DUAL-VIEW ANCHORED DECODING FOR MITIGATING OBJECT HALLUCINATION IN MLLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal Large Language Models (MLLMs) often suffer from object hallucination, generating objects or attributes that are absent from the visual input. We attribute this failure to the interaction between misleading visual evidence and grounding attenuation: some visual tokens induce uncertain or misleading textual interpretations, while autoregressive decoding progressively amplifies weakly grounded continuations as the textual context grows. Crucially, high epistemic uncertainty does not necessarily imply useless visual information, since a visual token may cover both misleading cues and useful object-specific details. We propose Dual-View Anchored Decoding (DVAD), a training-free decoding strategy that preserves the original full view and constructs an auxiliary uncertainty-suppressed anchor view by selectively suppressing visual tokens likely to carry unstable evidence. DVAD jointly decodes the two views with a shared generated prefix and fuses their next-token distributions at each step, thereby suppressing hallucinated continuations without discarding potentially useful information from high-uncertainty tokens. Without retraining, auxiliary models, or stochastic multi-view sampling, DVAD uses a single uncertainty-guided anchor view and can be implemented as a batched dual-stream decode with only marginal wall-clock overhead. Experiments on CHAIR, THRONE, and AMBER across multiple MLLM backbones **and diverse settings** demonstrate that DVAD consistently reduces object hallucination while preserving caption coverage and quality. Additional evaluations on MMBench and MMStar further show that these gains do not come at the cost of general vision-language understanding.

1 INTRODUCTION

Multimodal Large Language Models (MLLMs) have achieved remarkable progress in vision-language understanding, demonstrating strong performance across a broad range of tasks including visual question answering (Antol et al., 2015), image captioning (Rohrbach et al., 2018), and multi-modal dialogue (Liu et al., 2023; Dai et al., 2023; Zhu et al., 2024). By coupling a powerful vision encoder with the generative capacity of a large language model, these systems produce fluent, contextually rich responses grounded in visual input. Despite this progress, MLLMs remain susceptible to a reliability problem known as *object hallucination*, which refers to the generation of objects, attributes, or relationships that are not present in the image (Rohrbach et al., 2018; Li et al., 2023b; Wang et al., 2023). For example, a model may confidently describe a cat sitting on a table when no cat is present. This failure mode undermines trust and restricts the deployment of MLLMs in critical applications.

We trace object hallucination to two coupled causes rooted in how visual information is consumed during autoregressive generation. In a typical MLLM, a CLIP-based Vision Transformer (Radford et al., 2021) maps the input image into a sequence of patch-level feature vectors, which are projected into the language model's embedding space and prepended to the input context as visual tokens (Liu et al., 2023). First, *visual tokens are not equally reliable*. Reading each visual token through the language model's output head reveals that some tokens induce textual interpretations that deviate sharply from the image-level visual consensus, indicating high *epistemic uncertainty* in the sense of Fang et al. (2024). Such tokens can act as misleading visual evidence and seed hallucinated content. However, uncertainty is not a fixed deletion cue. A visual token is not always the smallest semantic unit of an image. Its semantic coverage depends on the effective visual interface presented to the language model. Dense-token interfaces preserve a large number of spatially localized patch

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.

representations, whereas sparse-token interfaces compress the image into a small set of globally aggregated representations, often through a query-based bottleneck. Consequently, a high-uncertainty token is more likely to represent a local outlier in a dense-token interface, but may itself carry salient object-specific evidence in a sparse-token interface. Treating all high-uncertainty tokens as disposable therefore risks removing useful fine-grained visual details together with misleading evidence.

Second, the decoding dynamics *amplify* these unreliable or ambiguous visual cues. As generation progresses, the number of textual tokens grows monotonically while the visual tokens remain fixed, so causal self-attention becomes increasingly dominated by statistical patterns accumulated in the textual context, progressively *attenuating* the influence of the visual signal. The model therefore drifts toward continuations that are linguistically plausible given the preceding text—including those seeded by misleading visual evidence—rather than tokens that are visually faithful to the image. This interaction between misleading or ambiguous visual evidence and grounding attenuation, compounding over decoding steps, is the mechanism by which object hallucination emerges.

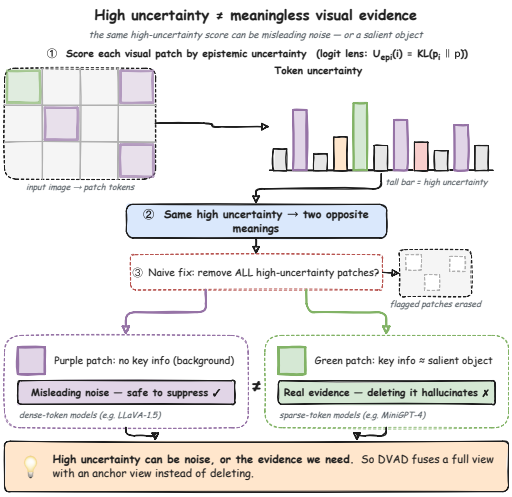

Figure 1: High-uncertainty visual tokens do not always carry the same meaning: the same score may indicate misleading noise or salient object evidence. DVAD therefore fuses the full view with an uncertainty-guided anchor view instead of deleting such tokens.

Existing approaches to mitigate object hallucination fall into several categories, each with notable limitations. Post-hoc correction methods (Gunjal et al., 2024; Zhai et al., 2023; Wu et al., 2025; 2026) detect and revise hallucinated outputs using external grounding models or verification pipelines, but introduce substantial inference overhead and depend critically on the quality of the auxiliary corrector. Supervised fine-tuning and reinforcement-based alignment methods (Gunjal et al., 2024; Zhou et al., 2024; Geigle et al., 2024) address the problem at training time, but require carefully curated, fine-grained visual annotations and produce models tightly coupled to specific training distributions. Contrastive and ensemble-based decoding methods (Leng et al., 2024; Chen et al., 2024b; Fang et al., 2024; Tong et al., 2025) offer training-free alternatives by contrasting the original visual context with degraded references or aggregating predictions from randomly masked contexts. However, these methods usually incur multiple forward passes per decoding step and rely on replacing or perturbing the original visual evidence; in particular, uncertainty-guided dropout treats high-uncertainty tokens primarily as candidates for removal, whereas such tokens may still contain useful object-specific details.

In this paper, we propose Dual-View Anchored Decoding (DVAD), a training-free decoding strategy that treats uncertainty not as a hard deletion cue, but as a signal for auxiliary consistency guidance. DVAD keeps the original *full view* throughout generation, preserving useful object-specific details even when they appear in high-uncertainty visual tokens, and constructs an uncertainty-suppressed anchor view by selectively suppressing tokens likely to carry misleading or unstable evidence. The two views are decoded jointly with a shared generated prefix, and their next-token distributions are fused at each step, allowing DVAD to suppress hallucinated continuations driven by suspicious evidence while still fully exploiting the original visual context.

This design mitigates hallucination during generation itself, without post-hoc editing, auxiliary models, or stochastic dropout ensembles. DVAD requires no retraining and incurs only marginal wall-clock overhead when implemented as batched dual-view decoding.

Our main contributions are as follows:

- We characterize object hallucination as the interplay of *misleading visual evidence* and *grounding attenuation*. We further show that high epistemic uncertainty is not a uniform deletion cue: depending on whether the visual interface preserves many spatially localized

tokens or compresses the image into a small set of aggregated tokens, uncertain tokens may either correspond to noisy outliers or contain useful object-specific details.

- We propose DVAD, a training-free and deterministic decoding strategy that preserves the full visual view while constructing an auxiliary uncertainty-suppressed anchor view. By jointly decoding and fusing the two views' next-token distributions, DVAD suppresses hallucinations without discarding potentially useful information from high-uncertainty visual tokens.
- We extensively evaluate DVAD on three representative object-hallucination benchmarks, CHAIR (Rohrbach et al., 2018), THRONE (Kaul et al., 2024), and AMBER (Wang et al., 2023). DVAD consistently reduces object hallucination while preserving caption coverage and quality, with substantially lower inference cost than ensemble-based decoding methods. Evaluations on MMBench (Liu et al., 2024b) and MMStar (Chen et al., 2024a) further confirm that these gains do not come at the cost of general vision-language understanding.

2 Related Work

2.1 Vision-Language Models

LLaVA (Liu et al., 2023) and LLaVA-1.5 (Liu et al., 2024a) establish strong baselines by connecting a CLIP vision encoder to a language model via a simple MLP projection, trained on visual instruction-following data. LLaVA-OneVision (Li et al., 2024) further extends this framework to multi-image and multi-resolution visual understanding. InstructBLIP (Dai et al., 2023) adopts a Q-Former (Li et al., 2023a) to bridge the vision encoder and language model, enabling instruction-conditioned visual feature extraction. MiniGPT-4 (Zhu et al., 2024) aligns a frozen visual encoder with an advanced language model using a lightweight projection layer, while mPLUG-Owl2 (Ye et al., 2024) improves multimodal collaboration through a modular vision-language design. More recent systems, such as Qwen3-VL (Bai et al., 2025), Qwen3.5-Omni (Team, 2026), and InternVL3.5 (Wang et al., 2025), further illustrate the diversity of modern MLLM architectures. In this work, we evaluate DVAD on LLaVA-1.5, MiniGPT-4, and mPLUG-Owl2, covering both dense spatial-token interfaces and sparse compressed-token interfaces.

