

Keeping Eyes on the Image: Dynamic Visual Attention Recalibration for Vision-Language Reasoning

Yuna Jeong^{1*}, Junhyeong Park^{1*}, Mingyu Kang¹, Hyein Seo²,
Keon Kim², Yong Suk Choi^{2†}

¹Department of Artificial Intelligence, Hanyang University, Seoul, Korea

²Department of Computer Science, Hanyang University, Seoul, Korea

{dbsdk, junhyeong820, alsrb15788, appleshi, doublek375, cys}@hanyang.ac.kr

Abstract

Large Vision-Language Models (LVLMs) achieve strong performance on various vision-language tasks, yet Chain-of-Thought (CoT) generation often weakens attention to visual tokens, degrading evidence-grounded reasoning. During the later stages of autoregressive decoding, models tend to increasingly rely on previously generated text rather than visual evidence, leading to plausible but visually inconsistent rationales. We propose DVAR (Dynamic Visual Attention Recalibration), a training-free method that selectively amplifies visually relevant tokens to mitigate visual attention decay during decoding. DVAR operates at three levels: (i) prompt level structure to encourage visually grounded reasoning; (ii) embedding level relevance weights derived from image-text alignment; and (iii) decoding level scheduling that adjusts recalibration by tracking shifts in visual attention. Without sensitive model-specific hyperparameter retuning, DVAR yields consistent performance improvements across six benchmarks on LLaVA and Qwen-VL models, and better preserves visual attention throughout decoding.

1 Introduction

Large Vision-Language Models (LVLMs) have achieved strong performance on a wide range of multimodal tasks, including image captioning, visual question answering, and document/chart understanding (Liu et al., 2024b; Bai et al., 2025; Li et al., 2023a; Lee et al., 2023). When prompted to generate Chain-of-Thought (CoT) reasoning, LVLMs can further produce intermediate reasoning steps in natural language, enabling step-by-step solutions to complex multimodal problems (Zhang et al., 2025; Chen et al., 2024b). However, in multimodal reasoning, the key challenge is not simply to produce longer rationales, but to ensure that

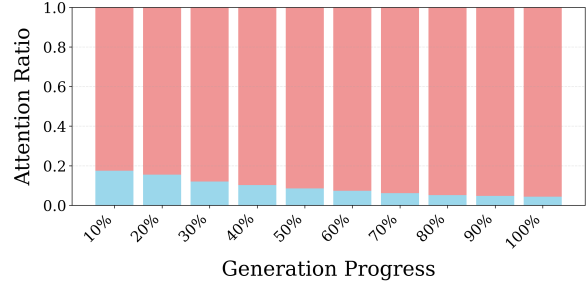


Figure 1: Fraction of attention mass assigned to visual (blue) and text (red) tokens for Qwen2.5-VL-7B, averaged over 500 M³CoT examples using “Let’s think step by step.” The horizontal axis shows normalized generation progress from 0% to 100%. Visual attention decreases as generation proceeds.

those rationales remain grounded in the visual input throughout generation (Man et al., 2025). If an explanation appears plausible yet fails to reflect the visual evidence, long-form generation can amplify errors rather than correct them.

This problem becomes more pronounced when reasoning unfolds over multiple dependent steps. Recent studies report that, as generation proceeds, LVLMs tend to rely increasingly on previously generated text instead of visual evidence. Attention allocated to visual tokens tends to decrease, while attention among text tokens becomes relatively dominant, a phenomenon known as visual attention decay (Tang et al., 2025). Figure 1 illustrates this trend. SPARC (Jung et al., 2025) further shows that visual attention weakens as generation steps accumulate, which can lead the model to rely on language priors and result in hallucinations and compounding mistakes (Yang et al., 2025; Li et al., 2023b; Wang et al., 2025a).

These observations suggest the need for decoding-time interventions that preserve visual grounding throughout reasoning. However, this is non-trivial for two reasons. First, attention is not a perfect proxy for relevance: prior work re-

*Equal contribution

†Corresponding author

ports a visual attention sink phenomenon, where visually irrelevant tokens can receive disproportionately high attention (Kang et al., 2025). Thus, naively amplifying visual tokens based only on observed attention may reinforce irrelevant regions. Second, fixed interventions are insufficient for long-form reasoning because the visual target can shift across reasoning steps and the required intervention strength can vary over time. Therefore, preserving visual grounding requires both spatial selection of visual patches relevant to the current reasoning step and temporal scheduling of recalibration strength over decoding.

To address these challenges, we propose DVAR (Dynamic Visual Attention Recalibration), a training-free, inference-time method that encourages persistent reference to visually relevant evidence throughout decoding. DVAR combines (i) vision-grounded prompting to structure the rationale around visual evidence and (ii) dynamic, selective visual attention recalibration guided by the current visual target and scheduled according to shifts in visual attention over time. By selectively amplifying visual tokens relevant to the current reasoning step rather than indiscriminately boosting all visual features, DVAR aims to preserve visual grounding while mitigating attention sinks.

Experiments on six benchmarks show that DVAR consistently improves performance across diverse LVLMs without additional training or model-specific retuning. Further analysis confirms that DVAR better preserves attention to visual tokens throughout long-form generation. Our contributions are summarized as follows:

- We introduce a selective visual attention recalibration strategy that determines token-specific amplification strengths for visual tokens, rather than uniformly applying a fixed scaling strength to all visual tokens.
- We make the attention recalibration time-adaptive by adjusting its strength across decoding steps, enabling temporal scheduling during long-form generation.
- We propose DVAR, a training-free, plug-and-play decoding-time intervention, and demonstrate consistent improvements across six benchmarks and multiple LVLMs (LLaVA-1.5/NeXT and Qwen2/2.5-VL), validating its effectiveness and generality.

2 Related Work

2.1 Large Vision-Language Models

Recent large vision-language models (LVLMs), including BLIP (Li et al., 2023a), LLaVA (Liu et al., 2024a,b, 2023), and Qwen-VL (Wang et al., 2024; Bai et al., 2025), typically encode images using a visual encoder such as a Vision Transformer (ViT) (Dosovitskiy et al., 2021) and project the resulting representations into the embedding space of a large language model (LLM). This unified representation enables strong performance across multimodal tasks such as image captioning, visual question answering, and document/chart understanding. However, as autoregressive generation becomes longer, the influence of visual information can gradually weaken (Yang et al., 2025), causing the model to rely more heavily on previously generated text and language priors. This issue is particularly important in long-form multimodal reasoning, where later reasoning steps still need to remain grounded in the original visual evidence. Our work addresses this problem at inference time by preserving and selectively reinforcing visually relevant information throughout decoding.

2.2 Reasoning in Large Vision-Language Models

Existing multimodal reasoning methods improve either the structure of intermediate reasoning or the use of visual evidence during decoding. DD-CoT (Zheng et al., 2023a), SCAFFOLD (Lei et al., 2025), and CCoT (Mitra et al., 2024) introduce sub-questions, coordinate information, or scene graphs to organize reasoning, but do not directly control visual evidence usage during generation. Other methods explicitly reinforce visual information. ICOT (Gao et al., 2025) reinserts attention-selected image patches between text segments, while FarSight (Tang et al., 2025) modifies causal masks to improve visual information propagation. IKOD (Yang et al., 2025) uses image-attention-guided key-value merging and collaborative decoding to mitigate visual attention degradation.

In contrast, DVAR selectively recalibrates visual tokens according to the current visual target and dynamically adjusts the intervention strength over decoding. By combining FOCUS-conditioned spatial selection with step-wise temporal scheduling, DVAR adapts both which visual evidence is strengthened and when stronger recalibration is applied.

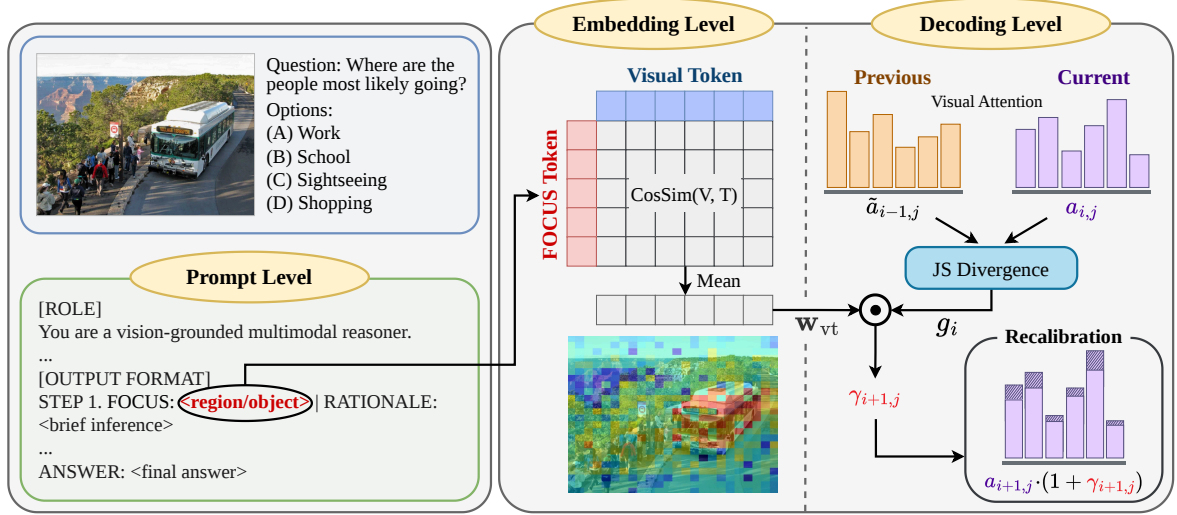


Figure 2: Overview of DVAR. Vision-grounded prompting structures the rationale, while focus-conditioned semantic relevance weights and an attention shift score guide step-wise visual attention recalibration during decoding.

3 Method

We propose DVAR, an inference-time intervention that mitigates visual grounding decay in autoregressive decoding by keeping the model anchored to currently relevant visual evidence. DVAR operates at three levels. At the Prompt level, we use a FOCUS-RATIONALE format that explicitly identifies what to look at and derives a short intermediate conclusion. At the Embedding level, we compute semantic relevance weights from the tokens in each generated FOCUS and update them whenever the rationale advances to a new FOCUS, allowing the grounding target to move during multi-step reasoning. At the Decoding level, we schedule step-wise intervention strength using an attention shift score that measures changes in visual attention across consecutive decoding steps, and softly amplify visual information accordingly. Figure 2 summarizes the overall pipeline.

3.1 Preliminaries

In an autoregressive LVLM, the input image x_v is transformed by a vision encoder $g(\cdot)$ into N_{img} patch features:

$$Z_v = g(x_v) \in \mathbb{R}^{N_{\text{img}} \times d_v}, \quad (1)$$

where d_v denotes the visual feature dimension. The features are projected to the language-model hidden dimension d through a linear projection $W \in \mathbb{R}^{d_v \times d}$:

$$H_v = Z_v W \in \mathbb{R}^{N_{\text{img}} \times d}. \quad (2)$$

The input text x_q is similarly embedded as

$$H_q \in \mathbb{R}^{N_q \times d}, \quad (3)$$

where N_q is the number of input text tokens. Thus, visual and text tokens share the same hidden dimension d .

