VISER: LEARNING TO SEE WHAT MATTERS FOR EFFICIENT VISUAL REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent "Thinking with Images" methods have improved visual reasoning by explicitly inspecting image regions or reasoning with latent visual representations. However, many of these methods introduce additional visual interactions or modify the base model's inference procedure. How can models learn to better attend to and use visual evidence while maintaining efficient inference? We propose **VISER**, a two-stage post-training framework that combines visual-token evidence supervision with reinforcement learning to guide evidence use in answer generation. The first stage jointly optimizes evidence prediction at visual-token positions and answer generation. The second stage derives contrastive rewards from the model's responses to original, evidence-corrupted, and background-perturbed images, combining them with task-correctness rewards for reinforcement learning. These rewards encourage reliance on visual evidence and robustness to irrelevant background perturbations. All additional supervision is confined to training; VISER preserves the base model's inference procedure without additional region cropping, visual tool calls, latent visual reasoning tokens, or multi-turn visual interactions. Experiments demonstrate that VISER achieves the best overall performance among the compared methods across five image VQA benchmarks while maintaining inference latency close to that of the base model. Under the same evaluation setup on the full VStar benchmark, VISER incurs approximately 4% latency overhead over the base model and achieves approximately $2.4\times$, $58\times$, and $9.3\times$ speedups over TreeVGR, DeepEyes, and DeepEyesV2, respectively.

1 INTRODUCTION

Visual reasoning is an important aspect of human intelligence and a core capability of advanced multimodal models. Despite substantial progress in visual understanding, vision-language models (VLMs) may still over-rely on language priors (Leng et al., 2024) or place disproportionate attention on previously generated text while neglecting image tokens (Huang et al., 2024). Recent "thinking with images" methods (OpenAI, 2025) have made progress by incorporating visual information more actively into the reasoning process.

One line of work makes visual grounding explicit, often representing relevant region locations as coordinates in the generated text. Methods such as ViGoRL (Sarch et al., 2025), DeepEyes (Zheng et al., 2026), and Pixel-Reasoner (Su et al., 2025) use region localization, image cropping or magnification, and visual tools to obtain more detailed local information during reasoning. These methods improve performance on fine-grained visual tasks, but region localization and image processing also introduce computational overhead or system complexity at inference time. For methods that repeatedly revisit images, additional visual interactions further increase inference latency.

Another line of work, latent visual reasoning, introduces *continuous latent states* to process visual information during reasoning, adding intermediate positions to the inference sequence beyond the initially encoded visual tokens. In LVR (Li et al., 2026a), Mirage (Yang et al., 2026), and Monet (Wang et al., 2026b), decoder hidden states are propagated as continuous latent tokens before or between text segments. Latent Implicit Visual Reasoning (Li et al., 2026b) instead appends learnable latent tokens that aggregate visual information through attention, without requiring explicit supervision of intermediate visual targets. These designs incorporate latent visual reasoning tokens into the

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.

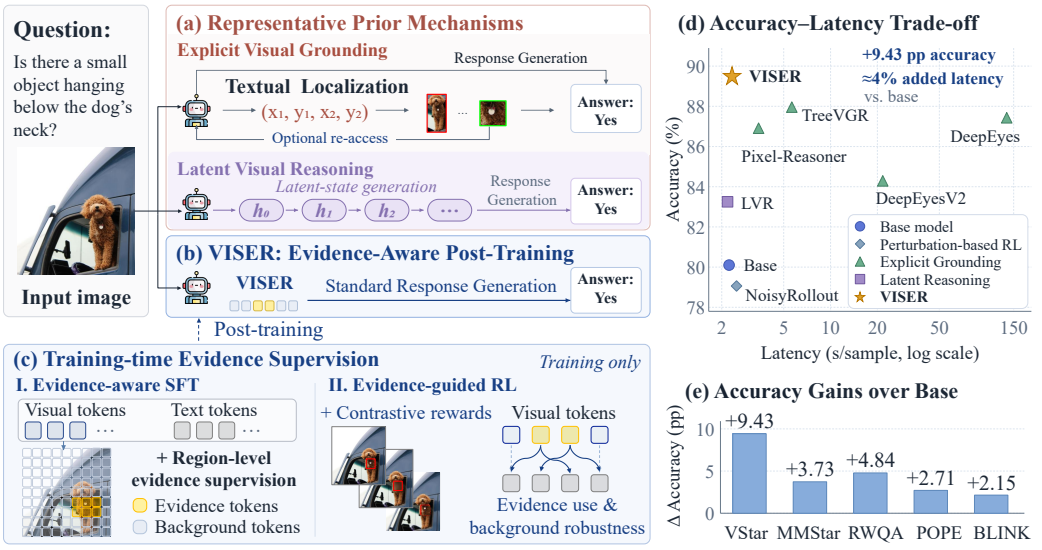


Figure 1: **VISER learns to use visual evidence through post-training. (a,b)** Explicit grounding and latent reasoning mechanisms versus VISER's standard single-turn inference. **(c)** Training-only supervision combines evidence localization with contrastive RL using original, evidence-corrupted, and background-perturbed images. **(d)** Accuracy–latency comparison on the full VStar benchmark under a shared evaluation setup (logarithmic latency axis). **(e)** Accuracy gains over Qwen2.5-VL-7B across five benchmarks. RWQA denotes RealWorldQA.

computation that supports answer generation, with method-specific mechanisms for constructing and training these tokens.

Despite their different mechanisms, many of these methods share a common strategy: extending the base model's inference procedure to enhance visual information use, either through additional visual interactions or through latent visual reasoning tokens. This motivates a complementary question: *How can models learn to better attend to and use visual evidence while preserving the base model's inference procedure?*

To explore this question, we propose **VISER**, a two-stage post-training framework for efficient single-turn visual reasoning (Figure 1(b,c)). We first use supervised fine-tuning (SFT) to incorporate evidence supervision into visual-token representations, then apply reinforcement learning (RL) to guide evidence use during reasoning. Specifically, Stage I introduces a lightweight evidence head, a trainable linear layer that predicts evidence scores from the multimodal backbone's final-layer hidden states at visual-token positions. Using binary labels derived from visual evidence annotations, we jointly optimize evidence localization and answer generation. Stage II reinforces evidence use through contrastive rewards. For each image–question pair, we construct three inputs: the original image, an evidence-corrupted image, and a background-perturbed image. We derive contrastive rewards from the model's responses across these conditions and combine them with task-correctness rewards, encouraging reliance on visual evidence and robustness to irrelevant background perturbations. All additional supervision and visual contrasts are confined to post-training. At inference time, VISER preserves the base model's inference procedure: it encodes the original image and autoregressively generates text conditioned on the image and question, without additional region cropping, visual tool calls, latent visual reasoning tokens, or multi-turn visual interactions.

Experiments show that VISER achieves the best performance among the compared open-source methods on VStar (Wu and Xie, 2024), MMStar (Chen et al., 2024), RealWorldQA (Grok, 2024), POPE (Li et al., 2023), and BLINK (Fu et al., 2024), with consistent improvements over the base model summarized in Figure 1(e). Figure 1(d) highlights its favorable accuracy–latency trade-off: VISER achieves the highest accuracy among the compared methods on the full VStar benchmark while maintaining near-base-model latency. VISER also substantially improves over the base model on the high-resolution benchmarks HR-Bench 4K/8K (Wang et al., 2025) and MME-RealWorld-Lite (Zhang et al., 2025), although methods that explicitly access local regions retain an advantage in some settings with greater demands on fine-grained visual perception. Together, these results suggest

that visual evidence supervision during post-training can strengthen the use of task-relevant visual information in standard answer generation without extending the inference pipeline. Our main contributions are as follows:

- We introduce VISER, a two-stage post-training framework that promotes visual evidence use for efficient single-turn visual reasoning within the standard VLM inference pipeline, without requiring region cropping, tool calls, or additional visual reasoning tokens.
- We combine visual evidence localization supervision with contrastive visual reinforcement learning. The former provides region-level evidence supervision at visual-token positions, while the latter constructs training signals from original, evidence-corrupted, and background-perturbed inputs to encourage reliance on visual evidence and robustness to irrelevant perturbations. Both forms of supervision are confined to training.
- We demonstrate strong performance across five image VQA benchmarks while maintaining inference speed close to that of the base model, with further evaluations on high-resolution inputs and a smaller model scale, alongside ablations of the training components.

2 METHOD

VISER is a two-stage post-training framework that improves the representation and use of visual evidence. Stage I incorporates evidence-region supervision into visual-token representation learning to build evidence-aware representations. Stage II uses feedback from responses to original and visually modified inputs to guide evidence use during answer generation. All additional supervision is confined to training. At inference time, the model retains standard single-turn autoregressive generation conditioned on the image and question.

2.1 PROBLEM SETUP

Given an image I and a question q, a VLM generates a response y autoregressively according to $\pi_\theta(y \mid I, q)$, where θ denotes the model parameters. Each training sample additionally includes a ground-truth answer $y^\star$ and an annotated evidence bounding box R. The evidence annotations provide visual-token supervision in Stage I and define the visual interventions in Stage II. At inference time, only the image and question are required.