2.2 Object Hallucination in MLLMs

Object hallucination has emerged as a central reliability concern in MLLMs (Rohrbach et al., 2018; Li et al., 2023b; Wang et al., 2023). Early work by Rohrbach et al. (2018) systematically characterized hallucinated objects in image captioning and introduced the CHAIR metric to quantify them. Li et al. (2023b) subsequently proposed POPE, a discriminative probing benchmark that evaluates whether models falsely assert the presence of non-existent objects. More recent benchmarks extend hallucination evaluation to richer and more challenging settings. MIRAGE (Dong et al., 2026) studies hallucinations in multimodal reasoning chains, while MIHBench (Li et al., 2025) and MIOH (Min et al., 2026) evaluate object hallucination in multi-image scenarios involving existence, counting, attributes, and cross-image consistency. ReactBench (Zhou et al., 2026) further diagnoses hallucination from a cause-driven perspective by targeting co-occurrence bias, language priors, comparative perception failures, and fine-grained counting. Collectively, these benchmarks establish that hallucination arises not only from object-existence errors, but also from insufficient visual grounding under complex perception and reasoning demands, directly motivating our analysis of grounding attenuation.

2.3 Decoding-Time Interventions

Decoding-time methods are attractive because they are training-free and plug-and-play. VCD (Leng et al., 2024) reduces hallucination by contrasting the original output distribution with that from a noise-corrupted image, suppressing tokens driven by language priors rather than visual evidence. Recent variants strengthen this contrast by masking object-centric evidence (Chen et al., 2026), constructing dynamic step-specific references (Suo et al., 2025), or detecting hallucination-prone tokens probabilistically (Fieback et al., 2025). However, these methods typically require separate contrastive reference forwards during decoding, which increases per-token latency. DVAD also uses

a dual-view decoding process, but constructs the anchor view once per image and decodes the two views synchronously as a batched dual-stream forward. This design avoids stochastic multi-view sampling and auxiliary models while achieving comparable or stronger hallucination reduction.

3 METHOD

Figure 2 illustrates DVAD. Our starting point is the observation that object hallucination in multimodal LLMs is governed by a small subset of visual tokens whose textual reading is unreliable: patches whose interpretation diverges from what the image as a whole supports. DVAD turns this observation into a decoding procedure with three ingredients. First, during an additional probe prefill it measures, for every visual token, an epistemic uncertainty that flags such unreliable patches. Second, it suppresses the flagged patches by zeroing their projected visual features, forming a deterministic anchor view. Third, it decodes the original and anchor views jointly and fuses their next-token distributions at every step through a single signed coefficient; the sign of that coefficient selects, per backbone, between two complementary uses of the anchor view and is the only quantity DVAD adapts across models. The procedure is training-free, deterministic, needs no auxiliary model or external image distortion, and incurs only marginal wall-clock overhead when the two views are decoded as a batch.

3.1 EPISTEMIC UNCERTAINTY FROM THE PREFILL

In autoregressive decoding, the prefill pass processes the image tokens and the textual prompt once to initialize the KV cache before generating the first output token. Given an image I, the vision tower produces N patch features $F = \{f_1, \dots, f_N\} \subset \mathbb{R}^{d_v}$, which the cross-modal projector Φ subsequently maps into the LM embedding space, $\tilde{f}_i = \Phi(f_i) \in \mathbb{R}^{d_l}$, forming the visual tokens $\tilde{F}$ consumed by the language model.

We read out the model's own interpretation of each visual token from an additional probe prefill, performed once before decoding. Let $h_i^{(L)} \in \mathbb{R}^{d_l}$ be the final-layer hidden state at the position of visual token i, and let W_U denote the unembedding matrix. The logit-lens projection

$$p_i \;=\; \mathrm{Softmax}\big(W_U h_i^{(L)}\big) \;\in\; \Delta^{|\mathcal{V}|} \tag{1}$$

is the vocabulary distribution that patch i would emit in context, i.e., its textual reading. This readout is used only to score visual tokens and does not modify the model parameters or the subsequent generation process.

Aggregating over all image patches gives an internal visual-token consensus,

$$\bar{p} \;=\; \tfrac{1}{N} \textstyle\sum_{i=1}^{N} p_i, \tag{2}$$

which serves only as the model's internal reference, rather than a ground-truth semantic distribution. We then quantify the *epistemic* uncertainty of visual token i as

$$U_{\mathrm{epi}}(i) \;=\; \mathrm{KL}\big(p_i \,\|\, \bar{p}\big), \tag{3}$$

following prior uncertainty-guided decoding work (Fang et al., 2024).

A large $U_{\mathrm{epi}}(i)$ marks a patch whose textual interpretation stands out from the rest of the image. Crucially, what such standing-out means depends on the visual interface presented to the language model. When the interface retains many spatially localized patch tokens, a high-divergence token is more likely to be a local outlier. When the interface aggressively compresses the image into a small set of aggregated tokens, a high-divergence token may instead carry salient object-specific evidence. DVAD exploits this distinction through the sign of the fusion coefficient during dual-view decoding.

3.2 UNCERTAINTY-GUIDED ANCHOR VIEW CONSTRUCTION

Rather than perturbing decoding with randomly sampled token subsets, DVAD forms a single *deterministic* second view of the image by suppressing its most uncertain visual evidence. Let $\mathcal{M}_\rho \subset \{1, \dots, N\}$ index the $\lceil \rho N \rceil$ patches with the highest epistemic uncertainty (Eq. 3). The

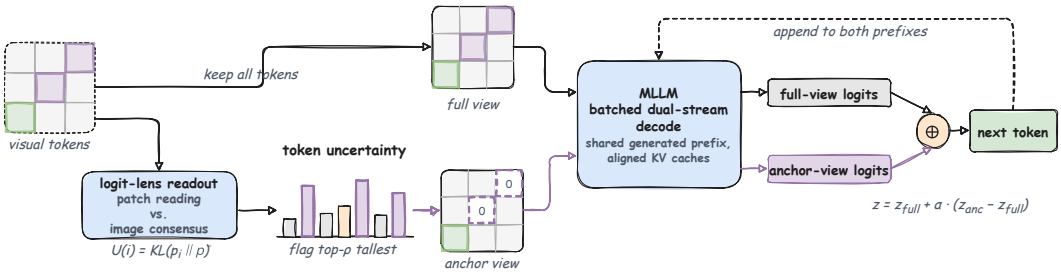

Figure 2: Overview of DVAD. During an additional probe prefill, each visual token is scored by its logit-lens divergence from the image-level consensus, and the top-ρ most uncertain tokens are zeroed to construct a deterministic anchor view. The full and anchor views are then decoded as a batched dual stream with a shared generated prefix, and their next-token logits are fused at every step via $z = z_{\text{full}} + \alpha\,(z_{\text{anc}} - z_{\text{full}})$, where the fused token is appended to both prefixes. The sign of α is set per backbone: consistency fusion ($\alpha > 0$) for dense-token models and contrastive fusion ($\alpha < 0$) for sparse-token models.

anchor view zeros their projected features while leaving all other visual tokens untouched:

$$\tilde{f}_i^{\text{anc}} = \begin{cases} \mathbf{0}, & i \in \mathcal{M}_\rho, \\ \tilde{f}_i, & \text{otherwise.} \end{cases} \tag{4}$$

For backbones whose projector output does not map one-to-one onto LM positions (e.g. the any-resolution packing of LLaVA-NeXT), the same masking is applied at the language-model input, which is position-exact for any architecture. The anchor view and the full view are *imbalanced* renderings of the same image, one with the high-divergence patches suppressed and one complete; the next section shows that the way to combine them is not fixed but flips with the backbone.

3.3 Signed Dual-View Fusion Decoding

DVAD decodes the two views *synchronously* as a two-row batch that shares the generated prefix and differs only in its visual tokens. At step t, the two rows produce logits z_t^{full} and z_t^{anc}, which are fused into a single logit vector:

$$\begin{aligned} z_t &= (1-\alpha)\, z_t^{\text{full}} + \alpha\, z_t^{\text{anc}} \\ &= z_t^{\text{full}} + \alpha \underbrace{\left(z_t^{\text{anc}} - z_t^{\text{full}}\right)}_{\text{evidence shift}}, \qquad \alpha \in \mathbb{R}. \end{aligned} \tag{5}$$

The next token is selected greedily from the fused logits and appended to both rows, so their prefixes remain identical and their KV caches stay aligned at the cost of one batched forward per step. The second form of Eq. 5 makes the role of α explicit: α moves the prediction along the *evidence shift*, i.e., the change in the model's beliefs caused by suppressing the high-uncertainty visual features. The *sign* of α determines whether DVAD moves with this shift or against it. DVAD counteracts grounding attenuation not by changing its structural cause but by intervening at every decoding step. Since the evidence shift is computed between two streams that differ only in their visual tokens, it re-weights visual evidence against the accumulating language prior before weakly grounded tokens can enter the prefix and compound.

