.7	62.9
Selective only ($\eta = 0$)	71.2	55.3	71.4	66.2	51.9	63.2
What to preserve (visual contrast)						
No negatives ($\mathcal{N} = \emptyset$)	69.0	54.4	70.6	65.3	51.0	62.1
Training mechanics						
Undetached weights	70.1	54.8	71.1	65.5	51.4	62.6
Off-policy targets	68.8	54.2	70.4	65.1	50.8	61.9
ReaLVR (full)	72.0	55.8	71.8	66.6	52.2	**63.7**

G Additional Diagnostic Results

The diagnostics examine answer dependence, visual grounding, representation geometry, and sensitivity to image edits. Each subsection defines the quantity being measured before presenting its results.

G.1 Fixed-Context Latent-Token Dependence

Readout attention ranks latent tokens, but it does not show whether the answer depends on them. The fixed-context audit replaces one latent token while holding the others fixed and measures the change in target-answer log-likelihood. Let $o^\star_{\text{ans}}$ be the canonical formatted target answer, whose probability is computed by teacher forcing over the entire answer sequence, and let $\bar{z}_t$ be a replacement token vector. Write $\tilde{z}_{1:K}^{(t\leftarrow\bar{z}_t)}$ for the latent sequence obtained by replacing z_t with $\bar{z}_t$. The fixed-context intervention score is

$$u_t(\bar{z}_t) = \log \pi_\theta(o^\star_{\text{ans}} \mid x, z_{1:K}) - \log \pi_\theta\Big(o^\star_{\text{ans}} \mid x, \tilde{z}_{1:K}^{(t\leftarrow\bar{z}_t)}\Big). \tag{G.1}$$

A positive value means that the original latent token gives the target answer higher likelihood than its replacement. This is an evaluation metric, not part of the ReaLVR training loss. Figure G.1 reports the fixed-context intervention.

Answer-read latent tokens become more load-bearing. We rank latent tokens by answer-to-token attention, replace the top-k tokens while holding the remaining context fixed, and measure the correct-answer probability. Figure G.1 shows the resulting probability changes. For ReaLVR, the correct-answer probability falls monotonically from 0.70 at $k = 0$ to 0.59 at $k = 8$, a drop of 0.11 that exceeds those of Monet and both LVR variants. Together with ReaLVR's higher five-benchmark average, this result is consistent with the model placing more answer-relevant computation in the latent tokens that the answer subsequently reads.

This drop quantifies local answer dependence on the selected latent tokens.

G.2 Target-Region Attention Enrichment

For generated answer positions $\mathcal{J}$, heads $\mathcal{H}$, and image token indices $\mathcal{R} \subseteq \{1, \ldots, N\}$, let $A_{j,n}^{(\ell,h)}$ denote the post-softmax attention from answer position j to image token n at layer ℓ and head h. We aggregate this attention as

$$s_\ell(\mathcal{R}) = \frac{1}{|\mathcal{J}|\,|\mathcal{H}|} \sum_{j\in\mathcal{J}} \sum_{h\in\mathcal{H}} \sum_{n\in\mathcal{R}} A_{j,n}^{(\ell,h)}.$$

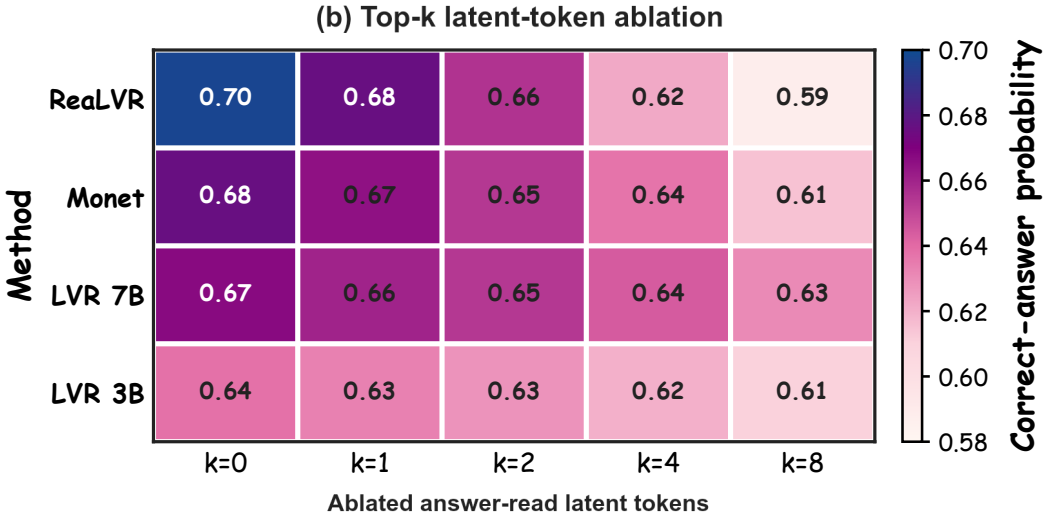


Figure G.1: **Fixed-context latent-token dependence.** Each cell gives the correct-answer probability after replacing the top-k answer-read latent tokens, with all remaining latent states held fixed. Columns increase k from 0 to 8. ReaLVR's probability falls from 0.70 to 0.59, the largest endpoint drop among the four methods.

Let $\mathcal{M}$ be the annotated target region and let $\Omega(\mathcal{M})$ contain same-area background windows outside it. We report

$$\rho_\ell = \frac{s_\ell(\mathcal{M})}{\mathbb{E}_{\mathcal{B}\sim\Omega(\mathcal{M})}[s_\ell(\mathcal{B})] + \epsilon}.$$

Here $\epsilon > 0$ stabilizes the denominator, and $\rho_\ell = 1$ indicates no enrichment.

Layer-wise target-region attention results. We next ask whether the answer attends to the visual region needed by the question. The diagnostic compares answer-to-image attention on the annotated target with same-area background windows, where a ratio of 1 denotes no enrichment. In Figure G.2, ReaLVR rises from near 1 in the lower layers to approximately 2 in the middle layers, then retains substantial enrichment through most upper layers. Monet peaks near 1.6, while the LVR curves remain at or below approximately 1.3. The correctly and incorrectly answered curves nevertheless track each other closely, and the incorrect curve is sometimes higher. Thus, ReaLVR strengthens spatial alignment, but looking at the target is not sufficient for answering correctly. This distinction motivates separating a scalable grounding signal from the answer-level learning signal.