2.2 STAGE I: LEARNING EVIDENCE-AWARE REPRESENTATIONS

Stage I jointly supervises response generation and visual-token evidence prediction. We project the annotated region R onto the visual-token grid to obtain binary labels m_j, with 1 indicating evidence and 0 indicating background. A linear head predicts the evidence probability $\hat{m}_j = \mathrm{sigmoid}(u^\top h_j^v + b)$, where h_j^v is the j-th visual token's representation at the final multimodal decoder layer and u, b are trainable parameters. We optimize the binary cross-entropy loss:

$$\mathcal{L}_{\mathrm{vg}} = -\frac{1}{N_v} \sum_{j=1}^{N_v} \left[m_j \log \hat{m}_j + (1 - m_j) \log(1 - \hat{m}_j) \right], \tag{1}$$

where N_{v} is the number of visual tokens. We jointly optimize this loss with next-token prediction on the supervised responses as $\mathcal{L}_{\mathrm{I}} = \mathcal{L}_{\mathrm{NTP}} + \lambda_{\mathrm{vg}} \mathcal{L}_{\mathrm{vg}}$, where λ_{vg} controls the evidence supervision weight. The evidence prediction head is used only during training.

2.3 STAGE II: GUIDING EVIDENCE USE WITH VISUAL CONTRASTIVE FEEDBACK

Stage I builds evidence-aware representations, but does not directly supervise how answer generation responds to changes in evidence versus background regions. Stage II provides this feedback by comparing responses to original and intervened images. Inspired by learning from different views (Chen et al., 2020b), we construct visual contrastive inputs to encourage correct original-image answers, sensitivity to evidence corruption, and stability under background perturbation.

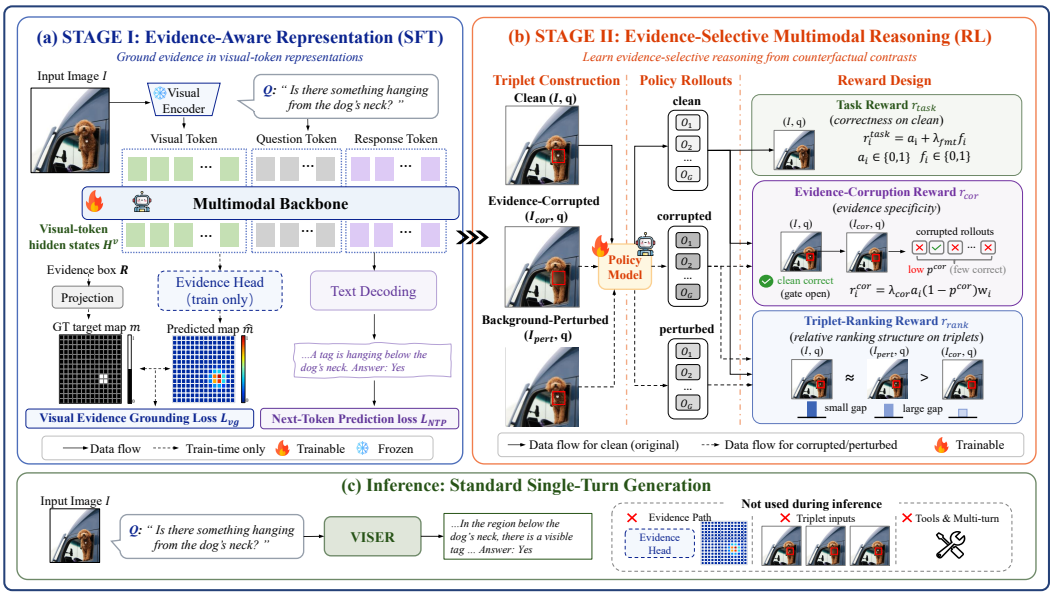


Figure 2: **Overview of VISER.** VISER improves visual evidence utilization through two training stages. Stage I applies evidence supervision at visual-token positions. Stage II encourages evidence-selective reasoning through counterfactual reinforcement over clean, evidence-corrupted, and background-perturbed inputs. At inference time, VISER uses the original input with standard single-turn autoregressive generation.

Constructing visual contrastive inputs. For each training sample, we construct I_{cor} by applying strong pixelation or blurring within the evidence region R. To construct I_{pert}, we preserve R and apply mild blurring only outside it. Together with the original image, these form a visual contrastive triplet $(I, I_{\text{cor}}, I_{\text{pert}})$. We prepare multiple candidates for each intervention type and randomly select one from each type at every training step to reduce dependence on a single intervention configuration.

Sampling and feedback collection. We initialize the policy with the model obtained from Stage I. For each question, the sampling policy π_{old} generates G response trajectories $\{y_i\}_{i=1}^{G}$ from the original image and also samples responses from each intervention branch. An auxiliary language model judges semantic equivalence between the predicted and ground-truth answers, yielding a binary correctness label $c(y, y^\star) \in \{0, 1\}$. For each original-image trajectory, $a_i = c(y_i, y^\star)$.

The sampled accuracy of each intervention branch is

$$p^b = \frac{1}{G_b} \sum_{j=1}^{G_b} c(y_j^b, y^\star), \qquad b \in \{\text{cor}, \text{pert}\}, \tag{2}$$

where G_b and y_j^b denote the sample count and the j-th response in branch b, respectively. The statistics p^{cor} and p^{pert} are shared by all original-image trajectories for the same question and provide question-level visual contrastive feedback.

Reward design. We combine a task reward, an evidence-corruption reward, and a triplet-ranking reward to incorporate both answer correctness and visual contrastive feedback into policy learning.

(1) Task reward. The task reward $r_i^{\text{task}} = a_i + \lambda_{\text{fmt}} f_i$ encourages correct answers and format compliance on original images. Here, $f_i = 1$ for a compliant response and 0 otherwise, and λ_{fmt} is the format reward weight.

(2) Evidence-corruption reward. For correct original-image trajectories, a higher failure rate $1 - p^{\text{cor}}$ after evidence corruption provides a stronger contrastive signal. We use this signal to scale a signed relative likelihood weight w_i, which favors higher-likelihood correct trajectories under the sampling policy:

$$r_i^{\text{cor}} = \lambda_{\text{cor}} a_i (1 - p^{\text{cor}}) w_i, \tag{3}$$

where λ_{cor} controls the reward weight. We compute each original-image trajectory's average token log-probability and obtain w_i through within-group standardization and clipping:

$$\ell_i = \frac{1}{T_i}\sum_{t=1}^{T_i}\log \pi_{\text{old}}(y_{i,t} \mid I, q, y_{i,<t}), \qquad w_i = \text{sg}\left[\text{clip}\left(\frac{\ell_i - \mu_\ell}{\max(\sigma_\ell, \epsilon_\ell)}, -c_w, c_w\right)\right]. \tag{4}$$

Here, T_i counts valid generated tokens, and μ_ℓ and σ_ℓ are computed over all original-image trajectories for the same question, including incorrect ones. The constants ϵ_ℓ and c_w ensure numerical stability and bound the weights, respectively; sg denotes stop-gradient. Positive and negative weights respectively increase and decrease the rewards of correct original-image trajectories.

(3) Triplet-ranking reward. To account for response stability under background changes, we penalize deviations from three desired relations using hinge terms. For each correct original-image trajectory, its correctness value 1 serves as the anchor:

$$\begin{aligned}\mathcal{L}_{\text{rank},i} = a_i\Big([m_{\text{oc}} - (1 - p^{\text{cor}})]_+ \\ + [m_{\text{pc}} - (p^{\text{pert}} - p^{\text{cor}})]_+ + [(1 - p^{\text{pert}}) - m_{\text{op}}]_+\Big).\end{aligned} \tag{5}$$

The corresponding reward is $r_i^{\text{rank}} = -\lambda_{\text{rank}}\mathcal{L}_{\text{rank},i}$, where λ_{rank} is the penalty weight and $[x]_+ = \max(x, 0)$. For correct original-image trajectories, the three terms encourage $1 - p^{\text{cor}} \geq m_{\text{oc}}$, $p^{\text{pert}} - p^{\text{cor}} \geq m_{\text{pc}}$, and $1 - p^{\text{pert}} \leq m_{\text{op}}$, respectively.

Policy optimization. The total reward for each trajectory sampled with the original image and question as input is

$$r_i = a_i + \lambda_{\text{fmt}} f_i + \lambda_{\text{cor}} a_i (1 - p^{\text{cor}}) w_i - \lambda_{\text{rank}}\mathcal{L}_{\text{rank},i}. \tag{6}$$

We combine all reward components before computing group-relative advantages. For group filtering, we evaluate this reward with $w_i = 1$ and discard groups whose original-image trajectories receive nearly identical rewards. This prevents likelihood weighting alone from making a group eligible for optimization. For each retained group, we compute advantages from the final rewards r_i, including likelihood weighting:

$$\hat{A}_i = \frac{r_i - \bar{r}}{\sigma_r + \epsilon_A}, \qquad \bar{r} = \frac{1}{G}\sum_{j=1}^{G} r_j, \tag{7}$$

where σ_r is the within-group reward standard deviation and ϵ_A ensures numerical stability. Each trajectory-level advantage is assigned to its valid generated tokens. All original-image trajectories in retained groups contribute to policy optimization, including incorrect ones. The intervention branches provide reward statistics only and do not directly contribute policy-gradient terms. We optimize the policy using the clipped GRPO objective (Shao et al., 2024b) with KL and entropy regularization, detailed in Appendix E.4.