Two regimes, one mechanism. The appropriate sign depends on the effective visual interface presented to the language model. *(i) dense-token interfaces* retain many spatially localized visual tokens (e.g. LLaVA-1.5 with 576 tokens). A high-uncertainty token is therefore more likely to be a local outlier carrying spurious evidence. Suppressing it produces a better-grounded anchor view, so DVAD fuses *toward* the anchor with $\alpha > 0$, reinforcing tokens supported by both views. *(ii) Sparse-token interfaces* compress the image into a small set of aggregated visual tokens (e.g. MiniGPT-4 with 32 query tokens and mPLUG-Owl2 with 64). In this regime, a high-uncertainty token may instead carry salient object-specific evidence. Suppressing it can shift the anchor toward a generic, language-prior continuation, so DVAD fuses *against* the anchor with $\alpha < 0$.

Model	Method	CHAIR		THRONE				AMBER			
		$C_S\downarrow$	$C_I\downarrow$	$P\uparrow$	$R\uparrow$	$F_1\uparrow$	$F_{0.5}\uparrow$	C$\downarrow$	Cov$\uparrow$	Hal$\downarrow$	Cog$\downarrow$
LLaVA-1.5	Greedy	43.9±0.8	13.1±0.3	79.9±1.6	72.9±1.6	75.3±0.6	77.6±1.2	6.0±0.3	48.2±0.4	24.0±0.9	2.8±0.1
	Dropout	42.7±3.0	13.0±0.7	79.5±0.7	72.1±2.0	74.7±0.9	77.2±1.0	6.2±0.4	47.4±0.2	25.1±1.8	3.0±0.1
	VCD	49.5±1.4	14.6±0.5	75.0±1.3	72.4±0.7	74.2±0.6	74.8±0.9	7.5±0.1	48.6±0.2	34.9±1.3	2.9±0.2
	SHIELD	43.7±0.8	13.1±0.4	78.9±1.2	73.1±1.7	**75.7**±1.4	77.4±1.4	6.0±0.3	**48.7**±0.7	23.8±0.2	2.8±0.2
	DVAD	**40.1**±0.4	**11.5**±0.7	**80.6**±0.6	**73.2**±1.9	**75.7**±0.8	**78.3**±0.9	**5.1**±0.1	**48.7**±0.5	**23.3**±1.1	**2.3**±0.1
MiniGPT-4	Greedy	36.6±0.9	11.7±0.9	81.3±0.9	59.6±1.5	67.5±1.2	73.7±0.7	18.2±0.7	65.6±0.2	80.9±1.1	17.1±0.4
	Dropout	35.6±0.2	11.6±1.2	78.9±0.9	56.6±0.3	64.9±0.9	72.1±0.6	18.3±0.2	65.0±0.3	81.3±0.7	16.6±0.7
	VCD	34.6±2.0	10.6±0.7	81.8±0.9	58.7±1.5	66.8±1.0	73.4±0.4	**17.9**±0.3	66.4±0.4	78.5±1.2	16.5±0.3
	SHIELD	35.5±0.3	**10.4**±0.3	81.7±2.6	59.9±1.5	68.1±0.4	74.3±1.0	18.7±0.3	66.5±0.3	80.5±0.2	17.1±0.1
	DVAD	**33.4**±1.7	10.6±0.2	**83.7**±0.4	**60.2**±1.1	**68.8**±0.4	**75.5**±0.3	**17.9**±0.3	**66.8**±0.2	**78.2**±1.3	**16.3**±0.4
mPLUG-Owl2	Greedy	54.2±1.9	16.0±1.0	78.0±1.2	72.5±0.7	74.7±0.7	76.4±0.9	6.8±0.2	50.1±0.1	27.2±1.2	3.1±0.3
	Dropout	58.8±0.8	16.8±0.6	74.8±0.8	74.6±1.0	74.1±0.4	74.3±0.3	7.1±0.4	49.5±0.4	31.5±1.1	2.8±0.0
	VCD	59.1±2.4	16.3±0.6	74.6±1.5	**76.5**±1.5	74.9±0.3	74.7±0.8	7.3±0.6	49.8±0.1	31.4±1.7	2.6±0.2
	SHIELD	57.3±1.0	16.4±0.8	73.9±0.4	73.2±0.8	73.5±0.9	74.0±1.0	8.9±0.5	49.0±0.2	35.3±2.1	3.2±0.3
	DVAD	**51.1**±0.9	**14.7**±0.6	**78.1**±1.6	73.3±1.1	**75.1**±0.1	**76.8**±0.7	**6.0**±0.4	**50.3**±0.5	**27.0**±1.8	**2.4**±0.3

Table 1: Comparison with decoding-based and contrastive baselines on three hallucination benchmarks. Results are averaged over three random seeds. Lower is better for C_S/CI/C/Hal/Cog, while higher is better for P/R/F_1/$F0.5$/Cov. Bold indicates the best result within each model block.

In practice, token count is a useful indicator but not a universal boundary. Models with a dense spatial-token interface use $\alpha > 0$, whereas those with an aggressively compressed visual interface use $\alpha < 0$. This single backbone-level choice selects between consistency and contrastive fusion; all other hyperparameters remain fixed.

Overall, DVAD adds a single probe prefill (Eq. 1) and decodes the full and anchor views as a two-row batch. This introduces two logical decoding streams but only marginal wall-clock overhead in practice, while requiring no training, no auxiliary model, and no external image distortion.

4 Experiments

4.1 Experimental Settings

Benchmarks and metrics. We evaluate on three hallucination benchmarks. CHAIR Rohrbach et al. (2018) measures object hallucination in open-ended COCO val2014 captioning with sentence- and instance-level rates, CHAIR_S and CHAIR_I. THRONE Kaul et al. (2024) compares free-form descriptions with ground-truth objects, reporting P, R, F_1, and $F_{0.5}$. AMBER Wang et al. (2023) is evaluated with its official generative metrics: CHAIR, Cover, Hal, and Cog. We additionally report MMBench Liu et al. (2024b) and MMStar Chen et al. (2024a) to assess general visual understanding. For CHAIR, THRONE, and AMBER, we sample N=500 images and report mean±std over three independently seeded subsets.

Backbones and baselines. We apply our training-free approach to three MLLMs, LLaVA-1.5 Liu et al. (2024a), MiniGPT-4 Zhu et al. (2024), and mPLUG-Owl2 Ye et al. (2024), and compare with Greedy, Dropout Decoding Fang et al. (2024), VCD Leng et al. (2024), and SHIELD Huang et al. (2025), using matched samples and seeds.

Implementation. All methods use greedy decoding (max 512 tokens) with frozen public checkpoints and no weight updates. We fix $\rho = 0.4$ across all models and set $|\alpha| = 0.15$ by default; the sign of α follows the dense/sparse visual-token regime described in the method section. All experiments are run on two NVIDIA A100 80GB GPUs.

4.2 Comparison with State-of-the-Art Methods

Hallucination Benchmarks. Table 1 compares DVAD with greedy decoding and three representative decoding or contrastive baselines on the CHAIR, THRONE, and AMBER benchmarks. Overall, DVAD consistently reduces hallucinations across different MLLM backbones while preserving recall-oriented metrics. Compared with greedy decoding, DVAD improves the main hallucination evaluation scores for all three models. On LLaVA-1.5, DVAD reduces CHAIR_S from 43.9 to 40.1 and CHAIR_I from 13.1 to 11.5, while also improving the THRONE F_1 and $F_{0.5}$ scores. On MiniGPT-4, DVAD achieves the best CHAIR_S and THRONE scores among all methods, improving F_1 from 67.5 to 68.8 and $F_{0.5}$ from 73.7 to 75.5. On mPLUG-Owl2, DVAD substantially reduces CHAIR_S from 54.2 to 51.1 and obtains the best hallucination-related AMBER scores among the compared methods. A key advantage of DVAD is that its gains are not achieved merely by suppressing generation. Dropout can sometimes reduce CHAIR scores, but it often degrades recall-sensitive metrics, such as THRONE recall and AMBER coverage. This suggests that the hallucination reduction achieved by DVAD does not simply stem from making the model more conservative.

Method	LLaVA-1.5		MiniGPT-4		mPLUG-Owl2	
	MMBench↑	MMStar↑	MMBench↑	MMStar↑	MMBench↑	MMStar↑
Greedy	73.2	33.5	49.0	26.2	–	32.7
Dropout	73.2	33.5	49.1	26.7	73.9	32.0
VCD	73.8	33.6	49.9	**27.7**	74.8	33.2
SHIELD	73.8	33.6	50.2	26.9	73.8	33.2
DVAD	**74.0**	**34.3**	**50.6**	**27.7**	**74.9**	**33.5**

Table 2: MMBench and MMStar accuracy (%). Best results within each model are shown in bold.