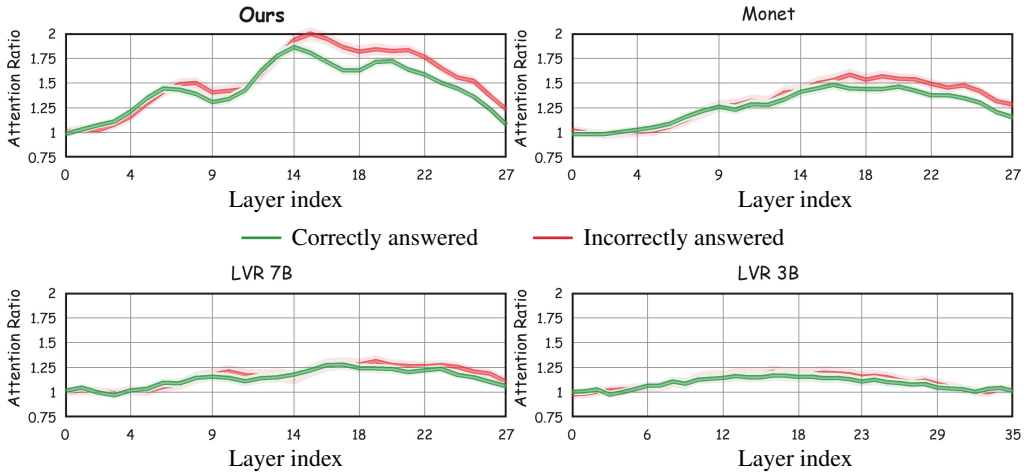


Figure G.2: **Target-region attention enrichment.** Layer-wise answer attention on the annotated region relative to matched background windows. Values above 1 indicate target-region enrichment; green and red curves correspond to correct and incorrect generations, respectively. Their overlap shows that alignment alone does not determine answer correctness.

G.3 Answer-Conditioned Saliency

After generating an answer $o^{\text{attr}} = (o_1^{\text{attr}}, \ldots, o_T^{\text{attr}})$, we run a teacher-forced attribution pass on its content positions $\mathcal{J}$. The cross-entropy target is that generated sequence, aggregated as

$$\mathcal{L}_{\text{CE}}(o^{\text{attr}}) = -\sum_{j \in \mathcal{J}} \log \pi_\theta\big(o_j^{\text{attr}} \mid x, z_{1:K}, o_{<j}^{\text{attr}}\big)\,.$$

The score for image token n is

$$S(n) = \frac{1}{|\mathcal{L}_{\text{dec}}|\,|\mathcal{H}|\,|\mathcal{J}|} \sum_{\ell \in \mathcal{L}_{\text{dec}}} \sum_{h \in \mathcal{H}} \sum_{j \in \mathcal{J}} \left| A_{j,n}^{(\ell,h)} \frac{\partial \mathcal{L}_{\text{CE}}(o^{\text{attr}})}{\partial A_{j,n}^{(\ell,h)}} \right|.$$

We resize the visual-token grid to the image and normalize the scores within each example. For panels explicitly labeled Saliency in Figure H.3 and the additional cases, token-level attribution retains the query-key matrix instead of projecting it onto image patches: rows index queries and columns index keys. Each case shows ReaLVR attention, LVR-7B attention, ReaLVR saliency, and LVR-7B saliency, in that order. Color intensity is normalized within each panel.

G.4 Similarity and Representation Geometry

High cosine similarity can reflect several properties of a representation. Table G.1 measures similarity within LVR trajectories and compares latent tokens with same-prefix dummy tokens and ordinary continuation tokens under different normalizations. Table G.2 then measures stability under perturbations and effective rank. The latent trajectories remain stable and highly similar while occupying fewer principal directions than the visual embeddings. These measurements describe representation geometry; similarity alone does not establish useful visual computation.

Table G.1: **Latent-token similarity diagnostics.** (a) Raw within-trajectory cosine similarity on two VISCOT examples. (b) Context-normalized similarity on BLINK (N=697), with same-prefix dummy and ordinary continuation tokens as references.

(a) Raw within-trajectory similarity

Example	Latent tokens	Off-diag. cos.	Adjacent cos.	Max cos.
VISCOT 18	126	0.7342	0.9417	0.9951
VISCOT 23	63	0.8408	0.9215	0.9958

(b) Context-normalized similarity

Comparison	LVR	Dummy	Ordinary	LVR–Ord.
Raw adjacent cosine	0.8877	0.9923	0.4817	+0.4059
Centered adjacent cosine	0.8166	0.9978	0.5210	+0.2956
Whitened adjacent cosine	0.6074	0.6481	0.0952	+0.5122
Residualized LVR adjacent cosine	0.7760	–	–	–
Same-task LVR mean cosine	0.9960	–	–	–
Different-task LVR mean cosine	0.9907	–	–	–

Table G.2: **Representation stability and effective rank.** (a) Representation consistency (RCS) is the mean pairwise cosine between repeated mean LVR trajectories under each condition. (b) Rank90, Rank95, and Rank99 count the principal directions needed to explain 90%, 95%, and 99% of the variance; PR is the participation ratio.

(a) Representation consistency under perturbations

Family	Condition	Mean RCS
Reference	No augmentation	0.99999997
Image augmentation	Brightness/contrast/rotation/crop	0.99917778
Mask sweep	Strength 0.05	0.99987373
Mask sweep	Strength 0.30	0.99974626
Blur+jitter	Strength 0.10	0.99994875
Blur+jitter	Strength 0.50	0.99988020
Affine sweep	Tested affine range	0.99917–0.99939

(b) Effective-rank diagnostics

Dataset	Representation	Shape	Rank90	Rank95 / Rank99	PR
BLINK	visual embedding	163868×3584	428	713 / 1697	54.31
	LVR latent tokens	11152×3584	4	5 / 24	2.60
	LVR mean	–	4	6 / 56	1.61
	last LVR token	–	2	3 / 17	1.35
VISCOT	visual embedding	242899×3584	404	675 / 1641	58.55
	LVR latent tokens	81945×3584	51	169 / 909	5.70
	LVR mean	–	21	65 / 289	4.70
	last LVR token	–	36	111 / 414	5.66

G.5 Target Alignment and Answer Readout

Table G.3 compares autoregressively generated latent trajectories with the visual targets used during reconstruction training. Each mixed schedule supplies the first k target vectors and lets the remaining latent states free-run. Supplying more targets increases cosine similarity, while fully free-running states remain weakly aligned with the targets. The fully forced trajectory matches the target by construction; it does not measure learned alignment during inference.

Table G.4 summarizes answer readout and residual injection. The attention measurements describe associations with answer correctness; residual injection measures local sensitivity. Neither alone assigns causal credit to a latent token.

Table G.3: **Alignment with visual targets under target forcing.** The first k latent positions receive the visual target vectors; later positions are generated autoregressively. MSE and cosine similarity compare the resulting trajectory with the visual targets.

	BLINK		VISCOT	
Generation schedule	**MSE**	**Cosine**	**MSE**	**Cosine**
Free-running	1.4218	0.2145	1.7700	0.2863
Force k=4	–	0.4221	–	0.4898
Force k=8	–	0.6145	–	0.6619
Force all k=16	–	1.0000	–	1.0000

Table G.4: **Answer readout and residual-injection diagnostics.** Summary statistics from generated rollouts. Attention deltas compare correct and incorrect predictions.

Diagnostic	**Evaluation**	**Reported values**
Answer-to-latent-token attention	`MMVP`	Acc. 64.00%; answer-to-LVR mass $\Delta = +0.0099$; top-1 attention $\Delta = +0.0307$.
	`BLINK`	Acc. 49.28%; mass $\Delta = +0.0029$; top-1 attention $\Delta = +0.0217$.
	`HR-4K` / `HR-8K`	Acc. 56.88% / 48.62%; correct-answer mass and top-1 mass are higher, with smaller deltas at 8K.
Residual injection	47,012 generated rollout records	Best Δ toward counterfactual prediction: $+0.0068$ to $+0.0137$; logit-margin deltas up to $+0.0358$.

