PAVE: Prefill-Conditioned Activation Editing for Hallucination Mitigation in LVLMs

Jingmin Zhu[1] Junae Kim[2] Dinh Phung[1]
Trung Le[1] Jianfei Cai[1] Qiuhong Ke[1*]
[1]Department of Data Science & AI, Monash University, Australia
[2]Defence Science and Technology Group, Australia

Abstract

Large vision–language models (LVLMs) can caption images, answer visual questions, and reason over scientific diagrams, yet they still produce hallucinations unsupported by visual evidence. Existing approaches mitigate hallucinations either by reshaping the decoding distribution or by steering hidden activations with calibration-based directions. The former adjusts token probabilities but operates only at the output level, whereas hallucinations often arise when language priors dominate visual evidence in intermediate representations. The latter applies a fixed calibration intervention, which may fail to capture image-specific hallucination drift or inadvertently suppress grounded evidence. To address these limitations, we propose **PAVE**, a training-free method for input-specific hidden-state intervention. PAVE constructs offline subspaces for hallucination-drift suppression and visual-evidence amplification, and adapts the hidden-state update to each input using a per-image visual basis computed during prefill. By editing hidden states with this prefill-conditioned subspace, PAVE strengthens grounded evidence while suppressing hallucination drift, without relying solely on decoding-level corrections or fixed activation edits. Experiments on LLaVA-1.5-7B and Qwen-VL show that PAVE substantially reduces hallucinations on CHAIR and POPE while preserving grounded utility on MME.

1 Introduction

Large vision-language models [23, 24, 2, 7] can caption photographs, answer questions about scientific diagrams, and follow grounded instructions in long multimodal contexts [44]. Their capabilities have transferred strongly from text-only backbones. However, a basic failure mode persists. A model may correctly describe a scene, yet still introduce plausible content that is not supported by the visual evidence. When this mismatch involves object content, it is commonly studied as object hallucination [18, 29, 22]. As LVLMs are deployed in clinical imaging, autonomous systems, and consumer assistants [37, 42], such errors become more than harmless captioning mistakes. They expose a core reliability gap in grounded generation.

At each decoding step, the hidden state mixes language priors with image-specific evidence. Hallucination arises when prior-driven content dominates the grounded signal and pulls generation toward plausible but unsupported objects [19, 12]. To address this, one approach is to correct the model during decoding. Existing mitigation methods mostly follow two routes. The first route modifies the decoding output distribution during inference [17], for example by contrasting perturbed visual conditions, decoder layers, attention states, or calibrated distributions [12, 10, 6, 33, 35, 43]. These methods directly adjust token scores and are often training-free. However, methods that rely on contrastive or beam-based correction typically add forward-pass or beam-search overhead. Moreover, because correction is applied at the token-score level [14], they repeatedly adjust the output

*Corresponding author: Qiuhong.Ke@monash.edu.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