At generation step i , the model predicts the next token y_i conditioned on the visual tokens H_v , input text tokens H_q , and previously generated tokens $y_{<i}$. Through transformer self-attention, the model computes the hidden state $\mathbf{h}_i \in \mathbb{R}^d$ at the current generation position.

At layer ℓ and head h , the query at generation position i and the key and value at sequence position t are defined as

$$\begin{aligned} \mathbf{q}_i^{(\ell,h)} &= \mathbf{W}_Q^{(\ell,h)} \mathbf{h}_i^{(\ell-1)}, \\ \mathbf{k}_t^{(\ell,h)} &= \mathbf{W}_K^{(\ell,h)} \mathbf{h}_t^{(\ell-1)}, \\ \mathbf{v}_t^{(\ell,h)} &= \mathbf{W}_V^{(\ell,h)} \mathbf{h}_t^{(\ell-1)}. \end{aligned} \quad (4)$$

Here, $\mathbf{W}_Q^{(\ell,h)}$, $\mathbf{W}_K^{(\ell,h)}$, and $\mathbf{W}_V^{(\ell,h)}$ are the corresponding projection matrices, and $\mathbf{h}_t^{(\ell-1)}$ denotes the representation at position t from the previous layer. In particular, $\mathbf{h}_i^{(\ell-1)}$ is the representation at the current generation position.

Let $\mathcal{V} = \{1, \dots, N_{\text{img}}\}$ denote the set of visual-token indices. We define the visual-only attention distribution by normalizing attention over $\mathcal{V}$. The attention weight assigned to visual token j is

$$\alpha_{i,j}^{(\ell,h)} = \text{softmax}_{m \in \mathcal{V}} \left(\frac{\mathbf{q}_i^{(\ell,h) \top} \mathbf{k}_m^{(\ell,h)}}{\sqrt{d_k}} \right)_j, \quad (5)$$

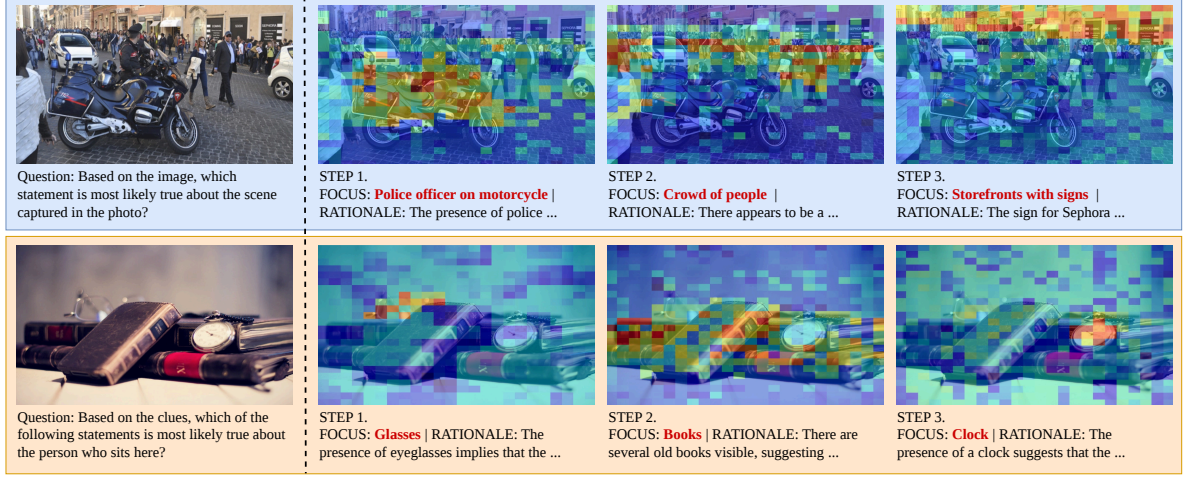


Figure 3: Visualization of the semantic relevance weights w_{vt} under different FOCUS phrases. Using LLaVA-1.5-7B on examples from M³CoT, we show how the relevance map changes according to the current FOCUS.

where d_k denotes the key/query dimension per head. We then average the visual attention across H attention heads:

$$a_{i,j}^{(\ell)} = \frac{1}{H} \sum_{h=1}^H \alpha_{i,j}^{(\ell,h)}, \quad (6)$$

where H is the number of attention heads.

Finally, the final-layer hidden state $\mathbf{h}_i$ is transformed by the language modeling head to obtain the next-token distribution:

$$p(y_i | x_v, x_q, y_{<i}) = \text{softmax}(\text{LMHead}(\mathbf{h}_i)). \quad (7)$$

3.2 Prompt Level: Vision-Grounded Prompting

In long CoT rationales, if the model does not clearly separate image-verifiable observations from subsequent inferences, autoregressive context can drive text-centric reasoning and cause the reasoning to drift away from the image evidence. To mitigate this, we design the prompt so that the model explicitly identifies a visual target at every reasoning step and then generates the corresponding rationale based on it. We adopt the FOCUS-RATIONALE prompting format not only to keep reasoning grounded in observable evidence, but also to explicitly state where to look at each step. This step-wise FOCUS phrase serves as a handle to update the model’s visual target as the reasoning progresses. The full template is provided in Appendix A.3.

(a) FOCUS: Explicit Spatial Referencing. FOCUS explicitly requires the model to identify the

image region or object that should serve as the visual target of the current reasoning step. Instead of referring to visual content implicitly, the model is required to name a concrete region, object, or visual element (e.g., an object on the table or the x-axis labels of a chart). This explicit spatial reference anchors the reasoning step to identifiable parts of the image and reduces reliance on purely linguistic context. In addition, the region or object specified by FOCUS is used to compute the semantic relevance weights w_{vt} , helping the recalibration coefficients place greater emphasis on the current visual target during reasoning.

(b) RATIONALE: Image-Grounded Reasoning. RATIONALE provides a brief observation or inference derived from the region specified in FOCUS. Each rationale is constrained to rely only on information visible in the referenced region, preventing unsupported speculation. By tying each reasoning step to its corresponding visual focus, the model maintains tighter alignment between visual evidence and textual reasoning throughout the generation process.

3.3 Embedding Level: Semantic Relevance Weights

In this section, we estimate the semantic relevance between image patches and the current FOCUS phrase during decoding, and define semantic relevance weights $w_{vt} \in \mathbb{R}^{N_{\text{img}}}$. The weights w_{vt} are computed once whenever a new FOCUS is produced and are then used as fixed weights for subsequent generation steps until the next FOCUS update.

The motivation for introducing this is to ensure that step-wise recalibration at decoding time operates on a spatially consistent basis. Because the FOCUS phrase is updated as the reasoning progresses, this design allows the recalibration to remain aligned with the changing intermediate visual target, rather than being tied to a single fixed region throughout decoding. To this end, we compute FOCUS-conditioned weights that reflect semantic alignment between image patches and the current visual target specified by FOCUS. Since the FOCUS phrase can refer to either a specific object/region or a broader visual context, the resulting relevance map should remain flexible rather than relying on hard masking. We obtain $\mathbf{w}_{\text{vt}}$ by normalizing the similarities between image patches and the current FOCUS with z-scoring and applying a softmax, yielding a smooth weighting map that assigns larger weights to regions more semantically related to the current FOCUS. Figure 3 illustrates how the semantic relevance weights $\mathbf{w}_{\text{vt}}$ evolve during the reasoning process for each sample.

Specifically, let $\mathbf{e}_{v,j} \in \mathbb{R}^d$ be the projected embedding of the j -th visual token, and let $\mathbf{e}_{f,k} \in \mathbb{R}^d$ be the embedding of the k -th token in the current FOCUS phrase, where N_{focus} denotes the number of tokens in the FOCUS. We denote their cosine similarity as $\text{CosSim}(\mathbf{e}_{v,j}, \mathbf{e}_{f,k})$. Then the raw relevance score of visual token j to the current FOCUS phrase is defined as

$$S_j = \frac{1}{N_{\text{focus}}} \sum_{k=1}^{N_{\text{focus}}} \text{CosSim}(\mathbf{e}_{v,j}, \mathbf{e}_{f,k}). \quad (8)$$

Let μ_S and σ_S denote the mean and standard deviation of the raw scores, respectively. The normalized z-score is computed as

$$z_j = \frac{S_j - \mu_S}{\sigma_S + \epsilon}. \quad (9)$$

Applying a softmax to the normalized scores, we obtain the semantic relevance weights:

$$\mathbf{w}_{\text{vt}} = \text{softmax}(\mathbf{z}), \quad w_{\text{vt},j} = (\mathbf{w}_{\text{vt}})_j. \quad (10)$$

In summary, the semantic relevance weights $\mathbf{w}_{\text{vt}}$ emphasize visual regions that are semantically related to the current FOCUS phrase. They are combined with the attention shift score g_i in the next section to guide selective attention recalibration during decoding.

3.4 Decoding Level: Step-Wise Dynamic Visual Attention Recalibration

As text generation proceeds, attention to relevant image regions can weaken, causing the generated reasoning to become less grounded in visual evidence. To mitigate this, we strengthen the contribution of relevant visual tokens at each decoding step. The key idea is to amplify the contribution of visual tokens in a soft, token-specific manner using semantic relevance weights $\mathbf{w}_{\text{vt}}$, where the amplification strength is modulated by the step-wise magnitude of changes in the visual attention pattern.

In this section, we use visual attention as an auxiliary signal for tracking changes in the model’s use of visual information. Accordingly, larger changes in the visual attention distribution indicate stronger shifts in visual information usage, which we use to dynamically adjust the recalibration strength.

Let $a_{i,j}$ denote the layer-wise averaged attention to visual token j at generation step i . To summarize the attention trend from previous steps, we define an exponential moving average (EMA) $\tilde{a}_{i-1,j}$ as

$$\tilde{a}_{i-1,j} = \beta \tilde{a}_{i-2,j} + (1 - \beta) a_{i-1,j}. \quad (11)$$

We normalize the current attention distribution and the EMA-based attention distribution at step i as

$$P_i(j) = \frac{a_{i,j}}{\sum_{m=1}^{N_{\text{img}}} a_{i,m}}, \quad Q_i(j) = \frac{\tilde{a}_{i-1,j}}{\sum_{m=1}^{N_{\text{img}}} \tilde{a}_{i-1,m}}. \quad (12)$$

We measure the difference between the two distributions using the Jensen-Shannon divergence (JSD) (Menéndez et al., 1997), defining the step-wise attention shift score as

$$g_i = \text{JSD}_{\log 2}(P_i, Q_i) \in [0, 1]. \quad (13)$$

Here, a larger g_i indicates that the visual attention distribution at step i has shifted more substantially from the EMA-based distribution, and accordingly a stronger visual attention recalibration is applied.