G.6 Task Structure and Counterfactual Sensitivity

Table G.5(a) tests whether representations encode the BLINK task label. Adjusted Rand index (ARI) measures cluster alignment with task labels; probe accuracy uses 5-fold cross-validation.

Table G.5: **Task decodability and sensitivity to answer-changing edits.** (a) Task-label structure on BLINK (N=697). (b) Sensitivity to synthetic image edits (N=2,048; 512 pairs per edit). LVR distance uses the mean-pooled latent trajectory; final distance uses the final hidden representation.

(a) Task-label decodability

Representation	**ARI**	**Bootstrap ARI**	**Linear accuracy**	**MLP accuracy**
h_{pre}	0.9489	0.9492	0.9994	0.9960
LVR mean	0.3885	0.3905	0.9991	0.9822
$h_{\text{lvr,last}}$	0.1298	0.1430	0.9966	0.9684
Final hidden	0.2714	0.2558	0.9954	0.9641

(b) Counterfactual sensitivity

Edit	**Should flip**	**Model flip**	**Correct flip**	**LVR distance**	**Final distance**
Color change	86.33%	5.66%	5.86%	0.000158	0.004764
Object removal	81.45%	13.09%	6.45%	0.000445	0.022313
Shape swap	84.57%	12.70%	5.27%	0.000253	0.012438
Spatial swap	84.96%	11.33%	4.69%	0.000162	0.008934

Panel (b) edits the image while holding the question fixed. "Should flip" is the ground-truth answer-change rate, and "Model flip" is the prediction change rate. The separately reported "Correct flip" rate counts predictions that reach the edited ground-truth answer. Both distance columns report cosine distance. The correct answer changes for most pairs, but LVR's answers change infrequently and its mean-pooled trajectories move only slightly.

H Additional Qualitative Results

The visualizations below examine individual generated answers and illustrate the visual reasoning tasks covered by our benchmarks. Appendix G reports the dataset-level diagnostics.

H.1 Image Saliency and Token Views

Each model first generates an answer, after which we compute $|\text{attention} \times \text{gradient}|$ saliency for that sequence. Figure 5 in the main text shows three paired image and token views. The image overlays use answer-conditioned $|\text{attention} \times \text{gradient}|$ saliency. The additional cases below show explicitly labeled attention and saliency panels.

H.2 Attention and Saliency Cases

Figures H.1–H.4 compare twelve image–question cases at a common scale. Each includes the answers and four labeled attention/saliency panels, using the token axes and attribution protocol in Appendix G.3.



Figure H.1: **Attention and saliency cases (01–03).** Top to bottom: the cap on a table, the bus beside a person, and the chair beside a tennis player.

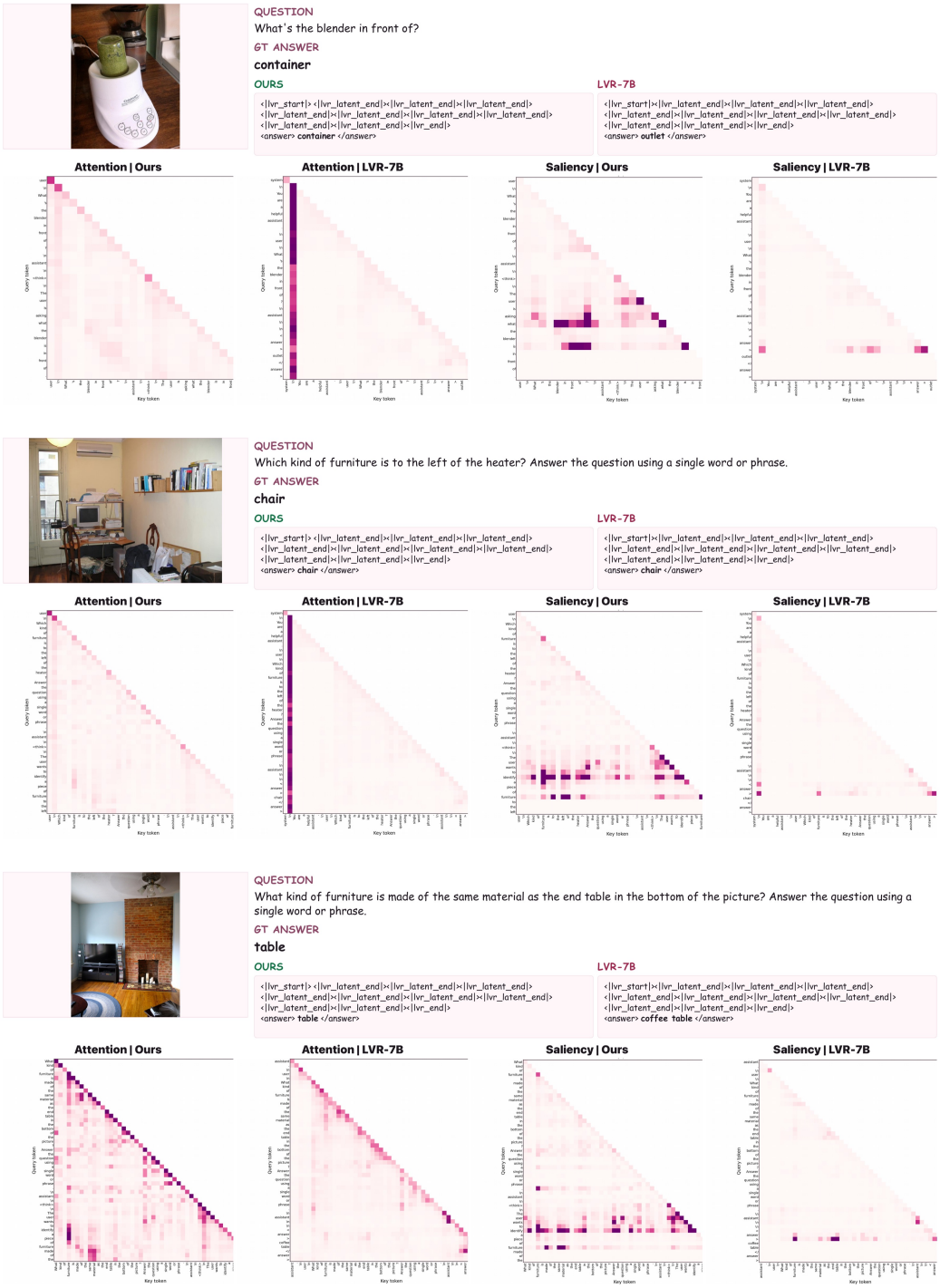


Figure H.2: **Attention and saliency cases (04–06).** Top to bottom: the object behind the blender, the furniture left of the heater, and a furniture-material comparison.