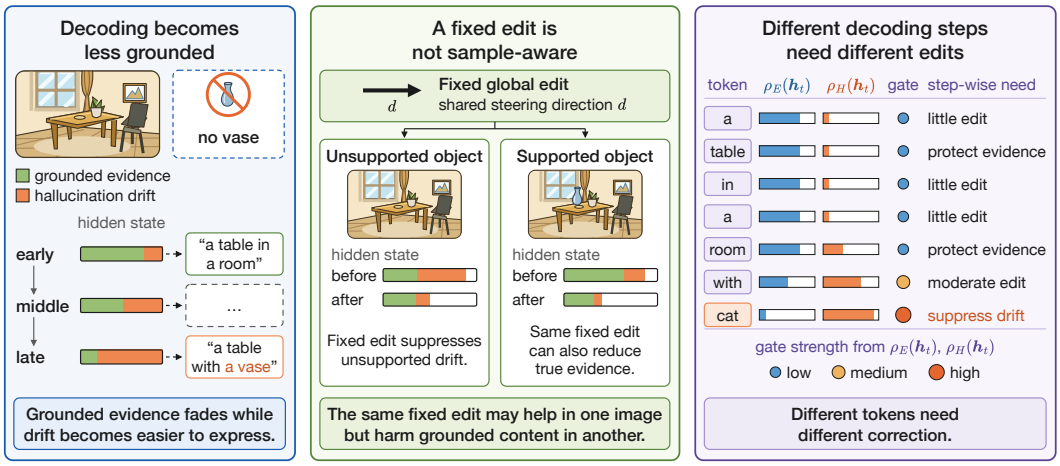


Figure 1: **PAVE motivation. Left:** Grounded evidence (green) fades while hallucination drift (orange) accumulates as decoding proceeds. The model can drift from "a table in a room" to "a table with a vase". **Middle:** A fixed steering direction shared across images is not input-specific. It suppresses drift when the object is unsupported, but reduces true evidence when the same object is in fact grounded. **Right:** Evidence ratio $\rho_E(\boldsymbol{h}_t)$ and drift ratio $\rho_H(\boldsymbol{h}_t)$ vary substantially across decoding steps. Gate strength must therefore adapt step-wise, low on grounded function-word steps and high only where evidence needs protection or drift needs suppression.

distribution rather than directly editing the visual-evidence and language-prior components in the hidden state. This makes the intervention less effective at addressing the internal evidence-prior imbalance that drives hallucinations, allowing such errors to persist across the decoding process. The second route steers hidden activations with directions or subspaces estimated from calibration data [38, 4, 20, 25, 19]. These methods act on the representation used for prediction, but the edit is usually fixed across inputs. Because each image provides its own visual evidence, a fixed calibration edit may fail to capture image-specific hallucination drift or may suppress faithfully grounded content.

This raises a more precise question: what geometry should a hidden-state intervention use at test time? **First**, the intervention should contain both hallucination-drift suppression and evidence amplification in the hidden state. Image evidence is gradually diluted along the residual stream [19], so removing hallucination-prior components without strengthening evidence may leave the decoder with too little grounded signal. **Second**, the intervention should be input-specific. A calibration subspace can capture recurring hallucination patterns, but it cannot fully describe the objects, layout, and evidence of the current image. A fixed prior-suppression direction may suppress unsupported content in some images, but overlap with grounded evidence in others. Conversely, an evidence-amplifying direction may also carry prior-driven content when it is reused without checking the current input. **Third**, the intervention should be step-wise, with the correction magnitude varying across decoding steps according to the local evidence/prior balance. Decoding steps corresponding to content tokens (e.g., object names) and function words (e.g., *the*) do not require the same magnitude of correction, as they differ in the level of available evidence and in the influence of the language prior. The intervention should therefore adapt to the relative contributions of evidence and the language prior at each step of the hidden state during decoding.

To this end, we propose **P**refill-conditioned **A**cti**V**ation **E**diting (**PAVE**) for hallucination mitigation in LVLMs. PAVE performs both evidence amplification and hallucination-drift suppression, adapts the edit to the current input, and determines at each decoding step how strongly to apply evidence amplification and hallucination suppression. To achieve this, PAVE operates in three stages: First, it estimates two offline priors H and E from a shared calibration pool at the same steering layer: H summarizes recurring hallucination drift, and E summarizes recurring evidence directions. Second, it adapts the priors during prefill. The visual-token states already computed for the current image provide a per-image evidence basis, enabling input-specific editing. Third, PAVE edits each decoding hidden step with this prefill-conditioned subspace weighted by step-wise gates. The update strengthens grounded evidence while suppressing hallucination drift, without relying on purely distribution-level correction or applying the same fixed activation edit to every image.

Our contributions are as follows:

- We identify that effective hallucination mitigation requires controlling the relative contributions of grounded evidence and the language prior during decoding, rather than relying solely on distribution-level correction or fixed activation edits.
- We propose PAVE, a training-free hidden-state editing method that adapts hallucination and evidence priors to the input via prefill-conditioned adaptation, enabling input-specific, step-wise updates.
- On POPE and CHAIR, while maintaining grounded utility on MME, PAVE outperforms recent training-free baselines. Ablations confirm the contribution of dual subspaces, prefill-conditioned adaptation, and step-wise control.