General Benchmarks. Beyond hallucination-specific benchmarks, we further evaluate whether DVAD preserves general visual understanding abilities. Table 2 reports the results on MMBench and MMStar, which cover a broader range of visual perception, reasoning, and instruction-following capabilities. Overall, DVAD does not degrade general visual understanding. Instead, it achieves the best or tied-best performance in nearly all model–benchmark combinations. On LLaVA-1.5, compared with greedy decoding, DVAD improves the MMBench score from 73.2 to 74.0 and the MMStar score from 33.5 to 34.3. On MiniGPT-4 and mPLUG-Owl2, DVAD likewise achieves the best results. These results indicate that DVAD reduces hallucinations without sacrificing general capabilities. This is important because a simple way to reduce hallucinations is to make the model more conservative, which may suppress correct and informative visual content. In contrast, DVAD improves hallucination-oriented metrics while maintaining, and even slightly improving, performance on general visual understanding benchmarks.

4.3 Further Analysis

Ablation on number of views. Our decoder fuses the full view with one uncertainty-suppressed anchor view. To test whether additional anchor views are beneficial, we vary the number of views from one, equivalent to standard greedy decoding, to four. Figure 3 reports the mean and standard deviation of C_S, C_I, P, and R over three random seeds. The main gain comes from adding the first anchor view: C_S drops from 43.9 to 40.1 and C_I from 13.1 to 11.5, while both precision and recall slightly improve. Beyond two views, performance largely saturates. Although three views achieve the best numerical C_S and C_I, the improvements over two views are within the standard deviation, and P and R are already best with two views. Since our anchor view is deterministic rather than sampled as in Dropout decoding, a single additional view is sufficient. We therefore use two views by default to balance quality and cost.

Ablation on each component. Figure 4 shows that the anchor view is the main source of hallucination reduction, lowering CHAIR_S from 43.9 to 40.9 and CHAIRI from 13.1 to 11.8. Full dual-view fusion further improves both metrics to 40.1 and 11.5, respectively, while also improving caption quality: precision increases to 80.6, and F_1 and $F0.5$ reach 75.7 and 78.3. Recall also slightly improves from 72.9 to 73.2, indicating that the hallucination reduction does not sacrifice

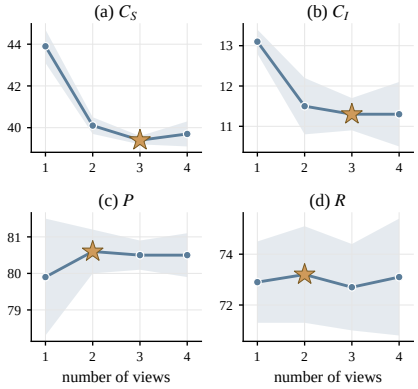

Figure 3: Effect of the number of views on LLaVA-1.5, evaluated on CHAIR (C_S, C_I) and THRONE (P, R). One view denotes greedy decoding and two views denote the default DVAD setting. Curves report mean±std over three seeds, and ⋆ marks each metric's best value.

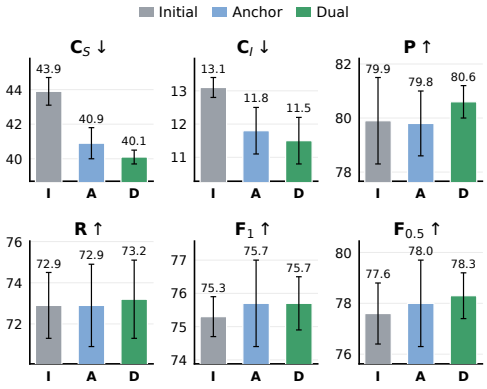

Figure 4: Component ablation on LLaVA-1.5, evaluated on CHAIR (C_S, C_I) and THRONE (P, R, F_1, $F_{0.5}$). Each component is enabled in turn: Initial (I), Anchor (A), Dual (D). Bars and error bars denote the mean and standard deviation over three seeds, respectively.

coverage. These results show that the two views are complementary and that the full model improves faithfulness while preserving coverage.

Selection strategy	C_S↓	C_I↓	P↑	R↑	F_1↑	$F_{0.5}$↑
Baselines						
Greedy (no DVAD)	43.9	13.1	79.9	72.9	75.3	77.6
Masked-patch selection						
Random	42.6	11.5	79.5	71.6	75.5	77.9
Entropy	41.0	**11.1**	**80.6**	70.8	75.2	77.8
KL-uncertainty (ours)⋆	**40.1**	11.5	**80.6**	**73.2**	**75.7**	**78.3**

Table 3: Ablation on masked-patch selection with LLaVA-1.5, evaluated on CHAIR (C_S, C_I) and THRONE (P, R, F_1, $F_{0.5}$). Best results are shown in bold.

Ablation on Masked-Patch Selection. Table 3 compares different masked-patch selection strategies. Random masking brings only limited gains over greedy decoding, indicating that simply removing visual tokens is insufficient. Entropy-based selection achieves the best CHAIR_I and improves precision, but reduces THRONE recall to 70.8, suggesting that it may discard useful visual evidence. In contrast, our KL-uncertainty strategy provides the best overall trade-off, achieving the lowest CHAIR_S of 40.1, higher recall of 73.2, and the best F_1 and $F_{0.5}$. These results indicate that selecting patches by local–global visual inconsistency yields a more effective anchor view than prediction uncertainty alone, leading to more grounded decoding.

Inference efficiency. Table 4 compares per-image decoding latency on LLaVA-1.5 using 50 COCO images, 512-token captioning, batch size one, and a single A100, excluding model loading. The Passes column is written as p+v, where p denotes logit-lens probe prefills and v denotes batched decode streams; CW_{30} denotes SHIELD's 30-step per-image adversarial attack. DVAD uses one probe and decodes the full and uncertainty-masked views in a single batched forward pass, making its latency nearly identical to greedy decoding (2.48 vs. 2.44 s/image, 1.02×). In contrast, Dropout votes over three masked streams (4.1×), SHIELD

Method	**Passes**	**s/img**↓	**Rel.**↓
Greedy	1	2.44	1.00×
Dropout	1+3	10.18	4.17×
VCD	2	2.67	1.09×
SHIELD	$1+\text{CW}_{30}$	5.58	2.29×
DVAD (ours)	1+2	**2.48**	**1.02**×

Table 4: Decoding speed on LLaVA-1.5.

performs a costly per-image attack (2.3×), and VCD is slightly slower (1.09×). Thus, DVAD achieves its accuracy gains with only marginal additional wall-clock latency.

Hyperparameter Analysis. Figure 5 summarizes the sensitivity to α and ρ. We observe two main trends. First, CHAIR metrics and recall exhibit opposite behavior: increasing α consistently reduces hallucination, while recall is highest at smaller α. This indicates that DVAD provides a monotonic control for balancing faithfulness and coverage. For example, $\alpha = 0.5$ reduces CHAIR_S below 20 but sharply lowers recall. Second, performance is relatively robust to ρ: metrics remain stable for $\rho \in [0.2, 0.4]$ and degrade only when ρ becomes very large. The LLaVA-1.5 default setting lies in this stable region, using a moderate α to reduce hallucination while preserving recall and achieving the best F_1. Overall, the method is insensitive to moderate changes in ρ, while α provides a direct control over the hallucination–recall trade-off. Importantly, the favorable region is not confined to a single grid point, suggesting that the observed gains do not rely on brittle hyperparameter tuning.

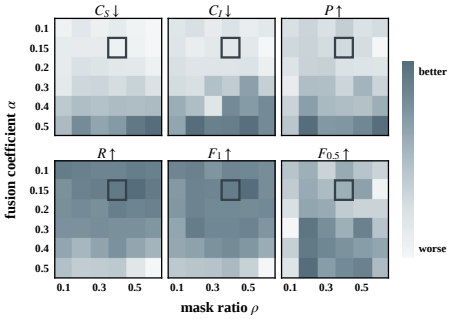

Figure 5: Hyperparameter sensitivity on the $\alpha \times \rho$ grid for LLaVA-1.5.