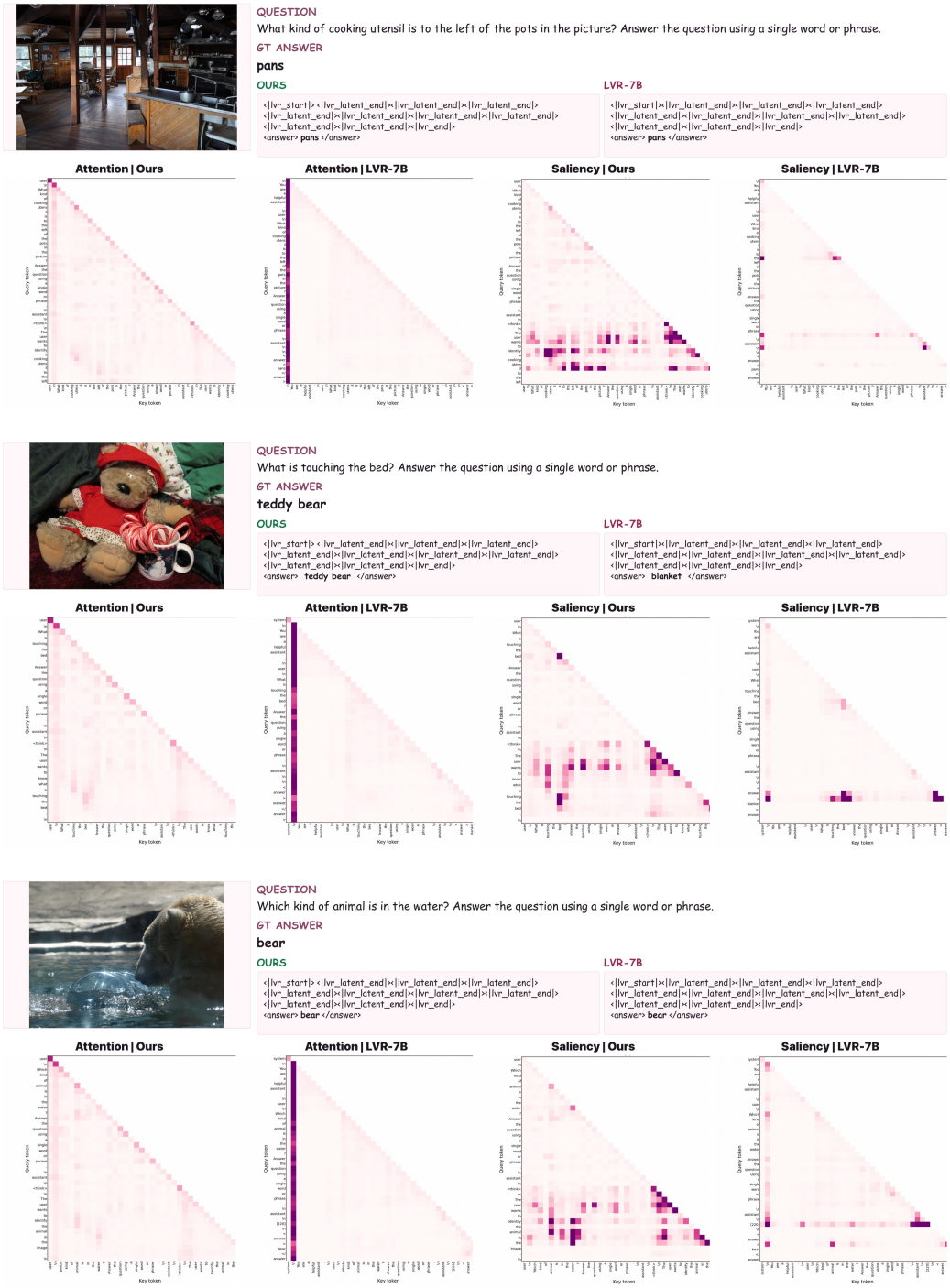


Figure H.3: **Attention and saliency cases (07–09).** Top to bottom: pans left of the pots, the teddy-bear answer comparison, and a bear in water. In the middle example (Case 08), ReaLVR answers "teddy bear" and LVR-7B answers "blanket" to "What is touching the bed?"

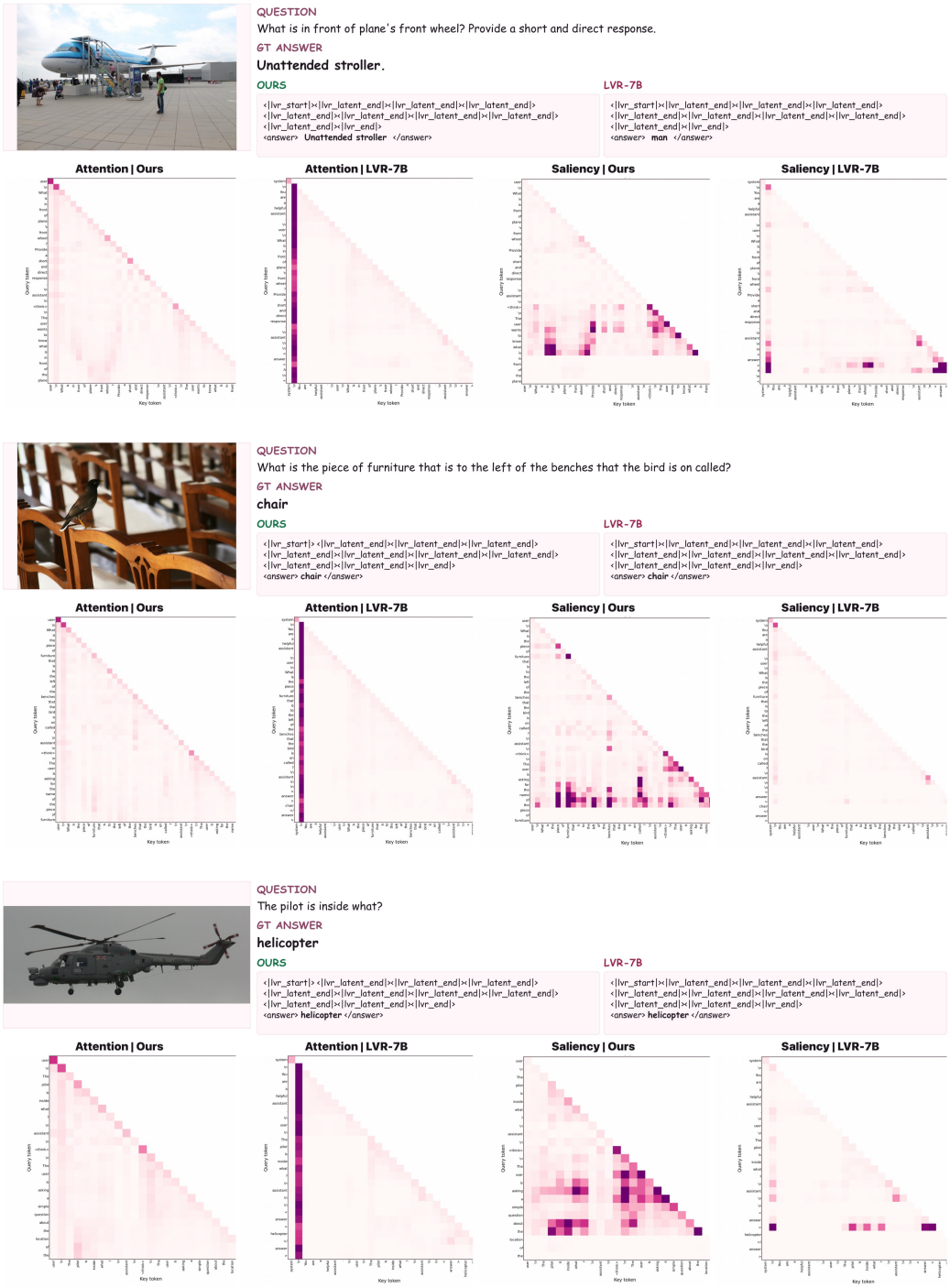


Figure H.4: **Attention and saliency cases (10–12).** Top to bottom: the object in front of an airplane's front wheel, the chair-and-bird scene, and the vehicle containing the pilot.