After training, the model performs standard single-turn autoregressive generation from the original image and question, without the evidence prediction head, intervention branches, or additional visual interactions.

3 Experiments

3.1 Experimental Setup

We evaluate VISER on Qwen2.5-VL-7B-Instruct and its 3B counterpart (Bai et al., 2025). Training primarily uses VisCoT (Shao et al., 2024a), with TreeVGR-RL-37K (Wang et al., 2026a) added in Stage II. Stage I uses LoRA-based supervised fine-tuning. Stage II starts from the merged Stage-I checkpoint and applies GRPO with Qwen3-1.7B as an answer-equivalence judge. The visual encoder remains frozen in both stages. Data processing and training configurations are detailed in Appendix E.

We report accuracy on VStar (Wu and Xie, 2024), MMStar (Chen et al., 2024), RealWorldQA (RWQA) (Grok, 2024), POPE (Li et al., 2023), and BLINK (Fu et al., 2024). We additionally

Table 1: Five-benchmark results. Open-source methods are grouped by Qwen2.5-VL backbone size. Bold and underline indicate the best and second-best scores within each size group. Δ denotes gains over the corresponding base model (pp). "*" denotes results from Li et al. (2026c), and "–" denotes unavailable results. BLINK results for Pixel-Reasoner, DeepEyes, and DeepEyesV2 are omitted because their original pipelines require adaptation to multi-image inputs.

Method	VStar	MMStar	RWQA	POPE	BLINK
Proprietary model (reference)					
GPT-4o	66.0*	64.7*	–	86.9*	–
Qwen2.5-VL-7B-based methods					
Qwen2.5-VL-7B (Bai et al., 2025)	80.10	60.67	67.45	86.51	<u>56.29</u>
NoisyRollout (Liu et al., 2025)	79.06	60.93	67.71	86.61	55.81
LVR (Li et al., 2026a)	83.25	62.67	68.50	85.84	46.92
Pixel-Reasoner (Su et al., 2025)	86.91	<u>63.67</u>	<u>70.07</u>	85.23	–
ViGoRL (Sarch et al., 2025)	79.06	59.93	67.06	87.16	49.55
DeepEyes (Zheng et al., 2026)	87.43	61.33	68.24	<u>89.19</u>	–
DeepEyesV2 (Hong et al., 2026)	84.29	63.47	66.54	–	–
TreeVGR (Wang et al., 2026a)	<u>87.96</u>	62.80	67.84	87.43	53.50
VISER (ours)	**89.53**	**64.40**	**72.29**	**89.22**	**58.44**
Δ over base	+9.43	+3.73	+4.84	+2.71	+2.15
Qwen2.5-VL-3B-based methods					
Qwen2.5-VL-3B (Bai et al., 2025)	<u>70.68</u>	<u>53.67</u>	65.23	<u>86.37</u>	48.66
ViGoRL (Sarch et al., 2025)	69.63	53.53	**66.93**	81.16	<u>49.55</u>
VISER (ours)	**76.44**	**55.00**	<u>66.14</u>	**88.99**	**50.08**
Δ over base	+5.76	+1.33	+0.91	+2.62	+1.42

Table 2: Accuracy and inference efficiency of 7B models on the full VStar benchmark (191 examples, batch size 1). Latency is reported as mean ± SD across examples; relative latency is normalized to Qwen2.5-VL-7B. Best accuracy is bold.

Method	VStar Acc. (%)↑	VLM Calls per sample↓	Latency (s/sample)↓	Relative latency↓
Qwen2.5-VL-7B (Bai et al., 2025)	80.10	1.00	2.23 ± 0.45	1.00×
NoisyRollout (Liu et al., 2025)	79.06	1.00	2.49 ± 0.99	1.12×
LVR (Li et al., 2026a)	83.25	1.00	2.18 ± 1.59	0.98×
Pixel-Reasoner (Su et al., 2025)	86.91	1.63	3.45 ± 0.82	1.55×
ViGoRL (Sarch et al., 2025)	79.06	5.00	26.80 ± 1.05	12.02×
DeepEyes (Zheng et al., 2026)	87.43	1.83	135.42 ± 86.96	60.73×
DeepEyesV2 (Hong et al., 2026)	84.29	2.03	21.65 ± 3.52	9.71×
TreeVGR (Wang et al., 2026a)	87.96	1.00	5.64 ± 3.98	2.53×
VISER (ours)	**89.53**	1.00	2.33 ± 0.37	1.04×

evaluate high-resolution visual understanding on HR-Bench 4K/8K (Wang et al., 2025) and MME-RealWorld-Lite (Zhang et al., 2025). Detailed benchmark descriptions are provided in Appendix E.5. Reproduced comparisons use official baseline implementations and VLMEvalKit (Duan et al., 2024), following common benchmark protocols and answer-extraction rules. We additionally report answer accuracy on the VisCoT validation set in Appendix D.2.

3.2 Performance and Efficiency

VISER improves over its 7B base model across all five image VQA benchmarks, including gains of 9.43 percentage points (pp) on VStar and 4.84 pp on RWQA (Table 1). VISER achieves these accuracy gains while retaining inference latency close to that of the base model. We measure inference latency on all 191 VStar examples with batch size 1 on the same single GPU, retaining each baseline's official default inference configuration (Table 2). VISER takes 2.33 s per example on average, compared with 2.23 s for the base model, corresponding to an overhead of approximately 4%.

Table 3: Results on high-resolution visual understanding benchmarks. HR-4K and HR-8K denote the two versions of HR-Bench; MME-RW denotes MME-RealWorld-Lite. Bold and underline indicate the best and second-best scores. Gains over Qwen2.5-VL-7B are in percentage points.

Method	VStar	HR-4K	HR-8K	MME-RW
Qwen2.5-VL-7B	80.10	68.13	65.50	44.55
NoisyRollout (Liu et al., 2025)	79.06	65.38	53.00	44.92
LVR (Li et al., 2026a)	83.25	70.50	64.00	50.50
Pixel-Reasoner (Su et al., 2025)	86.91	72.88	65.38	52.42
ViGoRL (Sarch et al., 2025)	79.06	73.12	67.50	47.84
DeepEyes (Zheng et al., 2026)	87.43	73.50	69.50	47.52
DeepEyesV2 (Hong et al., 2026)	84.29	<u>75.63</u>	<u>72.38</u>	<u>54.14</u>
TreeVGR (Wang et al., 2026a)	<u>87.96</u>	**76.75**	**74.75**	**57.48**
VISER	**89.53**	74.88	71.38	52.16
Gain over base (pp)	+9.43	+6.75	+5.88	+7.61

Table 4: Ablations of stage composition, the Stage-I objective, and Stage-II rewards. Reward variants start from the Stage-I checkpoint. Avg. is the unweighted mean over the five benchmarks.

Variant	VStar	MMStar	RWQA	POPE	BLINK	Avg.
Stage composition						
Base	80.10	60.67	67.45	86.51	56.29	70.20
+ Stage I only	81.15	62.40	71.37	88.41	57.76	72.22
+ Stage II only	88.48	64.20	69.93	89.10	54.23	73.19
+ VISER (Full)	**89.53**	**64.40**	**72.29**	**89.22**	**58.44**	**74.78**
Stage-I objective						
Answer-only SFT	79.06	62.20	70.46	87.67	57.18	71.31
Evidence-grounded SFT	81.15	62.40	71.37	88.41	57.76	72.22
Stage-II reward						
Task reward only	89.01	63.80	71.50	88.27	58.18	74.15
Task + ranking	89.01	63.93	71.63	89.17	58.07	74.36
Task + corruption	89.01	64.13	72.16	89.10	57.92	74.46
Full VISER	**89.53**	**64.40**	**72.29**	**89.22**	**58.44**	**74.78**

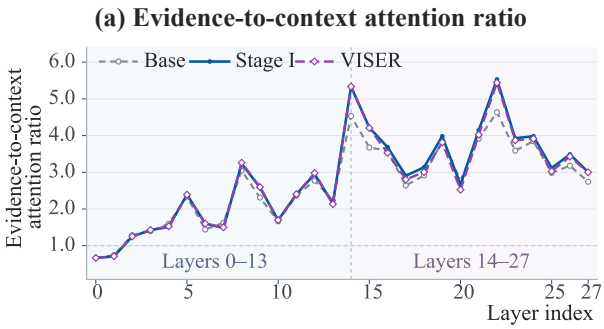
On VStar, VISER achieves speedups of approximately $58\times$ and $9.3\times$ over the multi-turn methods DeepEyes and DeepEyesV2, respectively, and $2.4\times$ over TreeVGR. We further validate VISER at the 3B scale, where it improves over the corresponding base model on all five benchmarks by 0.91 to 5.76 pp.