2 Related work

Decoding-time mitigation. Object hallucination has been documented through dedicated benchmarks [18, 29, 34, 8, 5] and attributed to language-prior dominance as well as the weakening of image evidence in the residual stream [22, 19]. A large class of training-free methods mitigates this failure by modifying the next-token distribution. VCD [12] contrasts clean and noised visual inputs. ICD [36] contrasts instruction variants. OPERA [10] penalizes over-trusted attention patterns during beam search. DoLa [6] contrasts decoder layers. DECO [33], DCLA [31], and DAMO [35] calibrate image-grounded logits with layer-wise, token-wise, or momentum-based signals. MARINE [43] mixes an image-grounded distribution into the base distribution. These methods differ in the source of their contrast, but their correction is applied after the hidden state has formed. Training-based alternatives use robust instruction tuning or feedback alignment [21, 30, 41]. PAVE instead edits the residual stream before logits are produced at each step, so the correction acts on the state that will continue through the decoder.

Hidden-state activation editing. Activation editing provides a more direct interface for hallucination control. Prior work shows that model behavior can often be steered through low-dimensional residual-stream directions [16, 46]. Nullu [38] removes an offline hallucination subspace from MLP weights. ICT [4] estimates an inference-time direction from contrastive responses. VTI [25] and VISTA [19] extract per-sample steering vectors from contrastive visual states. HulluEdit [20] constructs an evidence-aware anti-prior subspace. DMAS [40] retrieves vectors from a dual pool and applies them with a shared strength. These methods establish that residual-stream editing is effective, but they often bind suppression and amplification to a single direction, a fixed projection, or a uniformly weighted vector set. PAVE keeps the two roles separate. It estimates hallucination and evidence priors from different contrasts, adapts them with prefill evidence tokens from the current image, and uses step-wise energy gates to decide how strongly each side should act at each decoding step.

3 Method

PAVE is a training-free method that edits the decoder hidden state during generation. It is built around one design principle: hallucination mitigation should not use the same direction for every image and every step. Instead, it should adaptively combine stable offline priors, i.e., a hallucination prior H and an evidence prior E, conditioned on the current input, and apply corrections only when the hidden state signals a need for intervention, with the magnitude of the correction scaled by the strength of that signal. The method has three components. First, PAVE estimates two offline priors from contrastive calibration data: a hallucination prior H capturing recurring drift patterns, and an evidence prior E summarizing recurring grounded directions. Second, it adapts these priors to the current image using prefill visual-token states. Third, it applies a step-wise energy-gated update during decoding.

3.1 Preliminaries

Steering layer. Let the LVLM decoder have hidden dimension d. PAVE edits the residual stream at one steering layer ℓ_s. The same layer is used for offline calibration, prefill adaptation, and decoding-time intervention. This keeps all priors, evidence bases, and decoding states in the same representation space, so their geometry can be composed directly.

The steering layer is chosen from a mid-to-late region of the decoder. Earlier layers retain stronger visual signals, but hallucination-related drift may not yet be clearly expressed. Later layers expose