H.3 Task Examples Across Five Benchmarks

Figure H.5 illustrates the evidence required by each benchmark. Beyond counting and depth comparison, the map question relates numbered locations to countries, and the 8K scene requires finding a small boat before judging its position relative to distant buildings. The chart question combines reading numerical values with subtraction: $1{,}537 - 1{,}393 = 144$.

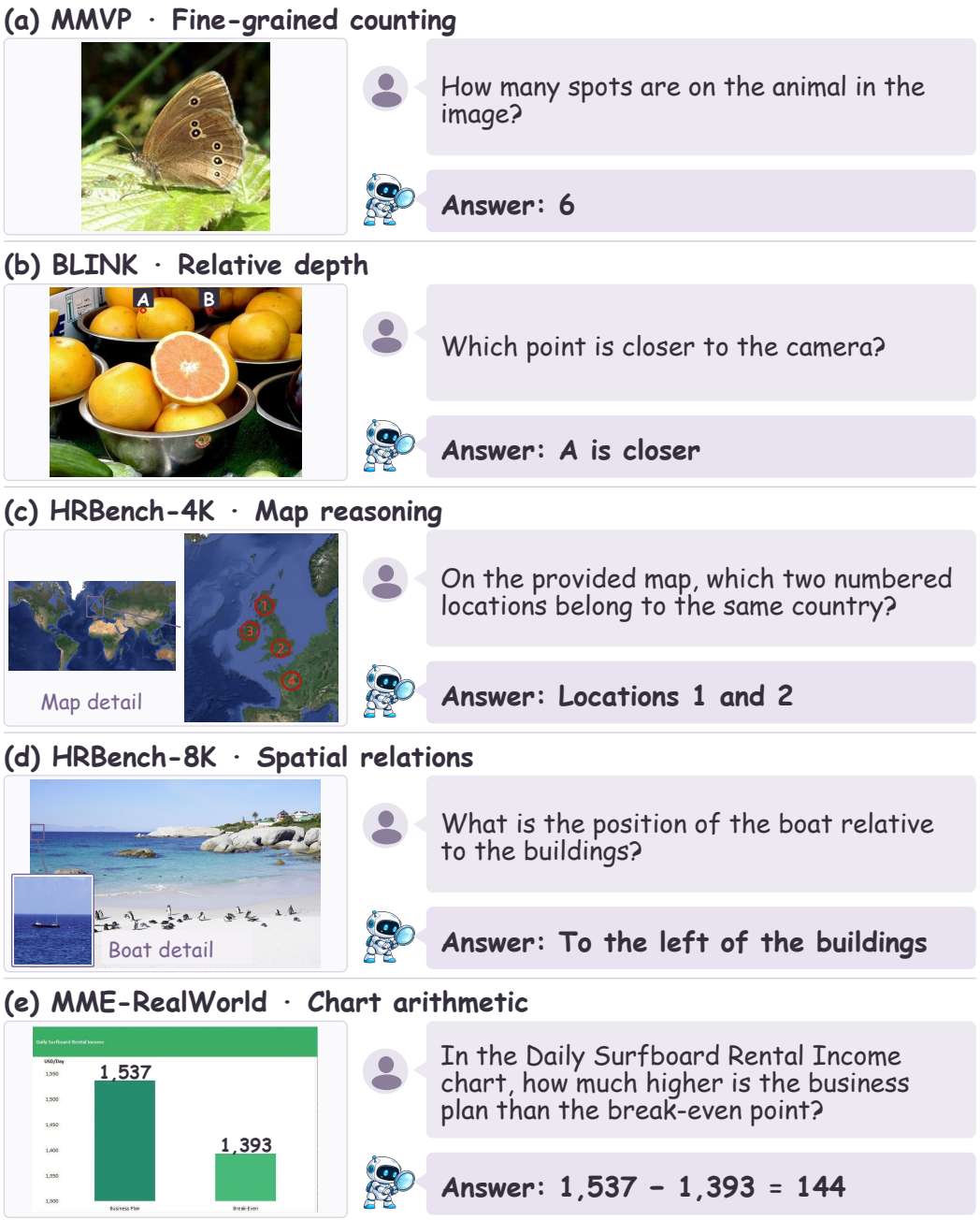


Figure H.5: **One question from each of the five benchmarks.** Images, questions, and answers come from the official datasets (Tong et al., 2024; Fu et al., 2024; Wang et al., 2025; Zhang et al., 2025). Robot bubbles show annotated answers, not recorded ReaLVR predictions; the chart calculation is added for clarity. Insets enlarge source-image details, and the chart is cropped to the relevant panel. Point and value labels are enlarged for readability. Multiple-choice options are omitted, and the chart question is shortened.

H.4 POSITION-WISE LATENT VARIATION

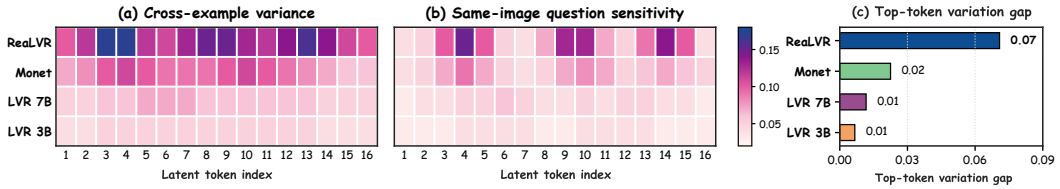


Figure H.6: **Position-wise latent variation.** (a) Cross-example variation. (b) Variation across questions about the same image. Heatmap columns index latent positions 1–16. (c) Reported top-token variation gap: 0.07 for ReaLVR, 0.02 for Monet, and 0.01 for each LVR variant. ReaLVR concentrates more of the measured variation at particular latent positions.

I Attention Analysis Designs

The following designs illustrate three complementary questions: where latent states read visual evidence, how evidence selection changes with the question, and which latent states support the answer. They complement the diagnostic definitions in Appendices G.2 and G.1.