For each visual token j , we define the recalibration coefficient for the next decoding step by combining the semantic relevance weight $w_{\text{vt},j}$ with the attention shift score g_i computed at step i :

$$\gamma_{i+1,j} = w_{\text{vt},j} \cdot g_i, \quad \forall j \in \mathcal{V}. \quad (14)$$

Accordingly, recalibration at the attention level can be conceptually interpreted as

$$a_{i+1,j} \leftarrow a_{i+1,j} \cdot (1 + \gamma_{i+1,j}). \quad (15)$$

Rather than directly modifying the attention weights, we realize the recalibration in Eq. (15) by scaling the value vectors of visual tokens. That is, for each token j , we update at every layer and head as

$$\mathbf{v}_j^{(\ell,h)} \leftarrow \mathbf{v}_j^{(\ell,h)} \cdot (1 + \gamma_{i+1,j}). \quad (16)$$

We use value-vector scaling for both efficiency and stability of the existing attention pattern. Directly modifying attention weights can redistribute attention mass across visual and text tokens through attention normalization, whereas value-vector scaling strengthens the contribution of selected visual tokens while preserving the original attention weights and text-context reasoning trajectory. In addition, cached value vectors can be updated directly during decoding with negligible additional memory overhead. Since the attention output is formed as a weighted sum of value vectors, Eq. (16) increases the contribution of visual tokens with larger $w_{\text{vt},j}$ at decoding steps with larger g_i . We provide an empirical comparison with attention-weight scaling in Appendix B.2.

4 Experiments

4.1 Models

We apply the proposed method to four representative LVLMs: LLaVA-1.5 (Liu et al., 2024a), LLaVA-NeXT (Liu et al., 2024b), Qwen2-VL (Wang et al., 2024), and Qwen2.5-VL (Bai et al., 2025). In the main experiments, all models use a 7B-scale language backbone. LLaVA-1.5 and LLaVA-NeXT use Vicuna v1.5 as the language decoder (Chiang et al., 2023), whereas Qwen2-VL and Qwen2.5-VL follow Qwen-based multimodal architectures (Yang et al., 2024a,b) in their design. To examine whether DVAR remains effective at larger model scales, we additionally conduct experiments on LLaVA-NeXT-13B, Qwen2.5-VL-32B and InternVL3.5-14B (Wang et al., 2025b). In all settings, we apply the same recalibration module in a plug-and-play manner without additional training (training-free), and follow the official implementations for each model pipeline.

4.2 Benchmarks

We evaluate the proposed method on the following six benchmarks: M³CoT (Chen et al., 2024a), ScienceQA (Lu et al., 2022), LLaVA-Bench In-the-Wild (LLaVA-W) (Liu et al., 2023), MM-Bench (Liu et al., 2024c), MathVista (Lu et al.,

2024), and V* (Wu and Xie, 2024). For LLaVA-W, we use a GPT-5-based LLM-as-a-judge (Zheng et al., 2023b; Singh et al., 2026) to evaluate long-form responses. For the remaining benchmarks, we report the accuracy (Acc.) metric from the official evaluation scripts, and detailed settings such as input/output formats, metrics, and evaluation prompts are summarized in Appendix A.1.

4.3 Baselines

We compare DVAR against several baselines: No-CoT, DDCoT (Zheng et al., 2023a), SCAF-FOLD (Lei et al., 2025), CCoT (Mittra et al., 2024), and ICoT (Gao et al., 2025). We follow the official prompts/settings and publicly available implementations whenever possible. For fair comparison, all methods are evaluated under the same decoding conditions and maximum generation length. Detailed descriptions and implementation choices for each baseline are provided in Appendix A.2.

4.4 Implementation Details

Unless otherwise specified, the maximum generation length is set to 1024 tokens. The embedding-level semantic relevance weights $\mathbf{w}_{\text{vt}}$ are recomputed whenever a new FOCUS phrase is generated and remain fixed until the next FOCUS update. The decoding-level step-wise attention shift score g_i is updated at every decoding step based on changes in the attention distribution. We set the EMA coefficient to $\beta = 0.1$ and the numerical safeguard to $\epsilon = 10^{-10}$; an ablation over different β values is provided in Appendix B.1. We repeat all benchmark evaluations over five independent runs and report the mean and standard deviation. All experiments use a single NVIDIA RTX A5000 GPU, except for the scaling-up and cross-family experiments: LLaVA-NeXT-13B and InternVL3.5-14B use two GPUs each, while Qwen2.5-VL-32B uses four GPUs.

5 Results

5.1 Main Results

Table 1 and Table 2 summarize the main results on four LVLM models. Despite being applied purely at inference time without additional training, DVAR yields consistent improvements across models and benchmarks. In particular, DVAR achieves the best performance in all settings reported in the main tables. Across the five accuracy-based benchmarks, DVAR outperforms the strongest baseline

Model	Method	M ³ CoT Acc.↑	ScienceQA Acc.↑	LLaVA-W GPT-5-Eval.↑	MMBench Acc.↑	MathVista Acc.↑	V* Acc.↑
LLaVA-1.5-7B	No-CoT	33.56 ± 0.03	56.76 ± 0.04	31.62 ± 0.59	70.22 ± 0.06	21.82 ± 0.05	35.18 ± 0.23
	DDCoT	30.90 ± 0.08	56.93 ± 0.07	28.89 ± 1.04	71.28 ± 0.07	19.94 ± 0.25	35.81 ± 0.29
	SCAFFOLD	35.62 ± 0.04	61.17 ± 0.05	37.70 ± 0.68	69.96 ± 0.18	20.98 ± 0.12	38.85 ± 0.57
	CCoT	37.84 ± 0.07	61.04 ± 0.09	35.73 ± 0.59	73.42 ± 0.08	23.23 ± 0.19	42.09 ± 0.29
	ICoT	36.13 ± 0.08	61.85 ± 0.04	37.03 ± 0.43	71.41 ± 0.11	21.07 ± 0.24	42.83 ± 0.57
	DVAR	39.62 ± 0.08	63.50 ± 0.05	39.03 ± 1.02	74.20 ± 0.04	24.36 ± 0.15	48.38 ± 0.60
LLaVA-NeXT-7B	No-CoT	35.57 ± 0.04	59.70 ± 0.02	33.03 ± 1.35	73.91 ± 0.11	29.76 ± 0.06	45.76 ± 0.29
	DDCoT	33.96 ± 0.05	60.03 ± 0.05	30.72 ± 0.72	72.51 ± 0.12	21.69 ± 0.15	47.23 ± 0.57
	SCAFFOLD	31.86 ± 0.04	64.28 ± 0.04	40.47 ± 0.56	72.56 ± 0.11	22.09 ± 0.12	48.38 ± 0.47
	CCoT	37.92 ± 0.09	65.61 ± 0.07	38.50 ± 1.03	75.39 ± 0.13	30.17 ± 0.18	53.09 ± 0.79
	ICoT	38.57 ± 0.06	64.06 ± 0.07	40.36 ± 0.98	73.14 ± 0.08	28.42 ± 0.13	49.84 ± 0.68
	DVAR	41.25 ± 0.04	66.85 ± 0.04	43.61 ± 0.77	76.30 ± 0.08	31.71 ± 0.16	56.65 ± 0.57

Table 1: Results of our method and baselines on LLaVA-1.5-7B and LLaVA-NeXT-7B across multiple benchmarks.

Model	Method	M ³ CoT Acc.↑	ScienceQA Acc.↑	LLaVA-W GPT-5-Eval.↑	MMBench Acc.↑	MathVista Acc.↑	V* Acc.↑
Qwen2-VL-7B	No-CoT	50.26 ± 0.05	71.01 ± 0.09	40.55 ± 0.76	78.57 ± 0.17	54.72 ± 0.05	59.90 ± 0.47
	DDCoT	50.14 ± 0.08	68.29 ± 0.04	39.31 ± 0.50	79.03 ± 0.12	52.52 ± 0.10	57.28 ± 0.29
	SCAFFOLD	55.77 ± 0.05	67.76 ± 0.05	48.11 ± 0.88	81.31 ± 0.18	49.81 ± 0.05	60.52 ± 0.29
	CCoT	53.91 ± 0.07	74.79 ± 0.10	51.90 ± 0.95	80.69 ± 0.13	54.94 ± 0.19	60.94 ± 0.47
	ICoT	55.69 ± 0.08	78.15 ± 0.05	50.94 ± 0.89	80.21 ± 0.13	53.68 ± 0.10	60.31 ± 0.44
	DVAR	58.64 ± 0.11	81.29 ± 0.07	53.04 ± 0.79	82.09 ± 0.11	55.77 ± 0.15	62.09 ± 0.29
Qwen2.5-VL-7B	No-CoT	55.42 ± 0.07	75.85 ± 0.04	41.62 ± 0.67	83.19 ± 0.09	56.47 ± 0.06	61.78 ± 0.37
	DDCoT	53.47 ± 0.09	73.18 ± 0.07	42.07 ± 0.70	82.43 ± 0.08	53.70 ± 0.25	58.32 ± 0.60
	SCAFFOLD	57.23 ± 0.09	78.34 ± 0.04	47.06 ± 1.13	81.46 ± 0.12	52.71 ± 0.10	65.45 ± 0.37
	CCoT	57.48 ± 0.08	76.66 ± 0.12	52.15 ± 0.89	83.92 ± 0.07	54.42 ± 0.22	62.93 ± 0.57
	ICoT	58.02 ± 0.06	78.94 ± 0.09	52.54 ± 0.70	81.87 ± 0.04	57.93 ± 0.16	62.62 ± 0.47
	DVAR	60.71 ± 0.11	82.09 ± 0.05	54.95 ± 0.82	84.84 ± 0.08	60.17 ± 0.19	65.76 ± 0.95

Table 2: Results of our method and baselines on Qwen2-VL-7B and Qwen2.5-VL-7B across multiple benchmarks.

by 1.95 points on average and No-CoT by 5.34 points on average. On LLaVA-W, it further improves the GPT-5-Eval score by 2.01 points over the strongest baseline and by 10.95 points over No-CoT, averaged across models. These results support our hypothesis that selectively correcting visual grounding decay during generation improves multimodal reasoning performance.

Importantly, DVAR’s gains are not confined to a specific model family or benchmark, but are repeatedly observed across diverse architectures and evaluation settings. Although LLaVA and Qwen-VL differ in multimodal integration and language backbones, DVAR improves performance across all four models, suggesting robustness across architectures. Moreover, DVAR is designed to avoid sensitive model-specific hyperparameter retuning. This demonstrates practical plug-and-play effectiveness: DVAR can be integrated into LVLM pipelines without retraining or delicate tuning, while mitigating visual grounding decay during decoding.

The gains also generalize across benchmarks with different evaluation formats. First, M³CoT, ScienceQA, MMBench, and V* follow represen-

tative multiple-choice settings, where answer accuracy (Acc.) is the primary metric. On these tasks, DVAR generally improves accuracy, suggesting that better preservation of visual grounding helps reduce errors that arise when the model relies too heavily on textual context during reasoning. Notably, the consistent improvements on MMBench—which covers diverse perception and reasoning skills—indicate that DVAR is not limited to a narrow class of question types, but instead contributes more broadly by sustaining the use of visual evidence throughout the reasoning process.