3.3 High-Resolution Visual Understanding

VISER also improves fine-grained visual understanding on high-resolution benchmarks without additional visual interactions (Table 3). It outperforms the base model by 6.75, 5.88, and 7.61 pp on HR-Bench 4K, HR-Bench 8K, and MME-RealWorld-Lite, respectively. These consistent gains suggest that VISER's training-time supervision also benefits visual understanding in demanding high-resolution settings. However, TreeVGR and DeepEyesV2 achieve higher accuracy on all three benchmarks, indicating that a performance gap remains in these settings. VISER improves evidence use within the standard inference pipeline, while tasks involving small or subtle details in large images may still benefit from explicit region inspection.

3.4 Ablation Studies

Complementary roles of the two stages. Combining the two stages outperforms either stage alone on all five benchmarks (Table 4). Stage I improves performance consistently over the base model,



(b) Stage I relative to Base

Statistic	Layers	Value
Mean ratio increase	0–27	**+6.5%**
	14–27	**+8.7%**
Layers with higher ratio	0–27	21/28
	14–27	13/14

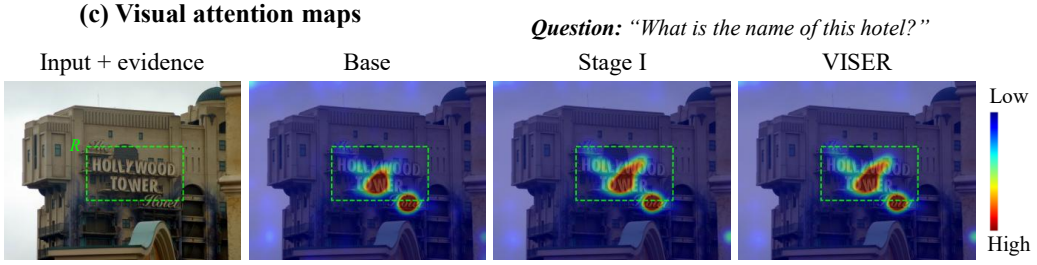


Figure 3: Evidence-related attention patterns. **(a)** Layer-wise evidence-to-context attention ratios. **(b)** Stage-I changes relative to Base, summarized over all layers and layers 14–27. **(c)** Input image and attention maps for Base, Stage I, and VISER. Dashed boxes mark annotated evidence.

whereas Stage II alone yields larger gains on VStar but reduces BLINK accuracy. The full framework combines stronger VStar performance with gains on BLINK, exceeding Stage II alone by 4.21 pp on the latter. This suggests that Stage I provides a useful initialization for RL, helping it achieve gains across benchmarks. These results support the benefit of evidence-supervised fine-tuning before reinforcement learning.

Evidence supervision beyond answer-only SFT. With the same training examples and answer targets, adding evidence supervision improves accuracy on all five benchmarks compared with answer-only SFT (Table 4). This comparison isolates the contribution of the evidence objective from that of training on the same question–answer data. The largest gain is on VStar (+2.09 pp), where answer-only SFT underperforms the base model, while evidence-grounded SFT surpasses it. These results support the contribution of explicit evidence supervision beyond answer supervision alone.

Contrastive feedback beyond task rewards. Starting from the same Stage-I checkpoint, the full reward improves over task-only RL on every benchmark, with a mean gain of 0.62 pp. Adding either ranking or corruption feedback alone improves average accuracy, while combining both yields the highest scores in this comparison. An additional ablation in Appendix C.1 evaluates the contribution of the background-perturbed view.

3.5 Analysis of Visual Evidence Use

Stage I shifts attention toward annotated evidence regions (Figure 3). At each layer, we average attention from answer-token positions over heads and compute the mean attention per visual token within evidence and non-evidence regions. The evidence-to-context ratio divides these two quantities after averaging each across examples. Relative to the base model, Stage I increases this ratio in 21 of 28 layers, including 13 of the final 14 layers. These increases are more pronounced in the middle-to-late layers, where the base model already exhibits higher evidence-to-context attention ratios than in earlier layers. Full VISER retains a similar pattern, showing that the increased relative attention to evidence persists after Stage II.

We next examine how interventions on evidence and background regions affect answer correctness on 500 held-out VisCoT examples (Table 5). Correctness drop rate (CDR) measures the percentage of predictions that become incorrect after an intervention among those correct on the original images. We compare background perturbation ($\mathrm{CDR}_{\mathrm{Pert}}$), background corruption matched to the evidence-corruption area ($\mathrm{CDR}_{\mathrm{Match}}$), and evidence corruption ($\mathrm{CDR}_{\mathrm{Evi}}$). The area-matched comparison

Table 5: Evidence selectivity on 500 held-out VisCoT examples. $\mathrm{Acc}_{\mathrm{clean}}$ denotes clean-input accuracy. CDR measures intervention-induced errors among clean-correct predictions; subscripts indicate background, area-matched background, and evidence interventions. SpecRatio and SpecGap are the ratio and normalized difference between evidence and matched-background CDRs. Corruption-dependent metrics are averaged across four settings.

Method	$\mathrm{Acc}_{\mathrm{clean}} \uparrow$	$\mathrm{CDR}_{\mathrm{Pert}} \downarrow$	$\mathrm{CDR}_{\mathrm{Match}} \downarrow$	$\mathrm{CDR}_{\mathrm{Evi}}$	$\mathrm{SpecRatio} \uparrow$	$\mathrm{SpecGap} \uparrow$
Qwen2.5-VL	77.0	10.13	6.49	32.86	5.05	0.6694
NoisyRollout	77.6	9.28	5.73	32.60	5.73	0.6988
TreeVGR	79.2	4.80	3.72	28.85	7.79	0.7676
VISER Stage I	80.6	3.72	1.61	27.11	17.19	0.8869
VISER	**81.0**	**1.73**	**0.68**	25.56	**39.21**	**0.9472**

helps distinguish sensitivity to evidence location from the effect of corrupting a larger region. Writing $E = \mathrm{CDR}_{\mathrm{Evi}}$ and $B = \mathrm{CDR}_{\mathrm{Match}}$, specificity is summarized by $\mathrm{SpecRatio} = E/B$ and $\mathrm{SpecGap} = (E - B)/(E + B)$; higher values indicate greater sensitivity to evidence corruption relative to matched background corruption. Corruption-dependent metrics are averaged over four settings.

Compared with Stage I, full VISER more than halves CDR under both background interventions, with matched-background CDR falling from 1.61% to 0.68%. Evidence corruption remains substantially more disruptive: its CDR is 25.56% for full VISER, compared with 0.68% for matched background corruption. Although evidence-corruption CDR decreases slightly from Stage I, both specificity scores increase as background sensitivity falls. Together, these results show improved robustness to the tested background changes while preserving a strong contrast between the effects of evidence and background corruption. Further protocol details and per-setting results are provided in Appendix B.2.

4 Related Work

Prior work improves visual reasoning through explicit grounding (Zheng et al., 2026; Wang et al., 2026a), latent reasoning (Li et al., 2026a; Wang et al., 2026b), and perturbation-based learning (Liu et al., 2025). Explicit grounding exposes localized evidence through coordinates or visual operations, while latent reasoning introduces dedicated intermediate tokens. VISER instead applies evidence supervision at existing visual-token positions without requiring these inference-time mechanisms. Most closely related to our Stage II, NoisyRollout (Liu et al., 2025) mixes trajectories from original and distorted images to improve exploration during RL. VISER instead uses evidence-corrupted and background-perturbed inputs to construct auxiliary rewards for trajectories generated from the original image. These region-specific contrasts distinguish sensitivity to evidence corruption from robustness to background perturbations, complementing the visual-token evidence supervision in Stage I. Appendix A provides a detailed discussion.

5 Conclusion

We presented VISER, a two-stage post-training framework that promotes visual evidence use within standard single-turn reasoning. Stage I combines visual-token evidence supervision with answer generation, while Stage II uses reinforcement rewards from region-specific image interventions. VISER achieves the strongest overall performance among the compared 7B methods across five image VQA benchmarks, with near-base-model latency. By confining additional visual supervision to training, VISER retains standard autoregressive decoding without inference-time cropping, tool calls, or additional latent reasoning tokens.

Limitations and future work. VISER also improves over the base model on high-resolution benchmarks, although it remains behind leading explicit visual grounding methods in these evaluations. Improving fine-grained visual understanding under this inference setting therefore remains an open challenge. The framework also relies on evidence-region annotations during training. These limitations motivate future work on automatic evidence discovery, weaker forms of visual supervision, and more effective representations of fine-grained evidence.

REFERENCES

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. *arXiv preprint arXiv:2502.13923*, 2025.