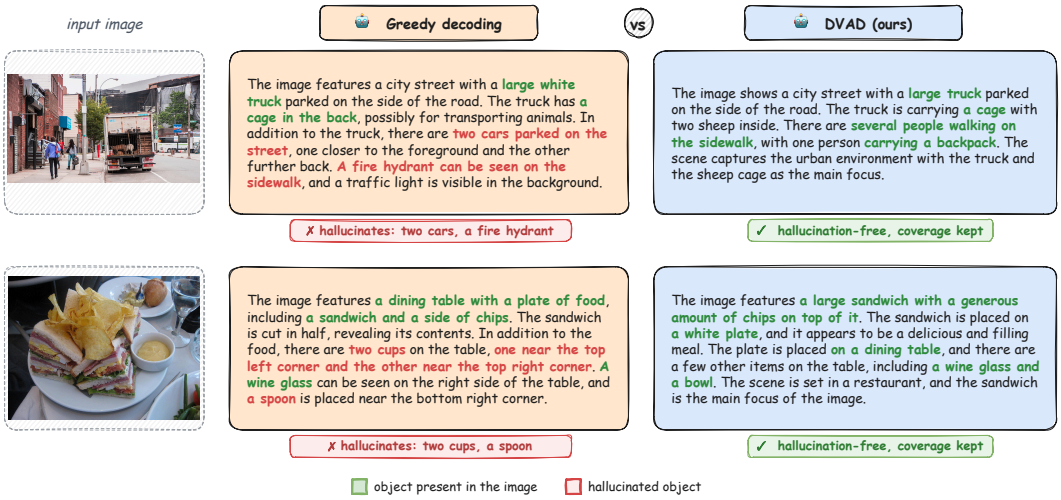

Figure 6: Qualitative comparison between greedy decoding and DVAD with LLaVA-1.5 on two COCO images. **Green** marks correct descriptions, while **red** marks hallucinations.

Qualitative Analysis. Figure 6 compares greedy decoding with our method on two images. Greedy decoding hallucinates objects, including *two cars* and *a fire hydrant* in the street scene, and *two cups* and *a spoon* in the food scene. Our method removes these hallucinations while preserving grounded content. By suppressing uncertain patches in the anchor view, misleading evidence receives less support during fusion, reducing hallucination without sacrificing caption coverage.

5 Conclusion

We presented DVAD, a training-free decoding framework for reducing object hallucination in MLLMs. DVAD treats high-uncertainty tokens as cues for constructing an anchor view rather than disposable noise. Jointly decoding the full and anchor views with a shared prefix suppresses hallucinations while preserving grounded details. Experiments on CHAIR, THRONE, and AMBER show consistent hallucination reductions across MLLM backbones while maintaining caption quality and visual coverage. MMBench and MMStar results further indicate preserved visual understanding. Overall, DVAD offers a deterministic, plug-and-play strategy for improving multimodal faithfulness without retraining or auxiliary models.

AI USE STATEMENT

In this work, we used generative AI tools to assist with coding and to improve the clarity and readability of the manuscript. All AI-assisted code was verified and tested for correctness, and all AI-assisted text was reviewed and revised by the authors. We did not use generative AI tools to generate research hypotheses, design the methodology or experiments, perform mathematical derivations or proofs, or analyze and interpret experimental results. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433, 2015.

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

Boqi Chen, Xudong Liu, and Jianing Qiu. Mask what matters: Mitigating object hallucinations in multimodal large language models with object-aligned visual contrastive decoding. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pp. 9–16, 2026.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? *Advances in Neural Information Processing Systems*, 37:27056–27087, 2024a.

Zhaorun Chen, Zhuokai Zhao, Hongyin Luo, Huaxiu Yao, Bo Li, and Jiawei Zhou. Halc: Object hallucination reduction via adaptive focal-contrast decoding. *arXiv preprint arXiv:2403.00425*, 2024b.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning, 2023. URL https://arxiv.org/abs/2305.06500.

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

Yixiong Fang, Ziran Yang, Zhaorun Chen, Zhuokai Zhao, and Jiawei Zhou. From uncertainty to trust: Enhancing reliability in vision-language models with uncertainty-guided dropout decoding. *arXiv preprint arXiv:2412.06474*, 1, 2024.

Laura Fieback, Nishilkumar Balar, Jakob Spiegelberg, and Hanno Gottschalk. Efficient contrastive decoding with probabilistic hallucination detection-mitigating hallucinations in large vision language models. *arXiv preprint arXiv:2504.12137*, 2025.

Gregor Geigle, Radu Timofte, and Goran Glavaš. Does object grounding really reduce hallucination of large vision-language models? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 2728–2742, 2024.

Anisha Gunjal, Jihan Yin, and Erhan Bas. Detecting and preventing hallucinations in large vision language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 18135–18143, 2024.

Yiyang Huang, Liang Shi, Yitian Zhang, Yi Xu, and Yun Fu. Shield: Suppressing hallucinations in lvlm encoders via bias and vulnerability defense. *arXiv preprint arXiv:2510.16596*, 2025.

Prannay Kaul, Zhizhong Li, Hao Yang, Yonatan Dukler, Ashwin Swaminathan, CJ Taylor, and Stefano Soatto. Throne: An object-based hallucination benchmark for the free-form generations of large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27228–27238, 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13872–13882, 2024.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

Jiale Li, Mingrui Wu, Zixiang Jin, Hao Chen, Jiayi Ji, Xiaoshuai Sun, Liujuan Cao, and Rongrong Ji. Mihbench: Benchmarking and mitigating multi-image hallucinations in multimodal large language models. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 3143–3152, 2025.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023a.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models, 2023b. URL `https://arxiv.org/abs/2305.10355`.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 26296–26306, 2024a.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pp. 216–233. Springer, 2024b.

Joonki Min, Chaeyun Kim, Hyungwook Choi, Yejin Kim, Kihyun Kim, Yohan Jo, and Joonseok Lee. Fine-grained multi image object hallucination benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18295–18305, 2026.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object hallucination in image captioning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 4035–4045, 2018.

Wei Suo, Lijun Zhang, Mengyang Sun, Lin Yuanbo Wu, Peng Wang, and Yanning Zhang. Octopus: Alleviating hallucination via dynamic contrastive decoding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 29904–29914, 2025.

Qwen Team. Qwen3. 5-omni technical report. *arXiv preprint arXiv:2604.15804*, 2026.

Bingkui Tong, Jiaer Xia, and Kaiyang Zhou. Mitigating hallucination in multimodal llms with layer contrastive decoding. *arXiv preprint arXiv:2509.25177*, 2025.

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

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Tsung-Han Patrick Wu, Heekyung Lee, Jiaxin Ge, Joseph Gonzalez, Trevor Darrell, and David Chan. Generate, but verify: Reducing hallucination in vision-language models with retrospective resampling. *Advances in Neural Information Processing Systems*, 38:65749–65777, 2026.

Yixuan Wu, Yang Zhang, Jian Wu, Philip Torr, and Jindong Gu. Postalign: Multimodal grounding as a corrective lens for mllms. *arXiv preprint arXiv:2506.17901*, 2025.

Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, and Fei Huang. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pp. 13040–13051, 2024.

Bohan Zhai, Shijia Yang, Chenfeng Xu, Sheng Shen, Kurt Keutzer, and Manling Li. Halle-switch: Controlling object hallucination in large vision language models. *arXiv e-prints, pp. arXiv–2310*, 2023.

Shizhe Zhou, Bohan Jia, Kai Wu, Yan Shen, Tongyun Li, Yuyang Wu, and Shaohui Lin. Reactbench: A cause-driven benchmark for multimodal hallucination via systematic evaluation. *arXiv preprint arXiv:2605.29579*, 2026.

Yiyang Zhou, Zhiyuan Fan, Dongjie Cheng, Sihan Yang, Zhaorun Chen, Chenhang Cui, Xiyao Wang, Yun Li, Linjun Zhang, and Huaxiu Yao. Calibrated self-rewarding vision language models. *arXiv preprint arXiv:2405.14622*, 2024.

Deyao Zhu, Xiaoqian Shen, Xiang Li, Mohamed Elhoseiny, et al. Minigpt-4: Enhancing vision-language understanding with advanced large language models. In *International Conference on Learning Representations*, volume 2024, pp. 18378–18394, 2024.

A Appendix

B Implementation Details

B.1 Where the Anchor View Is Masked

DVAD builds the anchor view by zeroing the projected features of the flagged visual tokens (Eq. 4 of the main paper). The exact insertion point depends on how a backbone maps projector outputs onto language-model positions, and we use one of two equivalent implementations.

Projector-output masking is used for LLaVA-1.5, MiniGPT-4, and mPLUG-Owl2. In these architectures the cross-modal projector emits exactly one vector per visual token, in one-to-one correspondence with language-model positions. We register a forward hook on the projector and multiply its output by a per-token gain vector $w \in \{0,1\}^N$, where the full-view row uses $w_i = 1$ everywhere and the anchor row uses $w_i = 0$ for $i \in M_\rho$. Because the hook sees the whole two-row batch at once, the full and anchor views are produced by a single forward pass rather than two.

Language-model-input masking is used when the projector output does not align one-to-one with LM positions, as in the any-resolution packing of LLaVA-NeXT, where patches are tiled and interleaved with newline separator tokens. There the gain vector is applied to the language-model input embeddings at the recorded visual-token positions instead. This variant is position-exact for any architecture and is what we fall back to whenever the merged visual block cannot be recovered from the projector alone.

In both cases the visual-token positions are recovered from the merged sequence by locating the contiguous block of image placeholder tokens; the merged visual length is obtained from the length difference between the output logits and the input token sequence. This handles both processing modes found in current implementations: expanded placeholders, where one placeholder token is emitted per visual feature, and the legacy in-forward merge, where a single placeholder is replaced by the whole feature block at run time.