I.1 Visual Evidence and Answer Readout

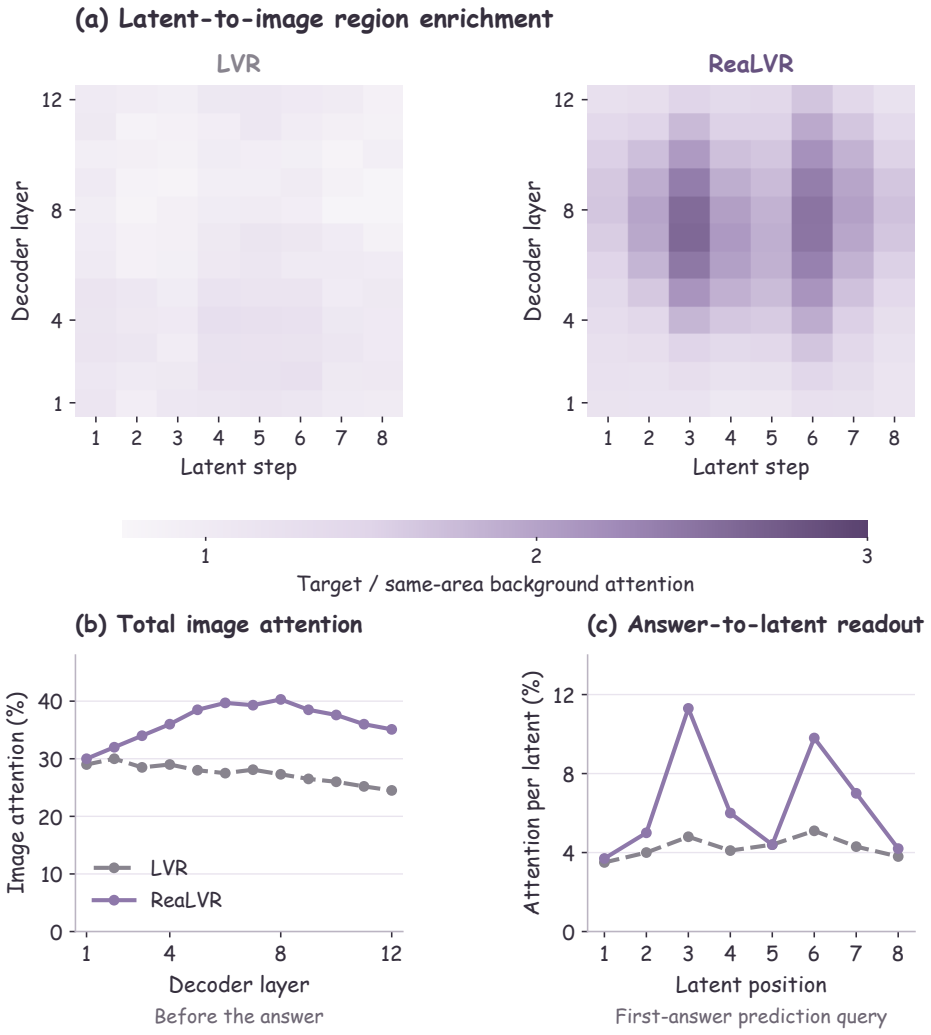


Figure I.1: **Visual evidence and answer readout.** (a) Target-region attention relative to same-area background windows, shown by decoder layer and latent step; 1 means no preference. Both heatmaps share a color scale. (b) Raw attention mass assigned to all image tokens before the answer. (c) Raw attention to each latent state from the query used to predict the first answer token, before that token is supplied. The panels illustrate aggregate post-softmax quantities.

I.2 Question-Dependent Evidence Selection

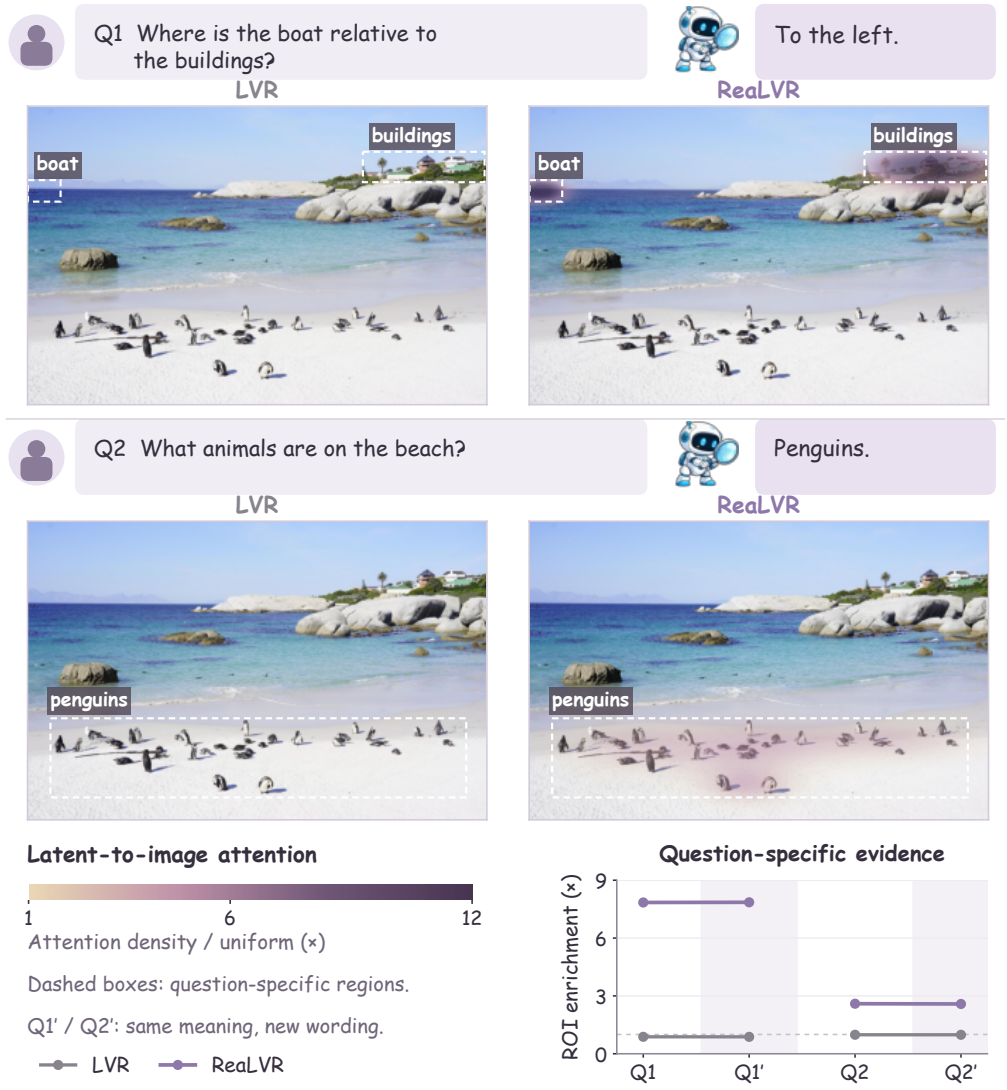


Figure I.2: **Question-dependent evidence selection.** The photograph is HRBench-8K example 798 (Wang et al., 2025). Q1 adapts its spatial-relation question; Q2 and both paraphrases are author-written. Dashed boxes mark the manually specified regions for the boat-and-buildings question and the animal question. All maps share a density scale normalized to a full-image mean of 1. The chart reports mean density within each question's region and compares that question with its paraphrase ($Q1'$ or $Q2'$). This uniform baseline differs from the matched-background baseline in Figure I.1.