Yishuo Cai, Jiahui Liu, Yuanxin Liu, Haobo Deng, Linli Yao, Yuhao Zheng, Kun Ouyang, Zhimo Li, Ziyue Wang, Xu Sun, et al. Thinking without images: Internalizing visual manipulation with on-policy self-distillation. *arXiv preprint arXiv:2606.08719*, 2026.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. Are we on the right way for evaluating large vision-language models? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=evP9mxNNxJ.

Long Chen, Xin Yan, Jun Xiao, Hanwang Zhang, Shiliang Pu, and Yueting Zhuang. Counterfactual samples synthesizing for robust visual question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10800–10809, 2020a.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020b.

Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, et al. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 11198–11201, 2024.

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*, pages 148–166. Springer, 2024.

Grok. Grok-1.5 vision preview. https://x.ai/news/grok-1.5v, April 2024. xAI News. Accessed: 2026-04-15.

Jack Hong, Chenxiao Zhao, Weiheng Lu, ChengLIn Zhu, Guohai Xu, and XingYu. Deepeyesv2: Toward agentic multimodal model. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=yDKawwfJ5O.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Ruina Hu, Chen Wang, Lai Wei, Jionghao Bai, Bin Yu, Weiran Huang, Kai Wang, and Yue Wang. Attend to evidence: Evidence-anchored spatial attention supervision for multimodal rlvr. *arXiv preprint arXiv:2605.30912*, 2026.

Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. Opera: Alleviating hallucination in multi-modal large language models via over-trust penalty and retrospection-allocation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13418–13427. IEEE, 2024.

Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13872–13882. IEEE, 2024.

Bangzheng Li, Ximeng Sun, Jiang Liu, Ze Wang, Jialian Wu, Xiaodong Yu, Emad Barsoum, Muhao Chen, and Zicheng Liu. Latent visual reasoning. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=j84WR5ORsC.

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*, pages 33457–33466, 2026b.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023. URL https://openreview.net/forum?id=xozJw0kZXF.

Zejun Li, Yingxiu Zhao, Jiwen Zhang, Siyuan Wang, Yang Yao, Runzhou Zhao, Jun Song, Bo Zheng, and Zhongyu Wei. Mixture-of-visual-thoughts: Exploring context-adaptive reasoning mode selection for general visual reasoning. In *The Fourteenth International Conference on Learning Representations*, 2026c. URL https://openreview.net/forum?id=8qk6eUnvbH.

Xiangyan Liu, Jinjie Ni, Zijian Wu, Chao Du, Longxu Dou, Haonan Wang, Tianyu Pang, and Michael Qizhe Shieh. Noisyrollout: Reinforcing visual reasoning with data augmentation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=9zD2i7YRot.

Yulei Niu, Kaihua Tang, Hanwang Zhang, Zhiwu Lu, Xian-Sheng Hua, and Ji-Rong Wen. Counterfactual vqa: A cause-effect look at language bias. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12700–12710, 2021.

OpenAI. Openai-o3. https://openai.com/index/introducing-o3-and-o4-mini/, 2025. Accessed: 2026-04-07.

Zhiyu Pan, Yizheng Wu, Jiashen Hua, Junyi Feng, Shaotian Yan, Bing Deng, Zhiguo Cao, and Jieping Ye. Through the lens of contrast: Self-improving visual reasoning in VLMs. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=ZymCPON45y.

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, Qixiang Ye, and Furu Wei. Grounding multimodal large language models to the world. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=lLmqxkfSIw.

Gabriel Herbert Sarch, Snigdha Saha, Naitik Khandelwal, Ayush Jain, Michael J. Tarr, Aviral Kumar, and Katerina Fragkiadaki. Grounded reinforcement learning for visual reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=1amnhVRQ3l.

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. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024a. URL https://openreview.net/forum?id=aXeiCbMFFJ.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024b.

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.

Haochen Wang, Xiangtai Li, Zilong Huang, Anran Wang, Jiacong Wang, Tao Zhang, Jiani Zheng, Sule Bai, Zijian Kang, Jiashi Feng, Wang Zhuochen, and Zhaoxiang Zhang. Traceable evidence enhanced visual grounded reasoning: Evaluation and method. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=zwMpimSZEq.

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*, pages 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, pages 7907–7915, 2025.

Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13084–13094, 2024.

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*, pages 33510–33520, 2026.

Haotian Zhang, Haoxuan You, Philipp Dufter, Bowen Zhang, Chen Chen, Hong-You Chen, Tsu-Jui Fu, William Yang Wang, Shih-Fu Chang, Zhe Gan, and Yinfei Yang. Ferret-v2: An improved baseline for referring and grounding with large language models. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=EEPBOB2Xww.

YiFan Zhang, Huanyu Zhang, Haochen Tian, Chaoyou Fu, Shuangqing Zhang, Junfei Wu, Feng Li, Kun Wang, Qingsong Wen, Zhang Zhang, Liang Wang, and Rong Jin. MME-realworld: Could your multimodal LLM challenge high-resolution real-world scenarios that are difficult for humans? In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=k5VHHgsRbi.

Ziwei Zheng, Michael Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, Chao Shen, and XingYu. Deepeyes: Incentivizing "thinking with images" via reinforcement learning. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=xUyMXkI958.

APPENDIX

This appendix expands the related-work discussion in Appendix A, provides additional analyses of visual evidence use in Appendix B, and reports supplementary ablations and evaluations in Appendices C and D. Implementation and evaluation details appear in Appendix E, followed by limitations, broader impacts, and statements on data and AI use.

A EXTENDED RELATED WORK

Explicit Visual Grounding. Explicit grounding connects language with localized image content. KOSMOS-2 (Peng et al., 2024) and Ferret-v2 (Zhang et al., 2024) develop region-level referring and grounding capabilities in VLMs. For visual reasoning, ViGoRL (Sarch et al., 2025), DeepEyes (Zheng et al., 2026), and Pixel-Reasoner (Su et al., 2025) incorporate region localization, image operations, or visual tools to access relevant details. Explicit grounding can also operate within textual reasoning: TreeVGR (Wang et al., 2026a) supervises bounding-box generation and answer prediction without requiring cropped images to be reintroduced. These approaches expose localized visual evidence through different reasoning interfaces. VISER instead uses region annotations to supervise internal visual-token representations and construct training-time visual contrasts, without requiring coordinate generation or visual tool use at inference.

Latent Visual Reasoning. Latent visual reasoning introduces dedicated intermediate tokens for processing visual information. LVR (Li et al., 2026a), Mirage (Yang et al., 2026), and Monet (Wang et al., 2026b) propagate decoder hidden states as continuous latent tokens before or between text segments. LVR uses annotated regions to supervise latent states through reconstruction of the corresponding visual embeddings. Latent Implicit Visual Reasoning (Li et al., 2026b) appends learnable latent tokens and trains them through a visual information bottleneck without explicit intermediate visual targets. These methods differ in their supervision and computation, but all introduce latent positions beyond the initial image tokens. VISER applies binary evidence supervision at existing visual-token positions and retains standard autoregressive text generation without dedicated latent reasoning tokens.

Perturbation and Counterfactual Learning. Counterfactual learning has been explored to reduce language bias in VQA. CSS (Chen et al., 2020a) constructs training examples by masking critical image objects or question words, while CF-VQA (Niu et al., 2021) reduces language bias through counterfactual inference. For recent VLMs, NoisyRollout (Liu et al., 2025) mixes trajectories from original and distorted images to improve exploration during RL. VC-STaR (Pan et al., 2026) uses contrastive VQA pairs to refine reasoning traces for supervised fine-tuning. VISER uses annotated evidence regions to construct two distinct interventions on the same image: evidence corruption and background perturbation. Together with the original input, these conditions provide contrastive RL rewards that encourage sensitivity to evidence corruption and robustness to background perturbations.

VISER's combination of evidence prediction at existing visual-token positions and contrastive visual rewards also differs from the supervision used in EASE and Imagine-OPD. EASE (Hu et al., 2026) aligns response-to-image attention with evidence annotations during RL, while Imagine-OPD (Cai et al., 2026) distills a teacher conditioned on evidence crops and ground-truth answers into a student operating on the original image.

B ADDITIONAL ANALYSES OF VISUAL EVIDENCE USE

This section complements the main-text analysis with additional attention visualizations, the full intervention protocol, and per-corruption results. These analyses characterize attention allocation and prediction sensitivity to evidence and background interventions.

B.1 ATTENTION VISUALIZATIONS

The layer-wise statistics in Figure 3 describe how attention is distributed between annotated evidence and the remaining visual tokens. Relative to Base, Stage I increases the evidence-to-context attention ratio in 21 of 28 layers, including 13 of the 14 later layers. Full VISER retains a similar pattern.