Backbone	Visual tokens	Interface	α	ρ	Masking site
LLaVA-1.5-7B	576	dense spatial (MLP projector)	$+0.15$	0.40	projector output
MiniGPT-4	32	sparse (Q-Former, 32 queries)	-0.15	0.40	projector output
mPLUG-Owl2	64	sparse (visual abstractor)	-0.15	0.40	projector output

Table 5: Per-backbone configuration used in every reported run. We fix ρ=0.40 and $|\alpha|$=0.15 across all backbones. Only the *sign* of α changes with the dense/sparse visual-token regime described in the main paper; no hyperparameter is tuned per model, benchmark, or seed.

B.2 Cost and Necessity of the Probe Prefill

The logit-lens readout of Eq. 1 needs the model's output distribution at every visual position, and those distributions only exist once the entire prompt has been processed. Since the anchor mask must be known *before* the decoding prefill is executed, the probe cannot be folded into that prefill: the mask is applied inside the very forward pass whose logits would otherwise supply the uncertainty scores. DVAD therefore performs one extra batch-1 forward over the prompt, and only then the batch-2 forward that begins generation.

This ordering constraint is worth stating explicitly because it bounds what any uncertainty-guided masking scheme can cost: at least one unmasked pass is unavoidable. The overhead is nonetheless small in practice. A prefill is a single forward over roughly 600 tokens, whereas decoding a caption is several hundred forwards over one token each, so the probe is amortised to about 1.6% of end-to-end latency (Table 4 of the main paper). Every latency we report includes the probe.

B.3 Per-Backbone Configuration

B.4 Decoding Configuration

All methods, ours and the baselines alike, use greedy decoding with a 512-token generation budget, beam size one, fp16 weights, frozen public checkpoints and no weight updates of any kind. The captioning prompt for LLaVA-1.5 is `USER: <image> Describe the image. ASSISTANT:`, and the equivalent prompt is placed in each other backbone's own conversation template so that no method gains an advantage from prompt formatting. Captions are post-processed only by stripping the template prefix and dropping sentences that contain decoder artefacts; identical post-processing is applied to every method before scoring. Benchmark evaluations are distributed across two NVIDIA A100 80GB GPUs; the latency experiment is conducted separately on a single A100 80GB GPU, as detailed in Section F.

C Algorithm

Algorithm 1 gives the complete procedure. Lines 1–8 run once per image and produce the anchor view; lines 10–15 are the decoding loop.

Two properties of the loop are worth making explicit. First, both rows receive the *same* fused logit vector, so greedy selection picks an identical token for each; the two prefixes therefore never diverge and their KV caches stay aligned for the entire generation. Second, because the views differ only in their visual tokens and share everything else, the evidence shift $z_t^{\mathrm{anc}} - z_t^{\mathrm{full}}$ isolates the contribution of the suppressed patches at that step, with the accumulated text context held fixed. This is what lets a single scalar α act as a well-behaved dial rather than a generic logit perturbation.

For the discriminative multiple-choice benchmarks the same fusion is applied to the first generated token only, restricted to the option letter token ids, and the answer is the argmax over that restricted set. No object prior is used there, since a multiple-choice question has no single queried object.

Algorithm 1 DVAD: Signed Dual-View Anchored Decoding

Require: Image I, prompt x, mask ratio ρ, coefficient α

$\tilde{F} \leftarrow \Phi(\text{VisionTower}(I))$ ▷ N projected visual tokens
Run probe prefill over $(\tilde{F}, x)$ ▷ batch 1, extra forward
for $i = 1$ to N **do**
  $p_i \leftarrow \text{Softmax}(W_U h_i^{(L)})$ ▷ logit lens, Eq. 1
end for
$\bar{p} \leftarrow \frac{1}{N} \sum_i p_i$ ▷ visual consensus, Eq. 2
$U_{\text{epi}}(i) \leftarrow \text{KL}(p_i \| \bar{p})$ ▷ Eq. 3
$M_\rho \leftarrow$ indices of the $\lceil \rho N \rceil$ largest U_{epi}
$\tilde{f}_i^{\text{anc}} \leftarrow 0$ if $i \in M_\rho$, else $\tilde{f}_i$ ▷ anchor view, Eq. 4
$y \leftarrow \varnothing$
while last token is not EOS and $|y| < L_{\max}$ **do**
  $(z_t^{\text{full}}, z_t^{\text{anc}}) \leftarrow \text{Forward}([\tilde{F}; \tilde{F}^{\text{anc}}], x, y)$ ▷ one batched forward, two rows
  $z_t \leftarrow z_t^{\text{full}} + \alpha(z_t^{\text{anc}} - z_t^{\text{full}})$ ▷ Eq. 5
  $y \leftarrow y \| \arg\max z_t$ ▷ appended to both rows
end while
return y

Benchmark	Task	Data	Size per run	Metrics
CHAIR	open-ended captioning	COCO val2014	500 images, 3 seeds	CHAIR_S, CHAIR_I
THRONE	open-ended captioning	COCO val2014	same generations	P, R, F_1, $F^{0.5}$
AMBER	open-ended captioning	AMBER generative	500 of 1004, 3 seeds	CHAIR, Cover, Hal, Cog
MMBench	multiple choice (A–D)	MMBench dev	4329 questions	accuracy
MMStar	multiple choice (A–D)	MMStar	1500 questions	accuracy

Table 6: Evaluation protocol. For the three hallucination benchmarks we draw three independently seeded subsets and report mean±standard deviation; within a seed, every method sees exactly the same images and the same prompts. The multiple-choice benchmarks are evaluated on their full splits. No metric in this paper uses an LLM judge or any external API: all scoring is deterministic and runs locally.

D Benchmarks and Evaluation Metrics

We evaluate on three hallucination benchmarks and two general-capability benchmarks. Table 6 summarises the protocol; the paragraphs below define each metric precisely and state exactly how we compute it, since several of these metrics admit more than one implementation in the literature.

D.1 CHAIR

CHAIR Rohrbach et al. (2018) measures object hallucination in free-form captions by comparing the MSCOCO objects a caption mentions against the objects actually present in the image. Mentioned objects are extracted with the benchmark's own synonym list, which maps surface forms onto the 80 MSCOCO categories; ground-truth objects are the union of the instance-segmentation labels and the objects named in the five human reference captions, which is the standard construction and is deliberately generous, so that an object a human annotator found worth mentioning is never counted as a hallucination.

Two rates are reported. The sentence-level rate

$$\text{CHAIR}_S = \frac{|\{\text{captions with} \geq 1 \text{ hallucinated object}\}|}{|\{\text{captions}\}|}$$

is the fraction of captions that contain at least one hallucination, and the instance-level rate

$$\text{CHAIR}_I = \frac{|\{\text{hallucinated object mentions}\}|}{|\{\text{all object mentions}\}|}$$

is the fraction of object mentions that are hallucinated. Lower is better for both. The two behave differently under length changes: a method that simply produces shorter captions lowers CHAIR_S almost mechanically, while CHAIR_I is normalised by the number of mentions and is therefore harder to game. Neither, however, rewards mentioning objects that are present, which is precisely why we pair CHAIR with recall-sensitive metrics rather than reporting it alone.

D.2 THRONE

THRONE Kaul et al. (2024) evaluates Type-I hallucination in free-form generation by scoring, per object class, whether the model's description asserts objects that are absent. It complements CHAIR by being explicitly two-sided: a model that avoids hallucination by describing very little is penalised through recall.

We reuse the same free-form captions generated for CHAIR and swap only the metric, so that CHAIR and THRONE describe the same outputs. For every image, CHAIR already yields the set of MSCOCO objects the model mentioned and the set present in the image, which is exactly what THRONE's class-wise scoring consumes. We accumulate class-wise counts over the 80 categories and compute, for each class c,

$$P_c = \frac{\mathrm{tp}_c}{\mathrm{pred}_c}, \qquad R_c = \frac{\mathrm{tp}_c}{\mathrm{gt}_c},$$
$$F_1{}_c = \frac{2P_cR_c}{P_c + R_c}, \qquad F^{0.5}{}_c = \frac{1.25\,P_cR_c}{0.25\,P_c + R_c},$$

and report the macro-average of each quantity across classes. Classes that are never predicted contribute no precision term and classes never present contribute no recall term; such entries are skipped in the average, matching the reference scorer's treatment of undefined classes. $F^{0.5}$ weights precision above recall and is the metric most directly aligned with hallucination avoidance, while F_1 is the balanced view.

We note one deviation for transparency: the official THRONE pipeline extracts objects from generations with a separate detector-based procedure, whereas we reuse CHAIR's synonym-based extraction so that the two metrics operate on an identical object vocabulary and no additional model is introduced into the evaluation. Our THRONE values are therefore directly comparable *across the methods in this paper*, which are all scored by the identical pipeline on identical samples, but should not be compared numerically against THRONE figures published elsewhere.