I.3 Latent Selection and Fixed-Context Intervention

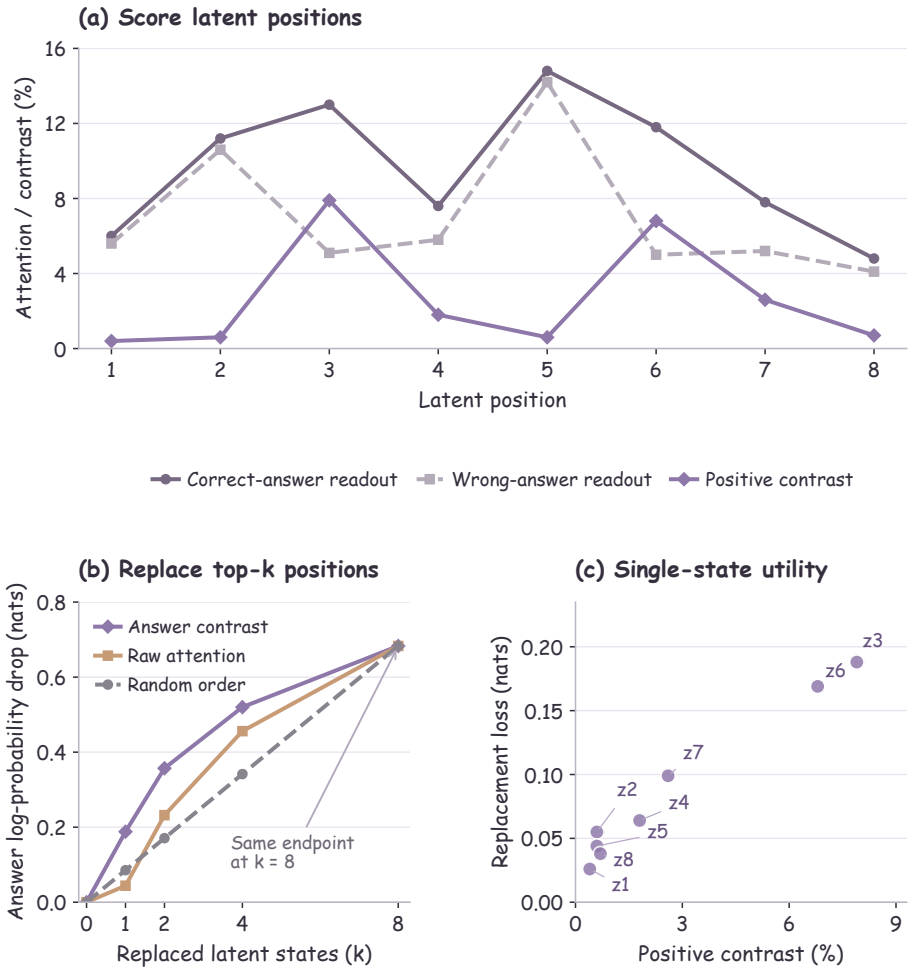


Figure I.3: **Latent selection and fixed-context intervention.** (a) Teacher-forced readout after supplying correct or incorrect answers; positive contrast is the unnormalized positive part of their difference. (b) Correct-answer log-probability loss after replacing positions selected by contrast, raw correct-answer attention, or random ordering, with the other latent inputs fixed. (c) Loss from replacing each single latent position, plotted against that position's positive contrast. Results are averaged over 1000 [BLINK] examples on `Qwen2.5-VL-7B`. All strategies replace all eight positions at $k = 8$ and therefore coincide at the endpoint. Losses are in nats.

J Paired Counterfactual and Mass-Matched Mechanism Tests

This section reports a paired counterfactual comparison, fixed-context latent exchange and a mass-matched test of the proposed evidence pathway.

J.1 Paired counterfactual behavior

For edit type e, let (x_i^0, x_i^1) be the original and edited images with the same question, and let (y_i^0, y_i^1) be their canonical ground-truth answers. We partition the pairs into $\mathcal{C}_e = \{i : y_i^0 \neq y_i^1\}$ and $\mathcal{U}_e = \{i : y_i^0 = y_i^1\}$. The four edit types are color change, object removal, shape swap, and spatial swap; each has 512 pairs before this partition. We evaluate Direct, LVR-RL, and ReaLVR on the identical pairs with the same prompt, image processing, answer parser, and decoding settings. Direct

uses the same pretrained backbone, Stage 2 examples, answer reward, and update budget but omits the latent span.

For $\mathcal{C}_e$, we report accuracy on each view, the fraction with two parseable but different predictions, and the *strict paired correct flip*,

$$F_{\text{correct}}(e) = \frac{1}{|\mathcal{C}_e|} \sum_{i \in \mathcal{C}_e} \mathbf{1}[\widehat{y}_i^0 = y_i^0 \ \wedge \widehat{y}_i^1 = y_i^1]. \tag{J.1}$$

Because $y_i^0 \neq y_i^1$, this event requires a correct answer change. For $\mathcal{U}_e$, the false-flip rate counts parseable predictions that change although the answer does not. Parse failures count as incorrect and not as valid prediction changes. Accuracy, prediction change, and strict correct flip use $|\mathcal{C}_e|$ as their denominator; false flip uses $|\mathcal{U}_e|$. Parse coverage is the fraction of all 2×512 view-level predictions with a parseable answer.

Protocol. Each pair is evaluated as a two-turn conversation. The first turn presents x_i^0 with question q_i; the second turn presents the edited image x_i^1 with the same q_i, keeping the first-turn exchange in context. This tests whether a model revises its answer when the visual evidence changes within a dialogue, rather than repeating its earlier prediction. `MMVP` instead scores each image in a separate conversation, where no earlier answer is available to anchor the prediction.

J.2 Latent exchange and position selection

We test whether an answer-changing edit can transfer information through the latent span while the recipient image remains fixed. For each pair in $\mathcal{C}_e$, we generate the two latent spans independently and test both donor–recipient directions within the same model. We keep the recipient image, question, control markers, and unselected latent inputs fixed, replacing selected recipient states with donor states at the same positions before answer decoding. A self-swap using the recipient's own states is the sham control; an unrelated-image donor tests generic perturbation effects. All swaps use the same replacement rule and donor states across position selectors.

For $k \in \{1, 2, 4\}$, we compare positions ranked by the positive correct-versus-donor-answer readout contrast on the unmodified recipient trajectory with uniformly sampled random positions of the same count. Random selections are repeated with fixed seeds. At $k = K = 8$ the entire span is replaced, so this endpoint measures whole-span sensitivity and cannot test the ranking. We report the swap-minus-sham change in the frequency of the donor ground-truth answer and in its score margin over the recipient ground-truth answer. Each score is the mean teacher-forced log probability over answer-content tokens. These fixed-context swaps measure local answer dependence.