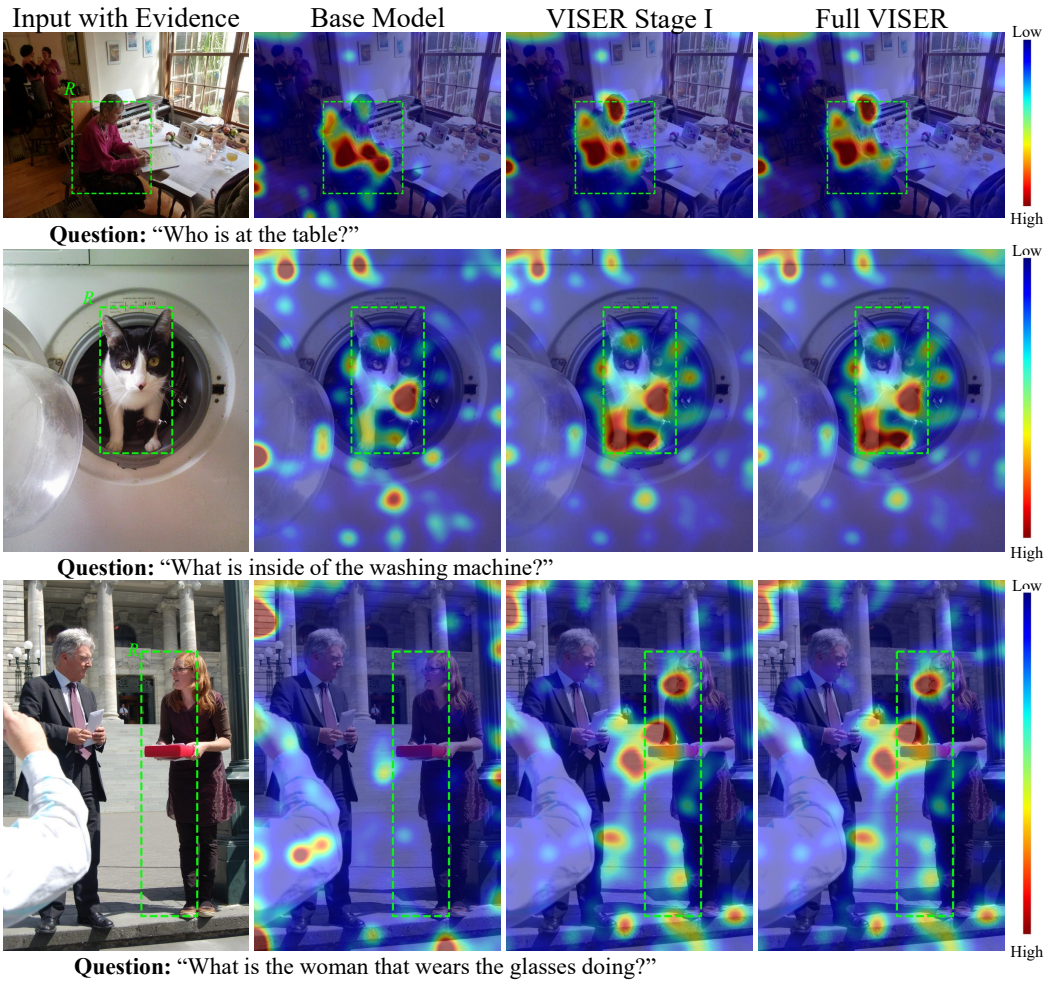


Figure 4: Additional attention visualizations across training stages. Each row shows an image-question example. The input image marks the annotated evidence region with a green dashed box; the remaining columns show attention maps for Base, Stage I, and full VISER. Warmer colors indicate higher attention.

Figure 4 provides additional qualitative examples for Base, Stage I, and full VISER. The maps illustrate changes in attention allocation around the annotated evidence regions across training stages. These visualizations complement the aggregate attention statistics.

B.2 Responses to Evidence and Background Interventions

Evaluation protocol and metrics. We evaluate prediction sensitivity on 500 held-out VisCoT examples. For each model, let $a_i^{\text{clean}} \in \{0, 1\}$ indicate whether its prediction for clean example i is correct, and define the clean-correct set $\mathcal{C} = \{i : a_i^{\text{clean}} = 1\}$. Let $a_i^{t,k}$ indicate correctness after intervention type t under setting k. The correctness drop rate (CDR) is

$$\text{CDR}_t^{(k)} = \frac{100}{|\mathcal{C}|} \sum_{i \in \mathcal{C}} \left(1 - a_i^{t,k}\right). \tag{8}$$

The clean-correct set is determined separately for each model, so comparisons across models need not involve identical subsets. The three intervention types are background perturbation ($t = \text{Pert}$), corruption of a non-evidence region with the same area as the evidence region ($t = \text{Match}$), and evidence corruption ($t = \text{Evi}$). The area-matched intervention provides a control for the amount of image area affected.

Table 6: Per-corruption evidence-selectivity results on 500 held-out VisCoT examples. Match and Evi denote area-matched background corruption and evidence corruption. CDRs are percentages; specificity scores are dimensionless. Clean accuracy and background-perturbation CDR are reported in Table 5.

Corruption	Model	$\mathrm{CDR}_{\mathrm{Match}} \downarrow$	$\mathrm{CDR}_{\mathrm{Evi}}$	SpecRatio↑	SpecGap↑
Blur $\sigma = 12$	Qwen2.5-VL	5.45	27.53	5.05	0.6693
	NoisyRollout	5.93	27.06	4.57	0.6406
	VISER Stage I	1.49	22.08	14.83	0.8737
	TreeVGR	4.04	24.75	6.12	0.7193
	VISER	**0.74**	22.22	**30.00**	**0.9355**
Blur $\sigma = 22$	Qwen2.5-VL	6.49	30.91	4.76	0.6528
	NoisyRollout	4.90	32.22	6.58	0.7361
	VISER Stage I	1.74	25.81	14.86	0.8739
	TreeVGR	3.54	25.25	7.14	0.7544
	VISER	**0.74**	22.96	**31.00**	**0.9375**
Pixel. $r = 0.05$	Qwen2.5-VL	7.27	37.40	5.14	0.6744
	NoisyRollout	6.44	37.11	5.76	0.7041
	VISER Stage I	1.99	32.26	16.25	0.8841
	TreeVGR	3.54	33.84	9.57	0.8108
	VISER	**0.74**	29.14	**39.33**	**0.9504**
Pixel. $r = 0.025$	Qwen2.5-VL	6.75	35.58	5.27	0.6810
	NoisyRollout	5.67	34.02	6.00	0.7143
	VISER Stage I	1.24	28.29	22.80	0.9160
	TreeVGR	3.79	31.57	8.33	0.7857
	VISER	**0.49**	27.90	**56.50**	**0.9652**

Within each corruption setting, we summarize the contrast between evidence and matched-background sensitivity using

$$\mathrm{SpecRatio}^{(k)} = \frac{\mathrm{CDR}_{\mathrm{Evi}}^{(k)}}{\mathrm{CDR}_{\mathrm{Match}}^{(k)} + \epsilon}. \tag{9}$$

$$\mathrm{SpecGap}^{(k)} = \frac{\mathrm{CDR}_{\mathrm{Evi}}^{(k)} - \mathrm{CDR}_{\mathrm{Match}}^{(k)}}{\mathrm{CDR}_{\mathrm{Evi}}^{(k)} + \mathrm{CDR}_{\mathrm{Match}}^{(k)} + \epsilon}, \tag{10}$$

where $\epsilon = 10^{-6}$. These metrics describe relative sensitivity and are interpreted together with the underlying CDRs.

Corruption settings and aggregation. Table 6 reports four analysis settings: Gaussian blur with $\sigma \in \{12, 22\}$ and pixelation with downsampling factors $r \in \{0.05, 0.025\}$. The main-text summary in Table 5 uses arithmetic means of the corruption-dependent CDRs over these settings. SpecRatio and SpecGap are likewise computed within each setting before averaging; their averages therefore need not equal the corresponding functions of averaged CDRs. Clean-input accuracy and $\mathrm{CDR}_{\mathrm{Pert}}$ are independent of the evidence-corruption setting and are reported in the main-text table.

Results. VISER achieves the lowest matched-background CDR and the highest specificity scores across all four settings in Table 6. Compared with Stage I, it consistently reduces sensitivity to matched-background corruption, while evidence corruption continues to cause substantially larger correctness drops.

C Additional Ablations and Robustness Analyses

C.1 Effect of Background Perturbation

We isolate the contribution of the background-perturbed view by removing it while retaining the evidence-corrupted view. All variants start from the same Stage-I checkpoint. As shown in Table 7,

Table 7: Ablation of the background-perturbed view in Stage II. "Remove background view" also removes the ranking terms involving p^{pert}, while retaining the evidence-corruption reward and the original–corrupted ranking term. All variants start from the same Stage-I checkpoint. Avg. denotes the unweighted mean accuracy over the five benchmarks.

Variant	VStar	MMStar	RWQA	POPE	BLINK	Avg.
Task reward only	89.01	63.80	71.50	88.27	58.18	74.15
Remove background view	89.01	64.00	71.90	89.00	58.23	74.43
Full VISER	**89.53**	**64.40**	**72.29**	**89.22**	**58.44**	**74.78**

Table 8: Sensitivity to 20% random jitter in evidence-box location and scale. Jittered boxes are used for Stage-I evidence targets or Stage-II triplets, as indicated. Avg. is the mean accuracy over the five main benchmarks.