MathVista involves a mixed format of multiple-choice and short-answer questions, where models often need to extract precise values from the image and then produce calculation results in text. In such cases, weakened visual grounding can corrupt the extracted quantities or destabilize the final answer format. DVAR nevertheless maintains clear gains on this benchmark, suggesting that the proposed method supports not only visual information extraction itself, but also the consistency of subsequent computation and textual generation. This further broadens its applicability to real-world multimodal

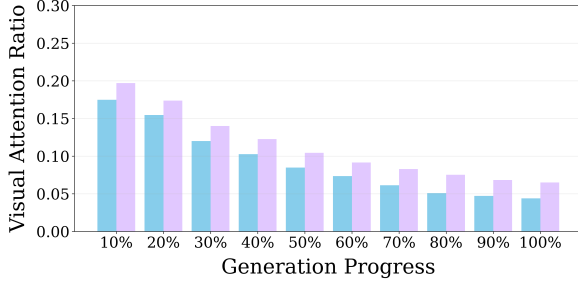


Figure 4: Baseline (blue) vs. DVAR (purple): visual attention ratio over generation progress on Qwen2.5-VL-7B, averaged over 500 M³CoT examples.

reasoning scenarios.

Finally, LLaVA-W is a subjective long-form benchmark evaluated with an LLM-as-a-judge protocol, where response quality depends not only on correctness but also on relevance to the question, coherence, and consistency with visual evidence. The fact that DVAR also improves scores on LLaVA-W suggests that it encourages the model to remain more faithfully grounded in visual evidence even when generating long responses.

Visual Attention Retention. To examine how DVAR affects the use of visual tokens during generation, we measure the visual attention ratio, defined as the fraction of attention mass assigned to visual tokens out of the total attention mass over all tokens. Figure 4 shows that DVAR maintains a higher visual attention ratio than the baseline throughout generation. Together with the task-performance gains and ablations, this supports DVAR’s ability to sustain visual information during long-form generation.

Scaling Up and Cross-Family Generalization. To examine whether DVAR generalizes across both model scales and LVLM architectures, we conduct additional experiments on M³CoT using LLaVA-NeXT-13B, Qwen2.5-VL-32B, and InternVL3.5-14B. The first two models extend the evaluation to larger variants of the LLaVA and Qwen-VL families used in our main experiments, while InternVL3.5-14B provides an additional model family with a different multimodal architecture. As summarized in Table 3, DVAR consistently outperforms the baselines across all three models. These results indicate that the effectiveness of DVAR is not restricted to 7B-scale models or to the LLaVA and Qwen-VL families, supporting its scalability and cross-family generalizability as a plug-and-play inference-time intervention.

Method	LLaVA -NeXT-13B	Qwen2.5 -VL-32B	InternVL 3.5-14B
No-CoT	38.07 $\pm$ 0.08	63.38 $\pm$ 0.02	61.79 $\pm$ 0.06
DDCoT	36.10 $\pm$ 0.08	63.90 $\pm$ 0.08	62.31 $\pm$ 0.07
SCAFFOLD	35.61 $\pm$ 0.04	66.28 $\pm$ 0.10	63.18 $\pm$ 0.09
CCoT	39.68 $\pm$ 0.02	68.33 $\pm$ 0.07	<u>65.11</u> $\pm$ 0.06
ICoT	<u>40.02</u> $\pm$ 0.10	<u>70.63</u> $\pm$ 0.07	64.78 $\pm$ 0.11
DVAR	42.17 $\pm$ 0.07	71.65 $\pm$ 0.17	68.28 $\pm$ 0.12

Table 3: Scaling-up and cross-family results on M³CoT.

Model	Components			M ³ CoT
	prompting	w_{vt}	g_i	
LLaVA -NeXT-7B	χ	χ	χ	34.36 $\pm$ 0.06
	χ	χ	✓	37.14 $\pm$ 0.09
	χ	✓	χ	35.94 $\pm$ 0.05
	χ	✓	✓	38.14 $\pm$ 0.06
	✓	χ	χ	35.74 $\pm$ 0.02
	✓	χ	✓	37.35 $\pm$ 0.09
	✓	✓	χ	38.52 $\pm$ 0.06
	✓	✓	✓	41.25 $\pm$ 0.04
Qwen2.5 -VL-7B	χ	χ	χ	56.70 $\pm$ 0.08
	χ	χ	✓	58.79 $\pm$ 0.06
	χ	✓	χ	57.06 $\pm$ 0.08
	χ	✓	✓	58.89 $\pm$ 0.09
	✓	χ	χ	56.91 $\pm$ 0.06
	✓	χ	✓	57.45 $\pm$ 0.07
	✓	✓	χ	58.51 $\pm$ 0.09
	✓	✓	✓	60.71 $\pm$ 0.11

Table 4: Ablation study of DVAR components under different combinations of FOCUS-RATIONALE prompting, w_{vt} , and g_i .

5.2 Ablation Studies

Setup and Ablation Variants. To examine the contribution of each component to DVAR’s performance, we conduct an ablation study on M³CoT. All experiments keep the same decoding settings as the main experiments. Table 4 summarizes the ablation variants by toggling three components: (i) vision-grounded prompting, (ii) semantic relevance weights w_{vt} , and (iii) the attention shift score g_i .

Concretely, in the prompt ablation, we replace the FOCUS-RATIONALE format with a generic reasoning prompt, “*Let’s think step by step.*” Since this setting does not provide explicit FOCUS phrases, the semantic relevance weights w_{vt} are computed using the embedding similarity between the user question and the image, and are kept fixed throughout decoding. In the w_{vt} ablation, we replace the semantic relevance weights with a uniform distribution over all visual tokens, i.e., $w_{vt,j} = 1/N_{img}$. Finally, in the g_i ablation used in

g	LLaVA-NeXT-7B	Qwen2.5-VL-7B
0.2	37.73 ± 0.12	56.72 ± 0.08
0.4	39.54 ± 0.15	57.54 ± 0.05
0.6	38.65 ± 0.09	58.84 ± 0.13
0.8	39.52 ± 0.10	58.28 ± 0.10
DVAR	41.25 ± 0.04	60.71 ± 0.11

Table 5: Constant vs. dynamic step-wise score. We replace the dynamic g_i with a step-independent constant g and compare it with DVAR.

Table 4, we replace the step-wise dynamic score with a step-independent fixed recalibration strength by setting $g_i=1.0$ for all steps. Table 5 separately studies several fixed constant g values to compare static scheduling with the dynamic g_i used in DVAR.

Effect of Vision-Grounded Prompting. Vision-grounded prompting helps structure reasoning around image-verifiable evidence. Without this component, the model is more likely to rely on textual context and continue with weakly grounded reasoning. Moreover, since $\mathbf{w}_{vt}$ is computed only once from the question-image similarity, it cannot adapt to shifting visual targets during multi-step reasoning. However, prompting alone is still insufficient: although it improves performance over a generic reasoning prompt, the gain remains relatively small. This indicates that DVAR’s improvements cannot be explained by prompting effects alone. While the prompt helps organize reasoning around visual evidence, substantial gains arise only when it is combined with decoding-time recalibration that selectively and dynamically strengthens visually relevant information.

Effect of Semantic Relevance Weights $\mathbf{w}_{vt}$. The semantic relevance weights $\mathbf{w}_{vt}$ provide the spatial selectivity needed for effective recalibration. Without $\mathbf{w}_{vt}$, attention-based recalibration amplifies all visual tokens according only to the step-wise attention shift score g_i , regardless of whether they are relevant to the current visual target specified by the FOCUS. As a result, even visual tokens unrelated to the current reasoning step can be over-amplified, which dilutes the intervention effect on truly relevant regions and makes it harder to maintain precise visual grounding as the target shifts across the reasoning process. Using only semantic relevance weights helps focus recalibration on visually relevant tokens, but the overall gain remains limited because the intervention strength

is not adjusted according to step-wise changes in visual attention.

Effect of Attention Shift Score g_i . The attention shift score g_i determines when stronger or weaker recalibration is needed during decoding. As generation steps accumulate, the required intervention strength varies across decoding steps; a fixed-strength strategy can therefore cause over- or under-intervention depending on the step. Using only g_i provides a temporal signal for when recalibration should be stronger, but without strong spatial constraints it can still amplify shifts in irrelevant regions, limiting its overall benefit.

The importance of dynamic g_i is further illustrated in Table 5, where we replace g_i with a constant g :

$$g_i = g, \quad g \in \{0.2, 0.4, 0.6, 0.8\}. \quad (17)$$

The static variant shows large performance fluctuations across different g values and consistently underperforms the dynamic g_i setting. This suggests that a single fixed strength is insufficient across diverse samples and decoding steps, while dynamic g_i more reliably mitigates visual attention decay through step-wise adjustment of recalibration. Overall, these ablations show that $\mathbf{w}_{vt}$ and g_i provide complementary spatial and temporal signals for recalibration.

6 Conclusion

This paper identifies a failure mode in CoT generation for large vision-language models, where visual grounding weakens as decoding progresses. To address this issue, we propose DVAR, a training-free inference-time method that preserves access to visually relevant evidence through selective and dynamic visual attention recalibration. DVAR combines vision-grounded prompting with FOCUS-conditioned spatial selection and step-wise temporal scheduling, enabling the model to better maintain grounding to the current visual target during decoding. Experiments on six benchmarks show that DVAR consistently improves performance over strong baselines, while better preserving attention to visual tokens throughout generation. These results suggest that decoding-time control over spatial selection and temporal scheduling is an effective and practical direction for improving evidence-grounded multimodal reasoning, with reduced reliance on model-specific retuning.

Limitations

Due to computational constraints, our experiments are limited to LVLMs of up to 32B parameters. While DVAR transfers well from 7B to 32B, we do not evaluate substantially larger models. Evaluating DVAR on larger LVLMs would provide further insight into its scalability and robustness and remains an important direction for future work.

DVAR also relies on generated FOCUS phrases to identify the current visual target. Although most FOCUS phrases are grounded in visible image evidence, an incorrect or ungrounded FOCUS can misdirect the semantic relevance weights and increase the likelihood of an incorrect prediction. Since DVAR uses FOCUS as a soft semantic anchor rather than a hard mask and updates it whenever a new FOCUS is generated, a single incorrect FOCUS does not necessarily determine the final prediction. Nevertheless, FOCUS quality remains an important limitation of DVAR. Detailed analyses of FOCUS reliability, failure cases, and type distribution are provided in Appendix B.5.

Acknowledgments

This work was supported by the Institute of Information and communications Technology Planning and evaluation (IITP) grant (No.RS-2025-25422680, No. RS-2020-II201373), and the National Research Foundation of Korea (NRF) grant (No. RS-2025-00520618) funded by the Korean Government (MSIT).