D.3 AMBER

AMBER Wang et al. (2023) is an LLM-free hallucination benchmark; we use its generative split and score it with the *official* evaluator, unmodified. The evaluator extracts nouns from a generated description with spaCy (`en_core_web_lg`), maps them through the benchmark's word-association file and a list of globally safe words, and compares the result against per-image annotations of objects that are truly present. Four metrics are reported:

- **CHAIR** (lower is better): the fraction of mentioned objects that are not supported by the image annotation. This is AMBER's own hallucination rate and is computed over a much larger and more fine-grained object vocabulary than MSCOCO's 80 categories.
- **Cover** (higher is better): the fraction of the ground-truth objects that the description actually mentions, in other words the coverage or completeness of the description.
- **Hal** (lower is better): the fraction of responses that contain at least one hallucinated object, the response-level analogue of CHAIR_S.
- **Cog** (lower is better): the fraction of a curated set of *hallucination-prone* target objects, chosen because humans and models tend to co-hallucinate them in these scenes, that the model mentions. It isolates language-prior-driven hallucination from incidental error.

CHAIR and Cover form the central trade-off: any intervention can lower CHAIR by describing less, and Cover is what exposes that. We report all four so that a coverage collapse cannot hide behind an improved hallucination rate. Because the generative split contains 1004 images and our other benchmarks use 500-image subsets, we evaluate AMBER on three seeded 500-image subsamples so that the seed structure matches the rest of the paper.

Method	Family	Extra computation per image	Key hyperparameters
Greedy	–	none	–
Dropout	stochastic ensemble	1 probe prefill + 3 unbatched decode streams	K=3 masked views, majority vote
VCD	contrastive decoding	1 additional decode stream (noised image)	α=1.0, β=0.1, noise step 500
SHIELD	encoder-level defence	30-step adversarial attack per image	published LLaVA settings
DVAD	dual-view fusion (ours)	1 probe prefill + 1 batched two-row decode	α, ρ (Table 5)

Table 7: Baselines and their per-image overhead. The distinction that matters for latency is not the number of logical streams but whether they are batched: DVAD's two views occupy two rows of a single forward pass, whereas Dropout's three masked views are executed sequentially in the reference implementation.

D.4 MMBench and MMStar

These two benchmarks do not measure hallucination; we include them to verify that a decoding intervention aimed at grounding does not quietly degrade general visual understanding, which is the most common hidden cost of hallucination-suppression methods.

MMBench Liu et al. (2024b) covers 20 leaf ability categories spanning perception and reasoning; we use the dev split (4329 questions), which has public answers. MMStar Chen et al. (2024a) contains 1500 questions curated so that they genuinely require the image, filtering out items answerable from text priors alone, grouped into six categories.

Both are multiple choice. We present the question with its lettered options and the instruction to answer with the option letter directly, then take the model's predicted letter and score exact match against the gold letter. For the generative baselines the letter is parsed from a short generated answer; for DVAD it is read directly as the argmax of the fused first-token logits restricted to the option-letter token ids, which is the same fusion rule applied to a single step. Accuracy is the only metric, no judge model or API is involved, and both benchmarks are exact-match by construction.

An important caveat for interpretation: these are single-token answers. The mechanism DVAD targets, grounding attenuation, is by definition an effect that accumulates as the generated text grows, so these benchmarks sit largely outside its operating regime. We report them as a guardrail against regression, not as evidence of benefit.

E Baseline Methods

We compare against one unmodified decoding procedure and three training-free hallucination-mitigation methods, chosen to cover the three families that do not require retraining: stochastic ensembling, contrastive decoding, and encoder-level defence. Table 7 summarises their cost.

E.1 Greedy Decoding

The unmodified backbone with greedy token selection. It is the reference point every method must beat, and also the correct cost baseline: any hallucination-mitigation method that cannot justify its overhead against greedy has limited practical value. Greedy decoding is deterministic, so its results carry no seed variance beyond the image sampling.

E.2 Dropout Decoding

Dropout Decoding Fang et al. (2024) is the method closest to ours in motivation and the one DVAD is most directly positioned against. It also reads per-token epistemic uncertainty from a probe pass, but it treats high-uncertainty visual tokens as candidates for *removal*: it samples K random masked

subsets of visual tokens, decodes K streams, and selects each output token by majority vote over the streams' individual argmax choices. We use $K{=}3$, the setting of the original work.

Two consequences follow, and they motivate DVAD's design. First, the selection is a hard vote over discrete tokens rather than a combination of distributions, so information about how strongly each stream preferred its choice is discarded. Second, the streams are stochastic, which means the output is not reproducible across runs without fixing the mask seed, and the variance we observe for Dropout across seeds is correspondingly larger than for the deterministic methods. In the reference implementation the K streams are not batched, which is why its wall-clock overhead exceeds what the stream count alone would suggest.

E.3 VCD

Visual Contrastive Decoding Leng et al. (2024) contrasts the output distribution obtained from the original image against the one obtained from a diffusion-noised version of the same image, on the premise that tokens which survive severe visual corruption are driven by language priors rather than by visual evidence. The contrasted logits take the form $(1+\alpha)z_{\text{orig}} - \alpha z_{\text{noisy}}$, restricted by an adaptive plausibility constraint parameterised by β that prevents the subtraction from promoting implausible tokens. We use the published settings $\alpha{=}1.0$, $\beta{=}0.1$ and noise step 500.

VCD is the natural point of comparison for the *sign* of DVAD's fusion coefficient. Both methods combine two streams that differ only in their visual input, but VCD always subtracts, because its second view is a globally degraded image that carries no useful evidence. Our anchor view is different in kind: it is a selectively degraded image that retains most of its content, so whether to move toward or away from it is a genuine question, and the answer depends on the visual interface rather than being fixed in advance.

E.4 SHIELD

SHIELD Huang et al. (2025) attacks the problem at the vision encoder rather than at the decoder. For each image it runs a multi-step adversarial optimisation, a 30-step CW attack in the published configuration, to identify visual features to which the model's output is unstably sensitive, and applies a bias correction derived from that analysis. It is the strongest baseline on its native backbone and the most expensive, since the per-image attack is inherently iterative and cannot be amortised.

SHIELD is published for LLaVA only. To include it on the other two backbones we port its mechanism, the CW-attack contrastive view together with the visual-feature bias correction, onto each backbone and retain its published LLaVA hyperparameters, since no per-backbone values are provided. We flag this explicitly because the resulting numbers should be read as a faithful port rather than as an author-tuned configuration, and any weakness observed off LLaVA may reflect the absence of tuning as much as the method itself.

F BACKBONES

The main comparison uses three backbones chosen to span the architectural distinction that DVAD's sign rule depends on.

LLaVA-1.5-7B Liu et al. (2024a) connects a CLIP vision transformer to a Vicuna language model through a small MLP projector, producing 576 spatially localised visual tokens with an essentially one-to-one correspondence between tokens and image patches. It is the canonical dense-token interface, and the one on which a high-uncertainty token is most likely to be a genuine local outlier.

MiniGPT-4 Zhu et al. (2024) places a Q-Former between a frozen visual encoder and the language model, compressing the whole image into 32 learned query tokens. Each token is a global, learned aggregate rather than a patch, so the notion of a spatially localised outlier does not straightforwardly apply.

mPLUG-Owl2 Ye et al. (2024) uses a visual abstractor to compress the image into 64 tokens and a modality-adaptive module inside the language model. Like MiniGPT-4 it is a compressed interface, though less aggressively so.

The 576 / 32 / 64 token counts are the concrete reason we expect the sign of α to differ. Suppressing one of 576 patch tokens removes a small, local piece of evidence; suppressing one of 32 query tokens removes a substantial fraction of everything the language model can see about the image. Token count is a useful indicator of this distinction rather than a proven criterion, and we say so in Section G.

G Inference-Efficiency Protocol

Latency is measured on 50 COCO val2014 images with a 512-token generation budget, batch size one, fp16, on a single A100-80GB. The timer covers the complete per-image loop, that is the probe prefill, the decoding prefill and every decode step, and excludes only the one-time model loading that all methods share. Every method is timed in the same process configuration on the same image list, so the comparison is like-for-like.

In Table 4 of the main paper the Passes column is written $p{+}v$, where p counts logit-lens probe prefills and v counts decode streams. DVAD's entry of $1{+}2$ therefore means one probe prefill followed by two views decoded as a single batched stream, not three sequential passes. This notation is deliberately conservative about our own cost: it counts the probe as a full pass even though it is one forward amortised over hundreds of decode steps. Conversely, Dropout's $1{+}3$ understates its cost, because those three streams are executed sequentially in the reference implementation rather than batched, which is why its measured overhead exceeds its stream count. The ordering in that table is governed by stream count and by SHIELD's per-image attack, and we found it stable across generation lengths and batch sizes.