Setting	VStar	MMStar	RWQA	POPE	BLINK	Avg.
Stage I: evidence target construction						
Stage I, clean boxes	81.15	62.40	71.37	88.41	57.76	**72.22**
Stage I, jittered boxes	81.15	62.27	71.11	88.18	58.02	72.15
Stage II: triplet construction						
VISER, clean boxes	89.53	64.40	72.29	89.22	58.44	**74.78**
VISER, jittered boxes	88.48	64.00	71.90	89.38	58.44	74.44

retaining only the evidence-corrupted view already improves average accuracy over task-only RL. Adding the background-perturbed view further improves all five benchmarks, with an average gain of 0.35 percentage points (pp). This comparison supports its incremental contribution alongside evidence-corruption feedback.

C.2 Sensitivity to Evidence Annotation Quality

We assess sensitivity to moderate annotation noise by applying 20% random jitter to evidence-box location and scale. Each box center is shifted by up to 20% of its width and height, and its width and height are independently scaled within $[0.8, 1.2]$. The resulting box is clipped to the image boundary. We examine noise in Stage-I target construction and Stage-II triplet construction separately.

Table 8 shows that VISER maintains stable average performance under the tested box jitter. For Stage I, using jittered boxes to construct evidence targets reduces the average score from 72.22 to 72.15, a drop of only 0.07 pp. For Stage II, using jittered boxes to construct visual triplets reduces the average score from 74.78 to 74.44, a drop of 0.34 pp. These results indicate that VISER tolerates moderate errors in evidence-box location and scale.

C.3 Evidence-Head Placement

Table 9 compares four locations for the Stage-I evidence head. Layer indices are zero-based, with layer 27 denoting the final layer of the 7B multimodal backbone. Different layers obtain the highest score on different benchmarks, and the average scores are close. The final layer gives the highest average in this comparison and is used as the default placement.

D Additional Evaluation Results

D.1 Detailed BLINK Results

Table 10 expands the overall BLINK comparison into its individual task categories. VISER achieves the highest overall accuracy among the listed methods. Its largest gains over Base include Jigsaw and Counting, and it also improves several correspondence and perception categories. Performance is not uniform across tasks: Relative Reflectance, Spatial Relation, and Multi-view Reasoning remain below Base.

Table 9: Evidence-head placement in Stage I. Layer indices are zero-based. Avg. is the mean accuracy over the five benchmarks. Best scores are bold.

Layer	VStar	MMStar	RWQA	POPE	BLINK	Avg.
7	80.63	62.53	**71.90**	88.33	57.02	72.08
15	81.15	**63.13**	71.11	88.19	56.71	72.06
23	**81.68**	62.20	70.72	**88.60**	57.13	72.07
27	81.15	62.40	71.37	88.41	**57.76**	**72.22**

Table 10: Detailed BLINK accuracy (%). Art: Art Style; Count: Counting; Foren: Forensic Detection; Func: Functional Correspondence; IQ: IQ Test; Jig: Jigsaw; Multi: Multi-view Reasoning; ObjL: Object Localization; Depth: Relative Depth; Refl: Relative Reflectance; SemC: Semantic Correspondence; Spat: Spatial Relation; VisC: Visual Correspondence; VisSim: Visual Similarity. Best and second-best scores are bold and underlined, respectively.

Method	Overall	Art	Count	Foren	Func	IQ	Jig	Multi
Qwen2.5-VL	56.29	69.23	70.00	50.00	26.15	23.33	60.00	54.89
NoisyRollout	55.81	67.52	71.67	47.73	23.85	22.67	63.33	54.14
LVR	46.92	47.01	70.00	25.00	24.62	29.33	52.67	**57.89**
TreeVGR	53.50	67.52	70.83	42.42	26.15	28.00	62.67	49.62
VISER	**58.44**	**70.09**	**77.50**	**53.03**	**28.46**	**30.67**	**70.67**	54.14

Method	ObjL	Depth	Refl	SemC	Spat	VisC	VisSim
Qwen2.5-VL	53.28	81.45	**39.55**	34.53	**88.81**	55.23	87.41
NoisyRollout	**54.92**	79.84	38.06	34.53	**88.81**	53.49	86.67
LVR	51.64	77.42	38.81	29.50	87.41	27.33	47.41
TreeVGR	51.64	76.61	29.10	34.53	79.72	50.58	85.19
VISER	**54.92**	**83.87**	36.57	**35.25**	81.12	**58.72**	**88.15**

D.2 VisCoT Answer-Accuracy Evaluation

We additionally evaluate answer accuracy on the 9,855-example VisCoT validation set (Shao et al., 2024a). Since VisCoT also supplies training data, we report this evaluation as a supplement to the five main benchmarks. VISER achieves 55.00% accuracy, exceeding Base and TreeVGR by 5.62 and 3.25 pp, respectively (Table 11). This evaluation is distinct from the intervention analysis on 500 held-out examples in Appendix B.2.

E Implementation and Evaluation Details

E.1 Training Data and Models

We evaluate VISER on Qwen2.5-VL-7B-Instruct and its 3B counterpart (Bai et al., 2025). Training primarily uses VisCoT (Shao et al., 2024a), with TreeVGR-RL-37K (Wang et al., 2026a) additionally used in Stage II. For Stage I, we remove the VisCoT dialogue turn that explicitly asks for bounding-box coordinates and retain the evidence annotations for constructing visual-token supervision. The 3B model uses the same training configuration and hyperparameters as the 7B model in both stages. The reported training times refer to the 7B model.

E.2 Stage-I Evidence Targets and Training

Token-level targets. After spatial merging, the visual tokens form a grid of size $h_g \times w_g$. We project the normalized evidence-box coordinates onto this grid to construct a binary evidence target $\mathbf{m} \in \{0,1\}^{N_v}$, where $N_v = h_g w_g$. The target map follows the same spatial layout as the visual tokens. A linear evidence head predicts the binary labels from the final backbone layer at visual-token positions. The evidence annotations provide supervision during training and are not required at inference.

Table 11: Answer accuracy on the VisCoT validation set (9,855 examples). Best and second-best scores are bold and underlined, respectively.

Method	Accuracy (%)
Qwen2.5-VL-7B	49.38
NoisyRollout (Liu et al., 2025)	35.16
LVR (Li et al., 2026a)	50.48
Pixel-Reasoner (Su et al., 2025)	45.15
ViGoRL (Sarch et al., 2025)	43.84
DeepEyes (Zheng et al., 2026)	45.21
DeepEyesV2 (Hong et al., 2026)	40.62
TreeVGR (Wang et al., 2026a)	<u>51.75</u>
VISER	**55.00**
Gain over Base (pp)	+5.62

Optimization. Stage I jointly optimizes answer generation and binary evidence prediction, with evidence-loss weight $\lambda_{\mathrm{vg}} = 0.003$. We apply LoRA (Hu et al., 2022) to the language model with rank 16, $\alpha = 32$, and dropout 0.01, while keeping the base model weights frozen. The evidence head is trained together with the adapters. We use AdamW with learning rate 5×10^{-5}, weight decay 0.1, warmup ratio 0.05, a cosine learning-rate schedule, and global batch size 4. Training runs for 8,000 steps with DeepSpeed ZeRO-3 and gradient checkpointing, taking approximately four hours on four GPUs. The LoRA weights are merged into the backbone before Stage II.

E.3 Stage-II Visual Triplet Construction

For each image-question pair (I, q) with an evidence box R, we prepare an evidence-corrupted image I_{cor} and a background-perturbed image I_{pert}. The region-processing procedure and candidate operators are specified below.

Region expansion and blending. We convert R to pixel coordinates, expand it around its center by a factor of 1.3, enforce a minimum side length of 32 pixels, and clip it to the image boundary. This expanded region is shared by the corruption and perturbation operators. We construct a binary mask for the expanded box and smooth it with a Gaussian filter of $\sigma = 15$, then clip the result to $[0, 1]$. The resulting soft mask blends corrupted content into the evidence region; its complement concentrates background perturbations outside that region.

Evidence corruption. We prepare four candidates: Gaussian blur with $\sigma \in \{20, 22\}$ and pixelation with downsampling factors $r \in \{0.025, 0.05\}$. For pixelation, the evidence patch is downsampled and then restored to its original extent using nearest-neighbor interpolation. These operations strongly degrade the annotated evidence region.

Background perturbation. We use the complementary soft mask to apply mild Gaussian blur to the background. The four candidates use strengths $\sigma \in \{1.2, 1.4, 1.6, 1.8\}$. These variants introduce mild changes to the surrounding context.

Candidate sampling. During training, one evidence-corrupted candidate and one background-perturbed candidate are randomly sampled and paired with the original image. The four candidates per branch provide variation in corruption type or strength while keeping the candidate pool small.

E.4 Stage-II Optimization and Reward Evaluation

Figure 5 summarizes the Stage-II training pipeline. The policy generates trajectories from the original image, while the evidence-corrupted and background-perturbed branches provide auxiliary evaluations for reward construction. The resulting rewards guide policy optimization on original-image trajectories.