References

- Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*.
- Qiguang Chen, Libo Qin, Jin Zhang, Zhi Chen, Xiao Xu, and Wanxiang Che. 2024a. [M³CoT: A novel benchmark for multi-domain multi-step multi-modal chain-of-thought](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8199–8221, Bangkok, Thailand. Association for Computational Linguistics.
- Yangyi Chen, Karan Sikka, Michael Cogswell, Heng Ji, and Ajay Divakaran. 2024b. [Measuring and improving chain-of-thought reasoning in vision-language models](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 192–210, Mexico City, Mexico. Association for Computational Linguistics.
- Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. [Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality](#).
- Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. [An image is worth 16x16 words: Transformers for image recognition at scale](#). In *International Conference on Learning Representations*.
- Jun Gao, Yongqi Li, Ziqiang Cao, and Wenjie Li. 2025. Interleaved-modal chain-of-thought. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19520–19529.
- Mingi Jung, Saehyung Lee, Eunji Kim, and Sungroh Yoon. 2025. [Visual attention never fades: Selective progressive attention ReCalibration for detailed image captioning in multimodal large language models](#). In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 28527–28551. PMLR.
- Seil Kang, Jinyeong Kim, Junhyeok Kim, and Seong Jae Hwang. 2025. See what you are told: Visual attention sink in large multimodal models. In *International Conference on Learning Representations*.
- Kenton Lee, Mandar Joshi, Iulia Raluca Turc, Hexiang Hu, Fangyu Liu, Julian Martin Eisenschlos, Urvasi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. 2023. Pix2struct: Screen-shot parsing as pretraining for visual language understanding. In *International Conference on Machine Learning*, pages 18893–18912. PMLR.
- Xuanyu Lei, Zonghan Yang, Xinrui Chen, Peng Li, and Yang Liu. 2025. [Scaffolding coordinates to promote vision-language coordination in large multimodal models](#). In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 2886–2903, Abu Dhabi, UAE. Association for Computational Linguistics.
- Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023a. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR.
- Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. 2023b. [Evaluating object hallucination in large vision-language models](#). In *Proceedings of the 2023 Conference on Empirical Methods in*

- Natural Language Processing*, pages 292–305, Singapore. Association for Computational Linguistics.
- Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024a. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26296–26306.
- Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024b. [Llava-next: Improved reasoning, ocr, and world knowledge](#).
- Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.
- Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. 2024c. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pages 216–233. Springer.
- Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2024. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations*.
- Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. [Learn to explain: Multimodal reasoning via thought chains for science question answering](#). In *Advances in Neural Information Processing Systems*, volume 35, pages 2507–2521. Curran Associates, Inc.
- Yunze Man, De-An Huang, Guilin Liu, Shiwei Sheng, Shilong Liu, Liang-Yan Gui, Jan Kautz, Yu-Xiong Wang, and Zhiding Yu. 2025. Argus: Vision-centric reasoning with grounded chain-of-thought. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 14268–14280.
- María Luisa Menéndez, Julio Angel Pardo, Leandro Pardo, and María del C Pardo. 1997. The jensen-shannon divergence. *Journal of the Franklin Institute*, 334(2):307–318.
- Chancharik Mitra, Brandon Huang, Trevor Darrell, and Roei Herzig. 2024. Compositional chain-of-thought prompting for large multimodal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14420–14431.
- Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, and 466 others. 2026. [Openai gpt-5 system card](#). *Preprint*, arXiv:2601.03267.
- Feilong Tang, Chengzhi Liu, Zhongxing Xu, Ming Hu, Zile Huang, Haochen Xue, Ziyang Chen, Zelin Peng, Zhiwei Yang, Sijin Zhou, Wenxue Li, Yulong Li, Wenxuan Song, Shiyan Su, Wei Feng, Jionglong Su, Mingquan Lin, Yifan Peng, Xuelian Cheng, and 2 others. 2025. Seeing far and clearly: Mitigating hallucinations in mllms with attention causal decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 26147–26159.
- Chenxi Wang, Xiang Chen, Ningyu Zhang, Bozhong Tian, Haoming Xu, Shumin Deng, and Huajun Chen. 2025a. MLLM can see? dynamic correction decoding for hallucination mitigation. In *International Conference on Learning Representations*.
- Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.
- Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, Zhaokai Wang, Zhe Chen, Hongjie Zhang, Ganlin Yang, Haomin Wang, Qi Wei, Jinhui Yin, Wenhao Li, Erfei Cui, and 56 others. 2025b. [Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency](#). *Preprint*, arXiv:2508.18265.
- Penghao Wu and Saining Xie. 2024. V*: Guided visual search as a core mechanism in multimodal llms. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13084–13094. IEEE.
- An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 43 others. 2024a. [Qwen2 technical report](#). *Preprint*, arXiv:2407.10671.
- An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiayi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 23 others. 2024b. [Qwen2.5 technical report](#). *Preprint*, arXiv:2412.15115.
- Jiabing Yang, Chenhao Cui, Yiyang Zhou, Yixiang Chen, Peng Xia, Ying Wei, Tao Yu, Yan Huang, and Liang Wang. 2025. Ikod: Mitigating visual attention degradation in large vision-language models. *arXiv preprint arXiv:2508.03469*.
- Ruohong Zhang, Bowen Zhang, Yanghao Li, Haotian Zhang, Zhiqing Sun, Zhe Gan, Yinfei Yang, Ruoming Pang, and Yiming Yang. 2025. [Improve vision](#)

language model chain-of-thought reasoning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1631–1662, Vienna, Austria. Association for Computational Linguistics.

Ge Zheng, Bin Yang, Jiajin Tang, Hong-Yu Zhou, and Sibe Yang. 2023a. Ddcot: Duty-distinct chain-of-thought prompting for multimodal reasoning in language models. *Advances in Neural Information Processing Systems*, 36:5168–5191.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023b. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

A Experimental Details

A.1 Benchmarks and Metrics

This section provides an overview of the benchmarks and evaluation metrics used in our experiments.

M³CoT (Chen et al., 2024a) is a multimodal CoT benchmark that requires multi-step reasoning across multiple domains (science, mathematics, and commonsense). In this study, we use the evaluation set consisting of 2,318 samples, which includes diverse reasoning types subdivided into 17 topics. Following the official script, we report the accuracy (Acc.) of the final answers.

ScienceQA (Lu et al., 2022) is a multimodal multiple-choice QA benchmark for solving science problems. Among the 4,241 test samples, we use the 2,017 multimodal questions that include images (ScienceQA-IMG). ScienceQA hierarchically categorizes problems by topic, category, and skill level, and evaluates a wide range of scientific concepts and reasoning abilities. Following the official script, we report answer accuracy (Acc.).

LLaVA-W (Liu et al., 2023) is a benchmark for evaluating long-form visual question answering. It consists of 24 images covering diverse visual types such as scenes, memes, paintings, and sketches, along with 60 challenging questions. In this study, to quantitatively compare responses produced by different CoT strategies, we use a GPT-5-based LLM-as-a-judge evaluation, and follow the official LLaVA-W protocol for the evaluation prompt. During evaluation, GPT-5 receives both the ground-truth answer (GT) and the model response for each query, and scores the model response quality on a 1-10 scale. The final reported metric is relative performance (%), obtained by setting the GT score to 100% and normalizing the model response score assigned by GPT-5 as a percentage relative to the GT. The detailed GPT-5 evaluation prompt is provided in Figure 5.

MMBench (Liu et al., 2024c) is a multiple-choice benchmark for evaluating multimodal understanding and reasoning capabilities of AI models. It emphasizes scenarios that require fusion of visual and textual inputs, and organizes capabilities in a hierarchical taxonomy. Specifically, it defines perception and reasoning as Level 1 (L-1) categories, and expands them into six Level 2 (L-2) and twenty Level 3 (L-3) fine-grained dimensions. Following the official script, we report answer accuracy (Acc.).

MathVista (Lu et al., 2024) is a benchmark for solving problems by combining visual cues and mathematical reasoning. In this study, we use the English question-answer setting consisting of 936 samples. This dataset requires both mathematical reasoning and precise visual perception across diverse visual contexts such as charts, graphs, geometry, and puzzles. Following the official script, we report answer accuracy (Acc.).

V* (Wu and Xie, 2024) is a benchmark that evaluates search and reasoning abilities based on fine-grained visual cues such as small objects, attributes, and spatial relations. In this study, we use the evaluation set consisting of 191 samples, composed of queries that require identifying small cues and reasoning about relations. Following the official script, we report answer accuracy (Acc.).

A.2 Baseline Details

This section provides brief descriptions of the baselines and the implementation choices used in our experiments.

No-CoT is a setting where the model directly generates an answer to the input image and question without additional reasoning elicitation.

DDCoT (Zheng et al., 2023a) prompts an LLM to decompose the input question into a sequence of basic subquestions, then obtains answers from a VQA model for subquestions that require visual information and constructs the final output. Since the original paper uses ChatGPT as the LLM, for fair comparison in this study we configure the LVLM to also play the role of the LLM in DDCoT.

SCAFFOLD (Lei et al., 2025) overlays a coordinate grid on the input image to explicitly provide relative visual positions. During reasoning, the LVLM is encouraged to reference fine-grained visual cues in the image using this coordinate information. We insert coordinates into images using the released scripts and evaluate the model with the official prompts.

CCoT (Mitra et al., 2024) first generates a scene graph (SG) using the LVLM itself, and then generates a response by including the SG in the prompt. The SG is a JSON-like structured description that contains compositional information such as objects, attributes, and relations in the image. Following the original settings, we generate the SG and the answer separately using the official prompts.

ICoT (Gao et al., 2025) is an interleaved-modal CoT approach that strengthens visual evidence by interleaving selected image patches between text

[Context]
 {image caption}

[Question]
 {question text}

[Assistant 1]
 {reference answer}

[End of Assistant 1]

[Assistant 2]
 {model answer}

[End of Assistant 2]

[System]
 We would like to request your feedback on the performance of two AI assistants in response to the user question displayed above. The user asks the question on observing an image. For your reference, the visual content in the image is represented with a few sentences describing the image. Please rate the helpfulness, relevance, accuracy, level of details of their responses. Each assistant receives an overall score on a scale of 1 to 10, where a higher score indicates better overall performance. Please first output a single line containing only two values indicating the scores for Assistant 1 and 2, respectively. The two scores are separated by a space. In the subsequent line, please provide a comprehensive explanation of your evaluation, avoiding any potential bias and ensuring that the order in which the responses were presented does not affect your judgment.

Figure 5: Prompt templates used for GPT-5-based LLM-as-a-judge evaluation on LLaVA-W.

segments during reasoning. Specifically, it selects top- k image patches based on attention maps and inserts the selected patches between text segments to alternately provide visual and textual evidence. In our experiments, we adopt the one-shot scenario (rather than zero-shot) for ICoT.

A.3 Query Prompt

In this section, we present the template for the vision-grounded prompting used in this paper. Figure 6 includes the full template. In all experiments, the input query (question and options) is inserted into the same template, encouraging the model to explicitly specify, at each reasoning step, the attended visual region (FOCUS) and produce a short image-grounded rationale (RATIONALE). The rationale is constrained to rely only on observations or inferences directly supported by the region specified in FOCUS. This template can be applied without additional training, and when combined with dynamic recalibration at decoding time, it encourages reasoning that remains grounded in visual evidence even during long-form generation.

B Additional Results

B.1 Ablation on the EMA Coefficient β

In the final method, we use $\beta = 0.1$. To examine the sensitivity of the method to this choice, we vary β over the range from 0.05 to 0.3. As shown in Table 6, the overall performance remains stable across this range, with only modest variation for both LLaVA-NeXT-7B and Qwen2.5-VL-7B. This suggests that β is not a highly sensitive hyperparameter that requires precise tuning, but rather a smoothing coefficient for estimating the attention trend through EMA.

B.2 Value-Vector vs. Attention-Weight Scaling

We further compare value-vector scaling used in DVAR with directly scaling attention weights. Both variants use the same recalibration coefficients and decoding settings, differing only in where the recalibration is applied. Table 7 reports the results on V^* using LLaVA-NeXT-7B.