J.3 Mass-matched where-by-what ablation

This 2×2 diagnostic crosses *where* visual supervision is allocated (uniformly or by answer contrast) with *what* evidence loss is used (positive-only alignment or positive–negative visual contrast). Let $w_t = \eta/K + (1-\eta)\gamma_t$ as in Section 4.4. For uniform (U) and contrast (C) allocation, respectively, define

$$q_t^{\text{U}} = \frac{1}{K}, \qquad q_t^{\text{C}} = \frac{w_t}{\sum_{s=1}^K w_s} = \frac{\eta/K + (1-\eta)\gamma_t}{\eta + (1-\eta)\sum_{s=1}^K \gamma_s}. \tag{J.2}$$

With $\eta > 0$, both rules satisfy $\sum_t q_t = 1$ on every example, including those with no valid wrong answer. The positive-only and contrastive per-position losses are

$$\ell_t^+ = [m_{\text{ev}} - \text{sim}(z_t, p^+)]_+, \qquad \ell_t^\pm = [m_{\text{ev}} - g_t]_+, \tag{J.3}$$

where g_t is the positive-versus-hardest-negative margin in Section 4.2. Each arm adds $\lambda_{\text{ev}} \sum_t \text{sg}(q_t)\ell_t^V$ to the same LVR Stage 2 objective. All arms share the Stage 1 checkpoint, examples and order, GRPO rollout group size and sampling procedure, reward, latent length, optimizer, update count, and evidence-loss coefficient. Negative prototypes and answer-read branches are computed in every arm to keep the forward budget comparable, even when their outputs do not enter that arm's loss.

Normalization matches the *coefficient mass* of visual supervision, not necessarily the active-hinge fraction or gradient magnitude. We therefore treat active-hinge fractions, gradient norms, and training compute as separate checks when interpreting the four-way comparison. We report the five benchmark accuracies and, for each independent Stage 2 seed, macro-average the strict paired correct-flip and false-flip rates over the four edit types. The interaction between the two factors can be assessed from matched-seed differences, rather than inferred solely from the best single row.

Table J.1: **Complete paired counterfactual evaluation.** The first four rate columns use changed-answer pairs $\mathcal{C}_e$; false flip uses unchanged-answer pairs $\mathcal{U}_e$. Each edit type is evaluated on the same pairs for all methods. Rates are percentages computed from single deterministic greedy rollouts on the partitioned sets $\mathcal{C}_e$ and $\mathcal{U}_e$; counts are in parentheses (parse coverage counts view-level predictions).

Edit	Method	$\lvert\mathcal{C}_e\rvert/\lvert\mathcal{U}_e\rvert$	**Original acc.** (A_0)	**Edited acc.** (A_1)	**Prediction change** (Δ)	**Strict paired correct flip** (F_{correct})	**False flip on $\mathcal{U}_e$**	**Parse coverage**
Color change	Direct	442 / 70	75.8 (335)	70.4 (311)	61.3 (271)	53.6 (237)	11.4 (8)	99.8 (1022)
	LVR-RL	442 / 70	78.5 (347)	8.1 (36)	6.1 (27)	4.8 (21)	2.9 (2)	99.8 (1022)
	ReaLVR	442 / 70	**85.3** (377)	**81.2** (359)	**80.5** (356)	**73.1** (323)	**5.7** (4)	**100.0** (1024)
Object removal	Direct	417 / 95	74.3 (310)	68.6 (286)	63.8 (266)	52.3 (218)	12.6 (12)	99.5 (1019)
	LVR-RL	417 / 95	77.2 (322)	11.3 (47)	14.6 (61)	6.7 (28)	6.3 (6)	99.8 (1022)
	ReaLVR	417 / 95	**84.4** (352)	**79.1** (330)	**77.5** (323)	**69.8** (291)	**6.3** (6)	**100.0** (1024)
Shape swap	Direct	433 / 79	75.1 (325)	69.5 (301)	60.5 (262)	51.7 (224)	11.4 (9)	99.8 (1022)
	LVR-RL	433 / 79	77.8 (337)	9.9 (43)	13.6 (59)	5.8 (25)	7.6 (6)	100.0 (1024)
	ReaLVR	433 / 79	**83.8** (363)	**78.5** (340)	**78.3** (339)	**70.2** (304)	**5.1** (4)	**100.0** (1024)
Spatial swap	Direct	435 / 77	73.8 (321)	67.8 (295)	58.6 (255)	49.4 (215)	13.0 (10)	99.5 (1019)
	LVR-RL	435 / 77	76.6 (333)	9.2 (40)	12.2 (53)	5.1 (22)	6.5 (5)	99.8 (1022)
	ReaLVR	435 / 77	**82.8** (360)	**77.2** (336)	**76.8** (334)	**68.3** (297)	**6.5** (5)	**100.0** (1024)

Limitations and future work. Our visual evidence target is less spatially specific when region annotations are unavailable, and inference currently uses a prescribed latent-token budget. Future work could derive finer evidence targets from weak supervision and adapt the latent budget to each question.

Table J.2: **Fixed-context latent exchange.** The recipient image and question remain fixed. Donor-answer lift and donor-margin shift are differences from the self-swap control. Contrast and random rows at the same k use the same donors. The full-span row does not test position selection. Donor-answer lift is a percentage-point change in donor-answer frequency.

Method	**Positions**	k	**Donor-answer lift** (pp)	**Donor-margin shift** (nats)
LVR-RL	Contrast	1	+1.2	+0.02
	Random	1	+0.7	+0.01
	Contrast	2	+2.4	+0.04
	Random	2	+1.4	+0.02
	Contrast	4	+4.1	+0.07
	Random	4	+2.9	+0.04
	Full span	8	+5.8	+0.09
	Unrelated donor	8	-0.3	-0.04
ReaLVR	**Contrast**	1	**+12.4**	**+0.23**
	Random	1	+3.1	+0.05
	Contrast	2	**+23.6**	**+0.44**
	Random	2	+9.8	+0.17
	Contrast	4	**+35.2**	**+0.66**
	Random	4	+21.5	+0.39
	Full span	8	+46.8	+0.89
	Unrelated donor	8	+0.2	-0.31

Table J.3: **Mass-matched** 2×2 **ablation.** U/C denote uniform/answer-contrast position weights; $+$ and $\pm$ denote positive-only and positive–negative visual objectives. Every arm has unit per-example routing mass. Values are percentages, reported as mean $\pm$ standard deviation across independent training seeds.

Where	**What**	**MMVP**	**BLINK**	**HR-4K**	**HR-8K**	**MME-RW**	**Five-task mean**	**Strict paired correct flip**	**False flip**
U	$+$	67.2 ± 0.3	54.1 ± 0.2	70.3 ± 0.2	64.9 ± 0.3	50.8 ± 0.2	$\mathbf{61.5 \pm 0.2}$	22.4 ± 1.2	8.9 ± 0.6
C	$+$	69.1 ± 0.4	54.5 ± 0.3	70.7 ± 0.3	65.4 ± 0.2	51.1 ± 0.2	$\mathbf{62.2 \pm 0.3}$	36.8 ± 1.5	7.8 ± 0.5
U	$\pm$	69.6 ± 0.3	54.7 ± 0.2	71.1 ± 0.2	65.7 ± 0.3	51.4 ± 0.3	$\mathbf{62.5 \pm 0.2}$	42.5 ± 1.3	8.1 ± 0.4
C	$\pm$	$\mathbf{71.8 \pm 0.4}$	$\mathbf{55.7 \pm 0.3}$	$\mathbf{71.7 \pm 0.2}$	$\mathbf{66.5 \pm 0.3}$	$\mathbf{52.1 \pm 0.2}$	$\mathbf{63.6 \pm 0.2}$	$\mathbf{59.2 \pm 1.6}$	$\mathbf{7.2 \pm 0.4}$