H Limitations

- **The sign of α must be chosen per backbone.** We keep $|\alpha|{=}0.15$ and $\rho{=}0.40$ fixed for every backbone, but the sign is a single discrete choice determined by the visual interface and fixed before evaluation: applying the dense-regime sign to a sparse-token backbone, or the reverse, is worse than plain greedy decoding. Token count is a useful indicator of the regime rather than a validated boundary, and we have not tested where that boundary actually lies.
- **The uncertainty score selects the anchor but is not, by itself, the whole story on dense-token backbones.** What the fusion needs is a systematically degraded second view to contrast against; the epistemic score determines which tokens form that view. On a dense interface, where many patches are individually redundant, alternative selection criteria behave comparably. We use the KL score because it is parameter-free, deterministic, and because it is the signal whose sign-dependent behaviour distinguishes the two regimes in the first place.
- **Backbones with little hallucination gain little.** On a model whose greedy captions are already strongly grounded there is limited headroom, and DVAD's effect shrinks accordingly. Its benefit scales with how much hallucination the backbone actually exhibits.
- **Single-token benchmarks are outside the operating regime.** DVAD counteracts an effect that accumulates over long generations. In a one-token multiple-choice answer there is no accumulated context for the fusion to correct against, so suppressing patches can only remove information.
- **Large α degenerates.** The coefficient is a monotonic faithfulness/coverage dial, and pushing it far past our default drives hallucination metrics down dramatically while recall collapses, because the model stops committing to objects at all. The metric-optimal setting is therefore not a good operating point, and we deliberately choose a moderate value.
- **Evaluation scope.** All hallucination results concern object-existence hallucination in English captioning of natural photographs. We do not evaluate attribute or relation hallucination, multi-image settings, or reasoning chains, all of which recent benchmarks show to be distinct failure modes.

Additional qualitative results. Figures 7 to 10 present additional qualitative comparisons between greedy decoding and DVAD on diverse COCO images. Across indoor scenes, outdoor environments,

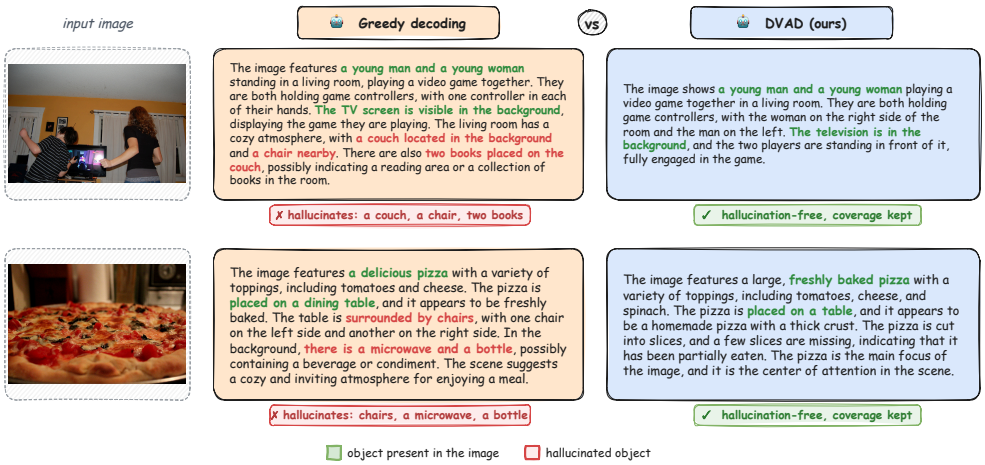

Figure 7: Qualitative comparison between greedy decoding and DVAD with LLaVA-1.5 on two COCO images. **Green** marks correct descriptions, while **red** marks hallucinations.

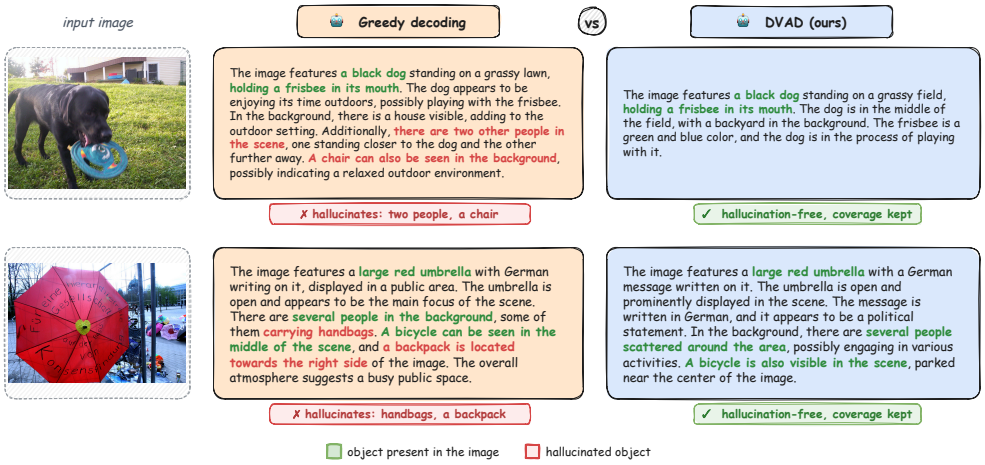

Figure 8: Qualitative comparison between greedy decoding and DVAD with LLaVA-1.5 on two COCO images. **Green** marks correct descriptions, while **red** marks hallucinations.

food images, and transportation settings, greedy decoding frequently introduces plausible but unsupported objects. Examples include a couch, chair, and books in the video-game scene; people and a chair in the dog scene; people and a truck in the airport scene; and a knife and spoon in the dessert scene. In contrast, DVAD consistently suppresses these hallucinated objects while retaining the main visually grounded content, such as the people and game controllers, the dog and frisbee, the airplanes, and the cake, plate, and fork. These examples further demonstrate that DVAD is effective across a broad range of object categories and scene compositions, rather than being limited to the cases shown in the main paper.

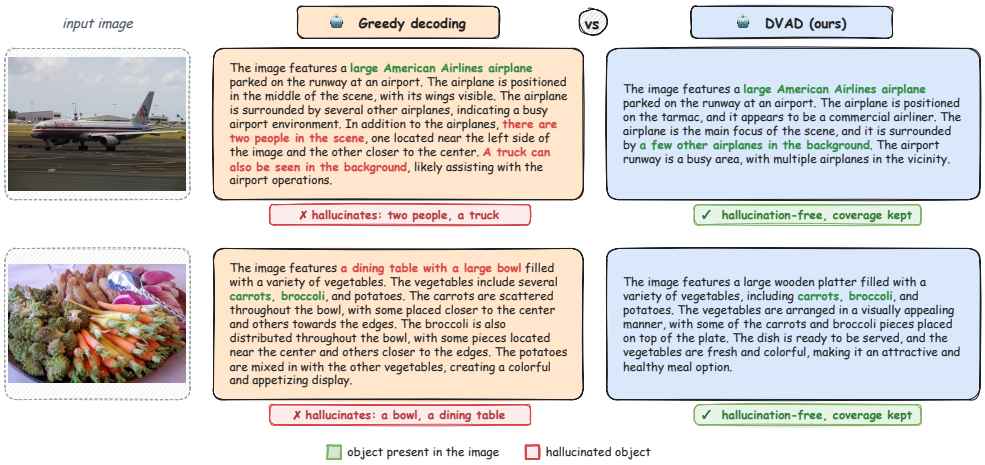

Figure 9: Qualitative comparison between greedy decoding and DVAD with LLaVA-1.5 on two COCO images. **Green** marks correct descriptions, while **red** marks hallucinations.

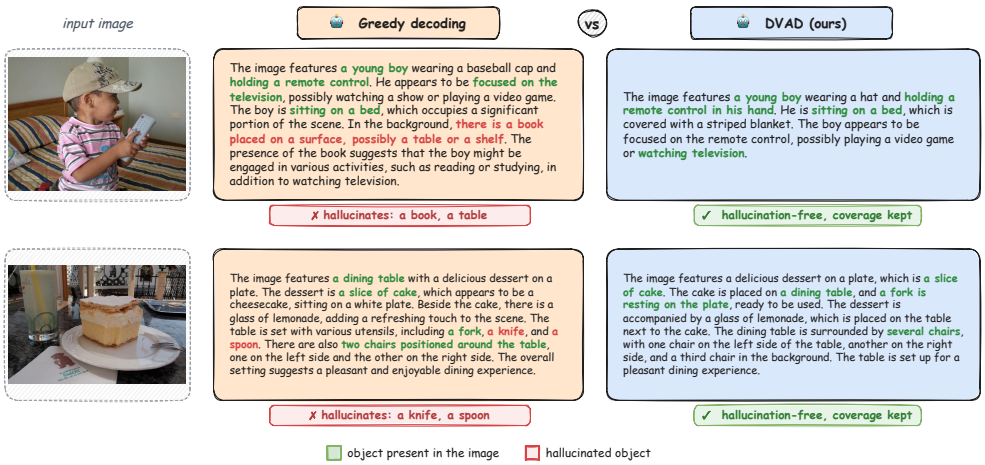

Figure 10: Qualitative comparison between greedy decoding and DVAD with LLaVA-1.5 on two COCO images. **Green** marks correct descriptions, while **red** marks hallucinations.