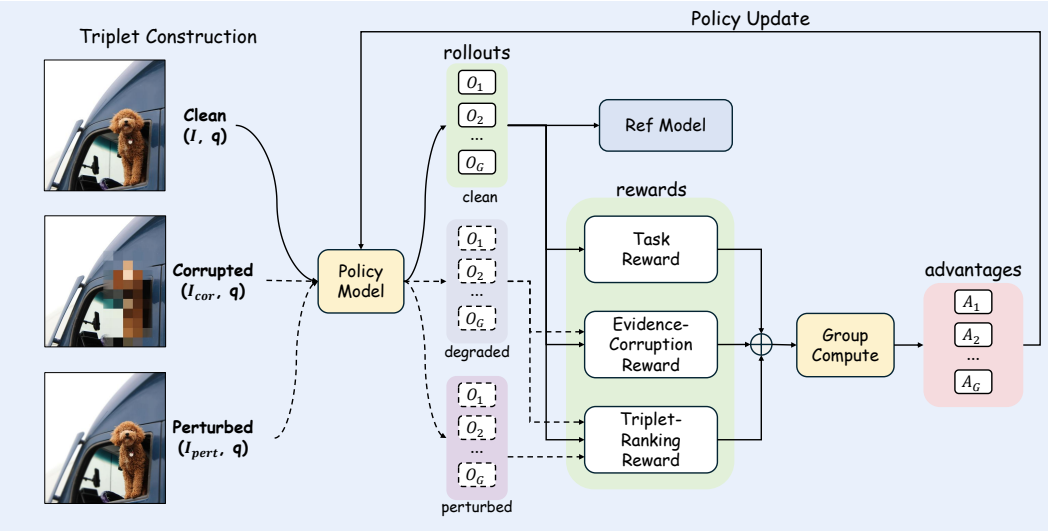


Figure 5: Stage-II GRPO training with visual triplets. Original-image trajectories form the policy-optimization targets. Evidence-corrupted and background-perturbed inputs provide auxiliary feedback for reward evaluation. The combined rewards determine group-relative advantages for the policy update.

Policy optimization. We use the total rewards and group-relative advantages defined in Eqs. 6 and 7. Group filtering evaluates the reward with $w_i = 1$, so that likelihood weighting alone does not make a group eligible for optimization. For retained groups, advantages are computed from the final rewards, including likelihood weighting, and each trajectory-level advantage is assigned to all valid generated tokens in that response.

Let π_{old} denote the sampling policy and π_θ the policy being optimized. For token $y_{i,t}$ in an original-image trajectory y_i, the importance ratio is

$$\rho_{i,t}(\theta) = \frac{\pi_\theta(y_{i,t} \mid I, q, y_{i,<t})}{\pi_{\text{old}}(y_{i,t} \mid I, q, y_{i,<t})}. \tag{11}$$

We maximize the clipped GRPO objective (Shao et al., 2024b):

$$\begin{aligned}\mathcal{J}_{\text{II}}(\theta) = \Big\langle \min\Big(\rho_{i,t}(\theta)\hat{A}_i,\ \text{clip}\big(\rho_{i,t}(\theta), 1-\epsilon, 1+\epsilon\big)\hat{A}_i\Big)\Big\rangle_{\text{valid}} \\ - \beta_{\text{KL}}\mathcal{L}_{\text{KL}} + \beta_{\text{ent}}\mathcal{H}.\end{aligned} \tag{12}$$

Here, $\langle\cdot\rangle_{\text{valid}}$ denotes aggregation over valid generated tokens in retained original-image trajectories, and ϵ is the policy-ratio clipping parameter. The term $\mathcal{L}_{\text{KL}}$ regularizes the policy relative to a reference policy, while $\mathcal{H}$ denotes the entropy of the current policy's generated-token distributions. The sampled rewards and advantages are treated as fixed during each policy update.

All original-image trajectories in retained groups participate in optimization, including incorrect responses. The correctness gates restrict the contrastive reward adjustments to correct responses, but do not exclude incorrect responses from the group-relative policy update. The intervention branches provide reward statistics only and do not directly contribute policy-gradient terms.

Training configuration. Training starts from the merged Stage-I checkpoint. We freeze the visual encoder and embedding layers and update the remaining backbone parameters. We use a training batch size of 128, a PPO mini-batch size of 16, four rollouts per prompt, and a learning rate of 5×10^{-7}. The KL regularization coefficient is $\beta_{\text{KL}} = 0.001$, using the low-variance formulation, and the entropy coefficient is $\beta_{\text{ent}} = 0.001$. The reward weights are $\lambda_{\text{cor}} = 0.3$ and $\lambda_{\text{rank}} = 0.1$. The ranking margins are $(m_{\text{oc}}, m_{\text{pc}}, m_{\text{op}}) = (0.5, 0.3, 0.1)$. Answer equivalence is evaluated by Qwen3-1.7B (Appendix E.6). Stage II takes approximately 20 hours on eight GPUs.

E.5 Benchmark Descriptions

Main benchmarks. VStar (Wu and Xie, 2024) evaluates fine-grained visual understanding in high-resolution, visually complex scenes. MMStar (Chen et al., 2024) evaluates visually indispensable perception and reasoning while reducing language-only shortcuts. RealWorldQA (RWQA) (Grok, 2024) focuses on visual understanding and spatial reasoning in real-world scenes. POPE (Li et al., 2023) evaluates object hallucination through questions about whether objects are present. BLINK (Fu et al., 2024) covers diverse perception and reasoning skills, including spatial relations, visual correspondence, and multi-view understanding. Detailed BLINK results are provided in Appendix D.1.

High-resolution benchmarks. We evaluate single-instance and cross-instance perception using HR-Bench (Wang et al., 2025), which covers attributes, text, maps, charts, and spatial relationships. We evaluate its 4K and 8K versions, abbreviated as HR-4K and HR-8K. MME-RealWorld (Zhang et al., 2025) covers OCR in the wild, remote sensing, diagrams and tables, monitoring, and autonomous driving. We use its official Lite subset with 1,919 questions and at most 50 examples per task, abbreviated as MME-RW in the result tables.

VisCoT. VisCoT (Shao et al., 2024a) provides VQA examples with visual evidence and reasoning annotations. Its additional answer-accuracy results and intervention analysis are reported separately in Appendices D.2 and B.2.

E.6 LLM Judge for Correctness Evaluation

We provide Qwen3-1.7B with the problem, the model prediction, and the reference answer. The judge is instructed to assess answer equivalence without solving the original problem. The same prompt is used for original, evidence-corrupted, and background-perturbed inputs to keep the correctness criterion consistent across the three conditions. The system prompt is reproduced below.

> **System prompt.**
> You are an expert in verifying whether two answers are equivalent.
> Your input is a problem and two answers: Answer 1 (**Model Predicted Answer**) and Answer 2 (**Ground Truth Answer**). You need to evaluate the model's predicted answer against the ground-truth answer.
> Your task is to determine whether the two answers are equivalent, without attempting to solve the original problem. Compare the answers to verify whether they represent identical values or meanings, even when written in different forms or notations.
> Your output should be only: `[[YES]]` if the answers are equivalent, or `[[NO]]` if they are not.

F Limitations and Broader Impacts

F.1 Limitations and Future Directions

VISER relies on evidence-region annotations during training. The box-jitter results (Appendix C.2) show limited sensitivity to moderate changes in box location and scale. Reducing the need for these annotations through weaker supervision or automatically discovered evidence is a promising direction for future work.

Our experiments cover image VQA and visual reasoning with 7B and 3B backbones. Extending the evaluation to other model families, larger models, video reasoning, and interactive settings would help establish the broader applicability of the framework. Tasks requiring fine-grained inspection of very small regions in large images may still benefit from additional visual interactions at inference. Combining VISER with selective visual inspection is a potential direction for addressing such tasks.

Stage II uses an auxiliary LLM judge for answer-equivalence evaluation. This introduces a dependence on automated answer verification; improving its reliability could further strengthen the training signal. The analyses in Appendix B provide complementary views of visual evidence use through

attention allocation and responses to image interventions. Future work could explore more direct ways to characterize visual evidence use during answer generation.

F.2 Broader Impacts

Strengthening the use of visual evidence may help reduce unsupported answers in multimodal applications. VISER nevertheless retains the failure modes of its underlying VLM and can produce incorrect or misleading responses. Applications involving sensitive decisions require appropriate human oversight.

G Data Availability and Ethics Statement

Our experiments use previously released datasets, evidence annotations, and benchmarks, subject to their original licenses and usage terms. We do not collect new private images, personal data, or human-subject annotations. The visual variants used for Stage-II reward evaluation are derived from the original training images. Their construction does not introduce new manually assigned semantic labels.

H AI Use Statement

We used generative AI tools to improve the clarity and readability of the manuscript, suggest organizational structures, assist with figure preparation, discuss and refine research ideas, and assist with code development. We also used Qwen3-1.7B as an automatic answer-equivalence judge in our experiments, as described in Appendix E.6. All AI-assisted text, figures, and code were reviewed by the authors. The authors take full responsibility for the final manuscript, methodology, implementation, and reported results.