Attention-weight scaling also achieves compet-

[ROLE]
You are a vision-grounded multimodal reasoner.
Look at the image and the question and think step by step to derive the answer and write rationale.

[RULES]
1) Rationale length: max 6 steps; Keep each step to 1–2 sentences.
2) Each step MUST start with "FOCUS" naming the current region/object (e.g., the food on the plate, the x-axis labels)
3) "RATIONALE" must reason only from the image region/object specified in "FOCUS" and include only observations or inferences directly supported by it.
4) YOU MUST INCLUDE THE FINAL ANSWER IN YOUR OUTPUT.
5) If the question is multiple-choice, the ANSWER field must contain only the option letter.
6) If the question is short-answer, the ANSWER field must contain the answer directly and concisely.
7) If the question is open-ended, provide a detailed ANSWER while staying grounded in the image.

[OUTPUT FORMAT]
STEP 1. FOCUS: <region/object> | RATIONALE: <brief inference>
STEP 2. FOCUS: <...> | RATIONALE: <...>
...
ANSWER: <your final answer>

[NOW DO THE TASK]
Image: <provided by the system>
Question: "{question_text}"

Figure 6: Prompt template used in DVAR for vision-grounded prompting. The prompt enforces a FOCUS–RATIONALE format, where each step specifies the attended region/object (FOCUS) and a brief image-grounded rationale (RATIONALE), followed by the final answer.

Model	β	M ³ CoT
LLaVA-NeXT-7B	0.05	40.95 $\pm$ 0.08
	0.1 (Ours)	41.25 $\pm$ 0.04
	0.15	41.12 $\pm$ 0.20
	0.2	40.91 $\pm$ 0.05
	0.3	40.80 $\pm$ 0.11
Qwen2.5-VL-7B	0.05	60.39 $\pm$ 0.07
	0.1 (Ours)	60.71 $\pm$ 0.11
	0.15	60.53 $\pm$ 0.09
	0.2	59.84 $\pm$ 0.06
	0.3	60.29 $\pm$ 0.08

Table 6: Ablation study on the EMA coefficient β on M³CoT.

Scaling Method	V*
Attention-weight scaling	56.13 $\pm$ 0.44
Value-vector scaling (DVAR)	56.65 $\pm$ 0.57

Table 7: Comparison of attention-weight and value-vector scaling on V* using LLaVA-NeXT-7B.

itive performance, while value-vector scaling improves the mean score from 56.13 to 56.65. To-

gether with preserving the existing attention pattern and allowing efficient updates of cached value vectors, this result supports our choice of value-vector scaling in DVAR.

B.3 Stability under Extended CoT Generation

Since DVAR repeatedly scales visual value vectors during decoding, we additionally examine whether the intervention causes unstable hidden-state growth under extended generation. We randomly sample 200 M³CoT examples and evaluate LLaVA-NeXT-7B and Qwen2.5-VL-7B with the maximum generation length increased from 1,024 to 2,048 tokens. To encourage longer reasoning, we remove the original constraint on the maximum number and length of rationale steps and instead instruct the model to “Write as many rationales as possible.” Under this setting, the average generated response length is 793.77 tokens, with a maximum observed length of 1,196 tokens.

We track the hidden-state L_2 norm of the last generated token at 100-step intervals to examine whether repeated value-vector recalibration leads

Step	Qwen2.5-VL-7B		LLaVA-NeXT-7B	
	Norm	Δ	Norm	Δ
100	138.27	–	47.88	–
200	136.11	-2.16	51.67	+3.79
300	139.39	+3.28	44.72	-6.95
400	137.84	-1.55	44.23	-0.49
500	137.71	-0.13	49.78	+5.55
600	132.77	-4.94	45.91	-3.87
700	132.89	+0.12	47.52	+1.61
800	134.06	+1.17	49.41	+1.89
900	133.14	-0.92	43.83	-5.58
1000	131.86	-1.28	–	–
1100	134.81	+2.95	–	–

Table 8: Hidden-state L_2 norms under extended CoT generation. Δ denotes the change from the preceding 100-step checkpoint.

to cumulative norm growth. Table 8 summarizes the results.

The hidden-state norms show no persistent monotonic increase or abrupt growth within the observed decoding range. For Qwen2.5-VL-7B, the norm remains approximately within 132–139, while LLaVA-NeXT-7B fluctuates within approximately 44–52. The checkpoint-wise changes likewise show no sustained positive trend. These results suggest that DVAR does not exhibit hidden-state norm explosion under the extended generation setting examined here.

B.4 End-to-End Inference Latency

In this section, we report inference latency measured for each method. We measure the end-to-end inference time from input preprocessing to output generation. Specifically, the measurement includes (i) text tokenization, (ii) image preprocessing (e.g., resize/normalize and any method-specific additional preprocessing), (iii) token generation, and (iv) decoding and output post-processing/formatting. Accordingly, the cost incurred by additional steps required by each method (e.g., sub-question processing, coordinate overlay, scene graph generation, patch selection/insertion) is also included in the end-to-end time.

All methods are compared under the same decoding settings (maximum generation length of 1024). For stable measurement, we perform 3 warm-up runs and then report the mean and standard deviation of latency measured on 500 randomly sampled M³CoT examples using Qwen2.5-VL-7B (the experimental environment is the same as Section 4.4).

Table 9 summarizes the end-to-end latency of each method. Compared to No-CoT,

Method	Latency (ms)
No-CoT	1068.82 $\pm$ 573.57
DDCoT	17353.02 $\pm$ 6106.63
SCAFFOLD	9527.61 $\pm$ 3814.26
CCoT	12758.53 $\pm$ 3326.18
ICoT	10593.63 $\pm$ 4431.75
DVAR	10375.24 $\pm$ 5414.61

Table 9: End-to-end inference latency on Qwen2.5-VL-7B for 500 randomly sampled M³CoT examples (mean $\pm$ std, in ms) after 3 warm-up runs.

CoT/grounding-based methods generally incur additional latency due to rationale generation and additional processing steps. Nevertheless, DVAR achieves latency comparable to, or lower than, other baseline methods, demonstrating that our recalibration mechanism introduces minimal overhead while preserving efficient inference.

B.5 Analysis of FOCUS Generation

Since DVAR uses generated FOCUS phrases to define the current visual target, we further analyze their grounding reliability and semantic types. We randomly sample 200 examples from M³CoT and analyze the FOCUS phrases generated by LLaVA-NeXT-7B and Qwen2.5-VL-7B using GPT-5 as an image-aware judge. For each example, GPT-5 is given the input image, example ID, and the list of generated FOCUS phrases, and jointly predicts a binary grounding score and a semantic type for each FOCUS phrase. The exact evaluation prompt is provided in Figure 7.

FOCUS Grounding Reliability. For each generated FOCUS phrase, GPT-5 assigns a binary score of 1 if the referred object, region, or visual cue is identifiable in the image, and 0 otherwise. Mean Score is computed by first averaging the binary scores within each sample and then averaging them across all samples, while FOCUS Acc. is computed over all generated FOCUS phrases.

As shown in Table 10, 81.22% and 82.82% of generated FOCUS phrases are grounded in visible image evidence for LLaVA-NeXT-7B and Qwen2.5-VL-7B, respectively. We further evaluate task performance on samples containing at least one ungrounded FOCUS phrase. Compared with the overall task accuracy, performance decreases by 6.17 and 5.41 percentage points, respectively. These results indicate that incorrect FOCUS phrases can increase the likelihood of failure and represent an important limitation of DVAR.

You are evaluating generated FOCUS phrases for a vision-language reasoning method.

You will be given:

- an image
- an example id
- a list of generated FOCUS phrases

For each FOCUS phrase:

1. Assign a binary score:
 - 1: the FOCUS can be found in the image.
 - 0: the FOCUS cannot be found in the image or is not visually grounded.
2. Classify the semantic type of the FOCUS phrase using the numeric type labels below.

Type labels:

- 1 = object: a single concrete object, multiple objects, or a group of objects.
- 2 = region: a spatial area or part of the image.
- 3 = relation: a spatial, comparative, action, or interaction relation between entities.
- 4 = attribute: a visual property such as color, shape, size, texture, material, number, or state.
- 5 = text_chart_map_element: text or chart/map/diagram elements such as labels, axes, bars, or points.
- 6 = other: none of the above.

Classify the type independently of the score.

Scoring rule:

- num_focus = total number of FOCUS phrases.
- num_correct_focus = number of FOCUS phrases with score 1.
- focus_score = num_correct_focus / num_focus.
- If there are no FOCUS phrases, set num_focus = 0, num_correct_focus = 0, and focus_score = 0.0.

Figure 7: Exact GPT-5 prompt used for FOCUS grounding and semantic-type evaluation. GPT-5 receives the input image and generated FOCUS phrases, and returns the grounding score and semantic type of each FOCUS phrase.

Model	FOCUS Grounding				Task Performance		
	#FOCUS	Correct	Mean Score	FOCUS Acc. (%)	Overall Task Acc. (%)	Task Acc. w/ Ungrounded FOCUS (%)	Drop (pp)
LLaVA-NeXT-7B	740	601	0.81	81.22	44.50	38.33	-6.17
Qwen2.5-VL-7B	646	535	0.84	82.82	64.50	59.09	-5.41

Table 10: Analysis of FOCUS grounding quality and task performance on 200 randomly sampled M³CoT examples. Task Acc. w/ Ungrounded FOCUS denotes task accuracy on samples containing at least one ungrounded FOCUS phrase, and Drop denotes the decrease from overall task accuracy in percentage points (pp).

Qualitatively, Figure 8 presents both failure and recoverable cases involving ungrounded FOCUS phrases. In the failure case, an incorrect FOCUS misdirects subsequent reasoning and leads to an incorrect final answer, whereas in the recoverable case, the remaining grounded FOCUS phrases provide sufficient visual evidence to reach the correct answer despite the presence of an ungrounded FOCUS.

This robustness is related to the soft design of

DVAR. FOCUS does not crop the corresponding image region or apply a hard mask to visual tokens. Instead, it is used to compute the semantic relevance distribution w_{vt} , which serves as a soft semantic anchor and is recomputed whenever a new FOCUS is generated. Therefore, while DVAR remains sensitive to FOCUS errors, a single incorrect FOCUS is not fixed throughout the entire reasoning process.

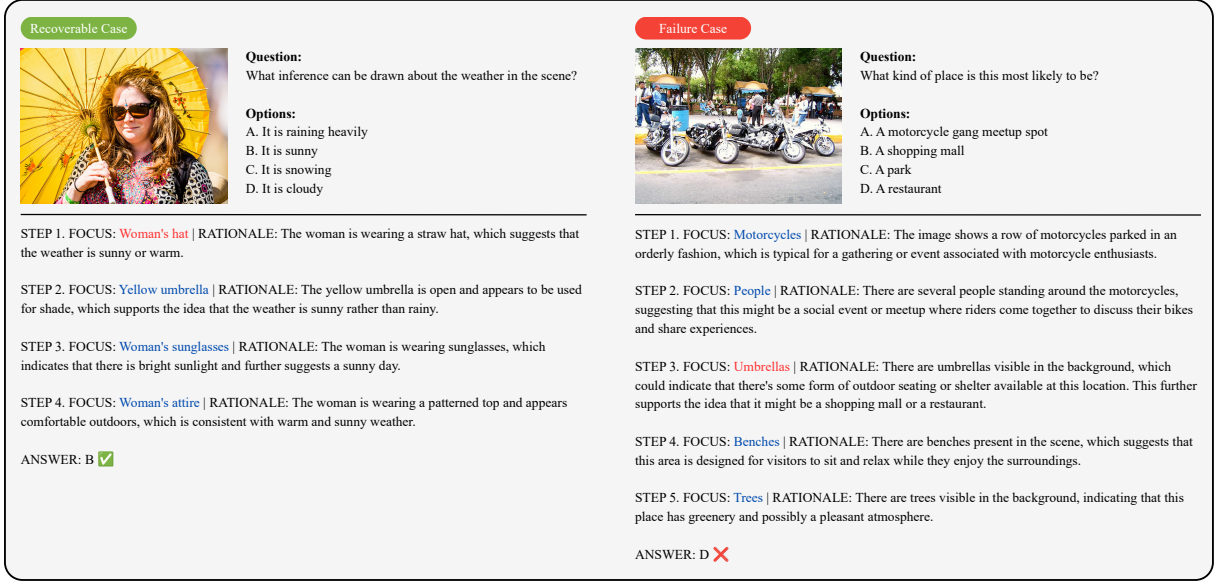


Figure 8: Examples of failure and recoverable cases involving ungrounded FOCUS phrases. In the failure case, an incorrect FOCUS misdirects the reasoning and leads to an incorrect final answer, whereas in the recoverable case, the remaining grounded FOCUS phrases provide sufficient visual evidence to reach the correct answer.

FOCUS Type Distribution. We further classify each generated FOCUS phrase into six semantic types: Object, Region, Relation, Attribute, Text/Chart/Map Element, and Other. Table 11 summarizes the resulting distribution.

Object and Region together account for 55.81% of FOCUS phrases in LLaVA-NeXT-7B and 60.06% in Qwen2.5-VL-7B, forming the largest group in both models. This is consistent with the intended role of FOCUS, which specifies a visually identifiable anchor for each reasoning step. Relation, Attribute, and Text/Chart/Map Element types are also observed, indicating that FOCUS is not restricted to simple object naming. More complex interpretations of relations between visual elements are further developed in the corresponding RATIONALE.

B.6 Full Results on MMBench

Table 12 reports the detailed results on MMBench across six Level-2 categories: Logical Reasoning (LR), Attribute Reasoning (AR), Relation Reasoning (RR), Fine Perception-Single (FP-S), Fine Perception-Cross (FP-C), and Coarse Perception (CP). Across all four backbones (LLaVA-1.5/NeXT and Qwen2/2.5), DVAR improves the overall score over the No-CoT baseline and remains competitive with strong CoT-based baselines. Improvements are also frequently observed in reasoning-oriented and fine-grained perception categories, suggesting that DVAR benefits both multi-step reasoning and

detailed visual understanding.

B.7 Additional Scaling-Up Results on V*

We further evaluate the larger and cross-family models on V*. As shown in Table 13, DVAR achieves the highest mean performance across LLaVA-NeXT-13B, Qwen2.5-VL-32B, and InternVL3.5-14B, providing additional evidence that its effectiveness extends across model scales and architectures.

B.8 Comparison with FarSight

We additionally compare DVAR with FarSight (Tang et al., 2025), a training-free inference-time method that mitigates visual information degradation during decoding. We evaluate both methods on M³CoT, MathVista, and V* using LLaVA-NeXT-7B and Qwen2.5-VL-7B.

As shown in Table 14, DVAR consistently matches or outperforms FarSight across all evaluated settings, with particularly clear gains on M³CoT and V*. On MathVista, DVAR improves Qwen2.5-VL-7B while showing comparable performance on LLaVA-NeXT-7B.

IKOD (Yang et al., 2025) is also conceptually related to DVAR in addressing visual attention degradation at inference time, but differs in its mechanism: IKOD employs image-attention-guided key-value merging and collaborative decoding, whereas DVAR uses FOCUS-conditioned spatial recalibration with step-wise temporal scheduling. However,

Model	Object	Region	Relation	Attribute	Text/Chart/Map Element	Other
LLaVA-NeXT-7B	347 (46.89%)	66 (8.92%)	123 (16.62%)	47 (6.35%)	114 (15.41%)	43 (5.81%)
Qwen2.5-VL-7B	303 (46.90%)	85 (13.16%)	85 (13.16%)	71 (10.99%)	28 (4.33%)	74 (11.46%)

Table 11: Distribution of semantic types of generated FOCUS phrases on 200 randomly sampled M³CoT examples.

Model	Method	LR $\uparrow$	AR $\uparrow$	RR $\uparrow$	FP-S $\uparrow$	FP-C $\uparrow$	CP $\uparrow$	Overall $\uparrow$
LLaVA-1.5-7B	No-CoT	46.46 $\pm$ 0.22	77.42 $\pm$ 0.16	60.63 $\pm$ 0.29	73.76 $\pm$ 0.15	61.13 $\pm$ 0.08	79.12 $\pm$ 0.18	70.22 $\pm$ 0.06
	DDCoT	45.39 $\pm$ 0.30	79.11 $\pm$ 0.18	61.21 $\pm$ 0.26	74.48 $\pm$ 0.15	62.66 $\pm$ 0.23	80.87 $\pm$ 0.13	71.28 $\pm$ 0.07
	SCAFFOLD	48.45 $\pm$ 0.28	72.99 $\pm$ 0.26	55.24 $\pm$ 0.33	75.03 $\pm$ 0.20	63.98 $\pm$ 0.38	79.66 $\pm$ 0.19	69.96 $\pm$ 0.18
	CCoT	49.27 $\pm$ 0.24	78.63 $\pm$ 0.19	67.01 $\pm$ 0.25	75.83 $\pm$ 0.17	67.35 $\pm$ 0.23	82.13 $\pm$ 0.15	73.42 $\pm$ 0.08
	ICoT	44.81 $\pm$ 0.37	77.77 $\pm$ 0.33	59.91 $\pm$ 0.34	73.79 $\pm$ 0.08	65.25 $\pm$ 0.21	82.42 $\pm$ 0.13	71.41 $\pm$ 0.11
	DVAR	52.52 $\pm$ 0.33	79.00 $\pm$ 0.16	70.29 $\pm$ 0.19	75.99 $\pm$ 0.05	68.14 $\pm$ 0.16	81.89 $\pm$ 0.12	74.20 $\pm$ 0.04
LLaVA-NeXT-7B	No-CoT	53.79 $\pm$ 0.33	74.91 $\pm$ 0.28	71.55 $\pm$ 0.20	75.77 $\pm$ 0.15	64.50 $\pm$ 0.16	84.33 $\pm$ 0.10	73.91 $\pm$ 0.11
	DDCoT	53.98 $\pm$ 0.28	74.74 $\pm$ 0.22	68.04 $\pm$ 0.19	72.96 $\pm$ 0.15	63.49 $\pm$ 0.21	83.64 $\pm$ 0.14	72.51 $\pm$ 0.12
	SCAFFOLD	49.08 $\pm$ 0.27	77.11 $\pm$ 0.18	68.54 $\pm$ 0.28	73.03 $\pm$ 0.21	63.34 $\pm$ 0.28	83.97 $\pm$ 0.12	72.56 $\pm$ 0.11
	CCoT	58.83 $\pm$ 0.33	77.51 $\pm$ 0.12	75.37 $\pm$ 0.30	76.20 $\pm$ 0.17	65.07 $\pm$ 0.24	84.35 $\pm$ 0.18	75.39 $\pm$ 0.13
	ICoT	50.97 $\pm$ 0.34	77.37 $\pm$ 0.16	68.94 $\pm$ 0.19	73.85 $\pm$ 0.15	64.80 $\pm$ 0.10	83.70 $\pm$ 0.08	73.14 $\pm$ 0.08
	DVAR	56.60 $\pm$ 0.40	79.31 $\pm$ 0.13	72.09 $\pm$ 0.19	78.02 $\pm$ 0.08	68.44 $\pm$ 0.08	85.43 $\pm$ 0.15	76.30 $\pm$ 0.08
Qwen2-VL-7B	No-CoT	63.20 $\pm$ 0.41	81.23 $\pm$ 0.12	77.80 $\pm$ 0.19	83.27 $\pm$ 0.13	68.48 $\pm$ 0.35	82.98 $\pm$ 0.26	78.57 $\pm$ 0.17
	DDCoT	61.84 $\pm$ 0.32	81.75 $\pm$ 0.16	76.72 $\pm$ 0.26	83.96 $\pm$ 0.17	69.64 $\pm$ 0.24	84.09 $\pm$ 0.15	79.03 $\pm$ 0.12
	SCAFFOLD	67.43 $\pm$ 0.27	82.43 $\pm$ 0.24	81.21 $\pm$ 0.47	87.23 $\pm$ 0.12	69.76 $\pm$ 0.31	85.30 $\pm$ 0.17	81.31 $\pm$ 0.18
	CCoT	67.72 $\pm$ 0.30	81.95 $\pm$ 0.24	82.25 $\pm$ 0.28	84.95 $\pm$ 0.12	69.68 $\pm$ 0.21	85.03 $\pm$ 0.18	80.69 $\pm$ 0.13
	ICoT	67.28 $\pm$ 0.32	83.09 $\pm$ 0.16	79.10 $\pm$ 0.28	84.46 $\pm$ 0.07	68.48 $\pm$ 0.23	84.96 $\pm$ 0.10	80.21 $\pm$ 0.13
	DVAR	70.10 $\pm$ 0.20	84.55 $\pm$ 0.18	81.89 $\pm$ 0.26	87.37 $\pm$ 0.13	69.57 $\pm$ 0.34	85.72 $\pm$ 0.16	82.09 $\pm$ 0.11
Qwen2.5-VL-7B	No-CoT	70.87 $\pm$ 0.51	84.95 $\pm$ 0.06	81.35 $\pm$ 0.32	87.18 $\pm$ 0.13	75.38 $\pm$ 0.28	87.09 $\pm$ 0.22	83.19 $\pm$ 0.09
	DDCoT	69.81 $\pm$ 0.33	82.75 $\pm$ 0.16	78.47 $\pm$ 0.19	86.28 $\pm$ 0.12	78.91 $\pm$ 0.21	86.26 $\pm$ 0.12	82.43 $\pm$ 0.08
	SCAFFOLD	69.81 $\pm$ 0.13	83.92 $\pm$ 0.22	79.96 $\pm$ 0.40	80.69 $\pm$ 0.17	78.09 $\pm$ 0.28	87.27 $\pm$ 0.20	81.46 $\pm$ 0.12
	CCoT	74.56 $\pm$ 0.20	82.03 $\pm$ 0.19	84.45 $\pm$ 0.29	87.69 $\pm$ 0.07	76.51 $\pm$ 0.16	88.08 $\pm$ 0.17	83.92 $\pm$ 0.07
	ICoT	61.75 $\pm$ 0.22	82.46 $\pm$ 0.16	83.73 $\pm$ 0.12	85.11 $\pm$ 0.12	78.12 $\pm$ 0.10	86.73 $\pm$ 0.12	81.87 $\pm$ 0.04
	DVAR	75.53 $\pm$ 0.32	85.98 $\pm$ 0.27	80.94 $\pm$ 0.19	87.67 $\pm$ 0.10	80.60 $\pm$ 0.17	88.30 $\pm$ 0.22	84.84 $\pm$ 0.08

Table 12: Results on MMBench with six Level-2 (L-2) categories: Logical Reasoning (LR), Attribute Reasoning (AR), Relation Reasoning (RR), Fine Perception-Single (FP-S), Fine Perception-Cross (FP-C), and Coarse Perception (CP).

Method	LLaVA -NeXT-13B	Qwen2.5 -VL-32B	InternVL 3.5-14B
No-CoT	54.55 $\pm$ 0.57	62.51 $\pm$ 0.47	69.84 $\pm$ 0.60
DDCoT	55.29 $\pm$ 0.47	63.46 $\pm$ 0.57	70.58 $\pm$ 0.44
SCAFFOLD	54.87 $\pm$ 0.23	64.61 $\pm$ 0.47	70.37 $\pm$ 0.29
CCoT	57.17 $\pm$ 0.44	64.08 $\pm$ 0.29	71.31 $\pm$ 0.23
ICoT	56.65 $\pm$ 0.57	65.97 $\pm$ 0.52	70.58 $\pm$ 0.44
DVAR	59.79 $\pm$ 0.57	66.39 $\pm$ 0.57	71.62 $\pm$ 0.78

Table 13: Scaling-up and cross-family results on V*.

Model	Method	M ³ CoT	MathVista	V*
LLaVA-NeXT-7B	FarSight	38.92 $\pm$ 0.08	31.58 $\pm$ 0.16	52.04 $\pm$ 0.29
	DVAR	41.25 $\pm$ 0.04	31.71 $\pm$ 0.16	56.65 $\pm$ 0.57
Qwen2.5-VL-7B	FarSight	56.62 $\pm$ 0.07	58.82 $\pm$ 0.21	63.77 $\pm$ 0.86
	DVAR	60.71 $\pm$ 0.11	60.17 $\pm$ 0.19	65.76 $\pm$ 0.95

Table 14: Comparison with FarSight.

we do not report a quantitative comparison with IKOD because an official public implementation was not available at the time of our evaluation, making a reliable direct comparison difficult.

B.9 Qualitative Results

To examine whether the quantitative performance gains also translate into qualitative improvements

in the actual reasoning process, we compare model outputs on M³CoT examples of different types.

In Figure 9, some CoT-based methods develop plausible narratives from initial observations but fail to consistently track key visual cues throughout the reasoning process, ultimately converging to an incorrect answer. In contrast, DVAR uses the FOCUS-RATIONALE format to explicitly specify the referenced region at each step. This region is then used for attention recalibration, helping the model remain anchored to relevant image evidence and avoid reasoning drift.

A similar trend is observed in Figure 10. While the baselines may initially perform plausible step-by-step reasoning, their intermediate reasoning can become overly speculative or drift away from the image evidence, leading to an incorrect final choice. In comparison, DVAR identifies the relevant visual region at each step through FOCUS and derives a short rationale grounded in that region. These examples qualitatively support that dynamic attention recalibration helps maintain grounding to the current visual target during generation and reduces the accumulation of reasoning errors.

		Question: What does the tangram shapes image resemble? Options: A. A car B. A rocket C. A boat D. A house
No-CoT: (D) A house ❌		
SCAFFOLD: To determine what the tangram shapes image resembles, let's analyze it step by step using the provided coordinates: - Identify Key Shapes and Their Positions: - The image consists of several geometric shapes arranged in a specific pattern. - There is a large teal rectangle at the center. - Two dark blue triangles are positioned below the teal rectangle, forming a triangular base. - A smaller pink triangle is placed on top of the teal rectangle. - Analyze the Spatial Relationships: - The teal rectangle appears to be the body of the object. - The dark blue triangles form the base, suggesting stability and grounding. - The pink triangle on top adds an element that could represent a roof or a pointed feature. - Consider the Overall Shape and Structure: - The arrangement of shapes and their relative positions give the impression of a structure with a pointed top and a rectangular middle section. - This combination of shapes and positioning is reminiscent of a traditional house design with a sloped roof. Given this analysis, the tangram shapes image most closely resembles: [[D]]. ❌		
CCoT: <pre>"Scene_Graph": { "Objects": [{ "Object": "triangle_tur", "Attributes": "pink triangle (roof)", }, { "Object": "rectangle_t5", "Attributes": "turquoise rectangle (body)", }, { "Object": "triangle_t6", "Attributes": "dark-blue triangles (base)" }], "Relationships": [{ "Relationship": "triangle_tur is on top of rectangle_t5", }, { "Relationship": "triangle_t6 is below rectangle_t5", }, { "Relationship": "roof+body silhouette resembles a house" }] } "Response": "D" ❌</pre>		
ICoT: To determine what the tangram shapes image resembles, let's analyze the shapes and their arrangement: - The pink shape at the top appears to be a small triangle. - Below the pink triangle is a larger rectangle. - Under the rectangle are two large triangles that form a larger triangle, but this larger triangle is oriented differently compared to the pink triangle. Now, let's consider the options: (A) A car - This would typically include wheels and a body shape that doesn't match the image (B) A rocket - A rocket usually has a triangular base, which is not present here. (C) A boat - A boat would typically have a hull shape and sails or other details that aren't evident in this image. (D) A house - A house often includes a roof and a base, which can resemble the pink triangle on top and the larger blue triangle at the bottom of the image. Given the arrangement and shapes: - The pink triangle could represent the roof. - The green square and rectangles could represent the main body of the house. - The blue triangles at the bottom could represent the base or foundation. Therefore, the most appropriate answer is: (D) A house ❌		
DDCoT: Sub-Question: (1) What are the specific shapes and arrangement of the tangram pieces in the image? (2) Can the tangram pieces be interpreted as a car, a rocket, a boat, or a house? Sub-Answer: (1) The image shows a tangram puzzle composed of seven geometric shapes arranged as follows: - Two Large Triangles (Dark Blue): Positioned at the bottom corners of the figure. - One Medium Triangle (Light Blue): Situated between the two large triangles, forming a larger triangle with them. - One Small Square (Light Blue): Located above the medium triangle, forming part of the larger triangle. - One Small Triangle (Pink): Positioned on top of the small square, forming the peak of the larger triangle. - One Medium Square (Light Blue): Situated below the small square, forming part of the larger triangle. These shapes collectively form a larger triangle with a smaller triangle on top, resembling a mountain or pyramid-like structure. (2) The tangram pieces can the image can resemble a house. Rationale : The tangram pieces in the image are resemble a house because the arrangement of the shapes forms a structure with a triangular roof and a rectangular base, which is characteristic of a house. Answers: (D) a house ❌		
DVAR: STEP 1: FOCUS: The overall structure of the tangram shapes. RATIONALE: The image consists of geometric shapes arranged in a symmetrical form that resembles a vertical tower with a pointed top. STEP 2: FOCUS: The pink triangle at the top. RATIONALE: It appears to be pointing upwards like a rocket's nose cone. STEP 3: FOCUS: The teal square and triangles below the pink triangle. RATIONALE: These shapes look like the body and fins of a rocket. STEP 4: FOCUS: The dark blue triangle at the bottom left. RATIONALE: This shape looks like the base of a rocket. ANSWER: B ✔️		

Figure 9: M³CoT qualitative result on Qwen2.5-VL-7B. DVAR yields the correct answer with a vision-grounded rationale, whereas some baselines drift toward an incorrect choice.

	Question: What can be inferred about the train cars in the picture? Options: A. They are in use for transporting goods B. They have been diverted and are no longer able to move along the tracks C. They have been cleaned and prepared for future use D. They are still on the train tracks
No-CoT: (D) They are still on the train tracks ✗	
SCAFFOLD: To determine what can be inferred about the train cars in the image, let's analyze it step by step: - Look at the train cars: - We can see train cars in the picture. - We can also see train tracks under the train cars. - This means the train cars are on the tracks. - Look at how the trains are sitting: - The trains are stopped. - But stopped trains can still be on the tracks. - They do not look like they went away from the rails. - Compare with the other choices: - We cannot clearly see that they are carrying goods. - We also cannot clearly see cleaning or preparation. - So those choices are harder to know from the picture. - Choose the most visible answer: - The easiest thing to see is that the train cars are still on the train tracks. - That matches choice D the most. Given this analysis, the train cars are most likely: [[D]] ✗	
CCoT: <pre>"Scene_Graph": { "Objects": [{ "Object": "Train Cars", "Attributes": ["Attached to each other", "Located on railway tracks"], "Relationships": ["Train Cars are on the train tracks"] }] }</pre> "Response": "D" ✗	
ICoT: First, we can see train cars in the picture. We can also see tracks under them. This means the train cars are still on the train tracks. The trains are not moving, but that does not mean they are off the tracks. They just look stopped. They do not look like they fell away from the rails. We also cannot clearly see that they are carrying goods. We cannot see anything that looks like cleaning or getting them ready either. Therefore, the correct option is: (D) ✗	
DDCoT: Sub-Question: (1) What objects are clearly visible in the image? (2) Are the train cars placed on top of railway tracks? (3) Is there evidence that the train cars are carrying goods? (4) Is there evidence that the train cars were cleaned or prepared for future use? Sub-Answer: (1) The image clearly shows train cars and railway tracks. (2) Yes, the train cars are placed on the railway tracks. (3) No, there is no evidence that they are carrying goods. (4) No, there is no evidence that they were cleaned or prepared for future use. Rationale : The image clearly shows train cars, and the tracks can also be seen under them. This means the train cars are still on the train tracks. The trains are stopped, but being stopped is not the same as being off the tracks. The image does not clearly show goods, and it also does not show cleaning or preparation. Therefore, the answer should be: Option (D): They are still on the train tracks. ✗	
DVAR: STEP 1. FOCUS: The train cars RATIONALE: Several train cars are parked and remain completely stationary, which suggests they are not currently being used in active transport service. STEP 2. FOCUS: The buffer stops RATIONALE: Red end-of-track barriers are placed directly in front of the trains, indicating that the trains are on terminating tracks where forward movement cannot continue. STEP 3. FOCUS: The rail tracks RATIONALE: The visible tracks are short siding tracks that end at the barriers, supporting the idea that the trains have been moved off a regular route and set aside. STEP 4. FOCUS: The rail yard RATIONALE: The scene appears to be a depot-like yard rather than a passenger platform or cargo-loading area, making active transportation use unlikely and suggesting storage or diversion. STEP 5. FOCUS: Surrounding vehicles RATIONALE: Parked cars are visible near the trains, while no passengers or loading equipment can be seen, making the overall scene more consistent with trains that have been diverted and are no longer moving along the tracks. ANSWER: B ✔	

Figure 10: M³CoT qualitative result on Qwen2.5-VL-7B. DVAR yields the correct answer with a vision-grounded rationale, whereas some baselines drift toward an incorrect choice.