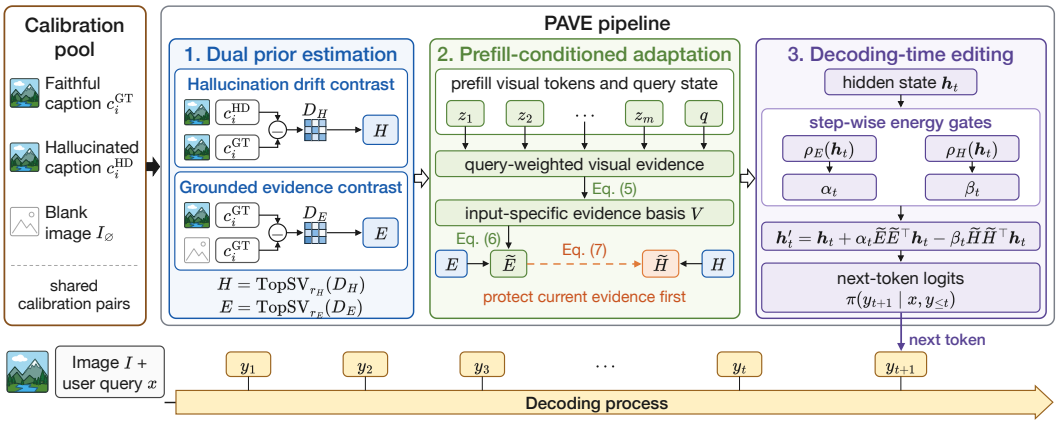


Figure 2: **PAVE pipeline. Left:** Offline calibration estimates a hallucination prior H and an evidence prior E from a shared calibration pool. **Middle:** Prefill visual-token states yield a per-image basis V, which expands the evidence side into $\widetilde{E}$ and residualizes the hallucination side into $\widetilde{H}$. **Right:** Each decoding hidden state is edited by amplifying its evidence projection and suppressing its hallucination projection with step-wise energy gates.

language-prior effects more directly, but image evidence is often weaker [6, 33, 11, 19]. A mid-to-late layer gives a practical compromise where both components remain measurable; Section 4 also analyzes this choice.

Offline caption states. Offline calibration uses teacher-forced captions. For image I and caption c, we read the residual hidden state $\boldsymbol{h}_{\text{last}}^{(\ell_s)}(I, c) \in \mathbb{R}^d$ at layer ℓ_s at the last text position of the caption (the position immediately before the end-of-sequence marker); this position aggregates contextual information from all preceding tokens via causal attention [46], so a single state suffices to summarize (I, c) for calibration. It is separate from the query state $\boldsymbol{q}$, which is used online to condition the current input.

Online states. During prefill, the model has already computed visual-token hidden states for the current image, which PAVE reuses without an additional forward pass. We write the visual-token states at layer ℓ_s as $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_m \in \mathbb{R}^d$, and the hidden state at the last prompt token (the prefill boundary position, which has aggregated all preceding visual and textual tokens through causal attention [46]) as $\boldsymbol{q} \in \mathbb{R}^d$. The visual-token states provide candidate evidence directions, while $\boldsymbol{q}$ summarizes prompt-level information and indicates which visual tokens the next decoded token will attend to under causal attention [32, 46, 19]. During decoding, $\boldsymbol{h}_t \in \mathbb{R}^d$ denotes the residual hidden state at decoding step $t \in \{1, \ldots, T\}$, where T is the caption length.

Notation. For an orthonormal basis U, the matrix $UU^\top$ denotes the orthogonal projector onto its span. We use $\mathrm{orth}(\cdot)$ to denote orthonormalization after removing empty directions, $\boldsymbol{I}$ for the identity matrix, and $\|\cdot\|$ for the Euclidean norm. $\mathrm{TopSV}_r(M)$ returns the top-r right singular vectors of M, and $\mathrm{TopEig}_r(M)$ returns the top-r eigenvectors of a symmetric M.

3.2 PAVE

Dual offline priors. PAVE first builds two offline priors H and E. This step provides a stable estimate of what should be suppressed and what should be preserved before seeing a test image. Unlike a single steering direction, the two-prior design keeps the hallucination prior H and the evidence prior E as separate objects, so suppression and evidence amplification can be controlled independently.

We construct a calibration pool

$$\mathcal{C} = \{(I_i, c_i^{\mathrm{GT}}, c_i^{\mathrm{HD}})\}_{i=1}^N.$$

Each tuple contains an image I_i, a faithful caption c_i^{GT}, and a hallucinated-detail caption c_i^{HD} for the same image. The faithful caption describes image-supported objects, while the hallucinated-detail caption minimally introduces unsupported object content (e.g. c^{GT} "a dog runs across the lawn" versus c^{HD} "a dog and a cat run across the lawn"). Since the image, prompt format, and caption style are kept as consistent as possible, the contrast focuses on hidden-state shifts caused by hallucinated content rather than unrelated caption variation. Appendix B gives the generation and filtering details.

For each calibration tuple, we compute the hallucination drift contrast

$$\boldsymbol{D}_H[i,:] = \boldsymbol{h}^{(\ell_s)}_{\text{last}}(I_i, c_i^{\text{HD}}) - \boldsymbol{h}^{(\ell_s)}_{\text{last}}(I_i, c_i^{\text{GT}}). \tag{1}$$

This shift is not assumed to be a pure language-prior direction. It is a measurable hallucination drift induced by replacing a faithful caption with a hallucinated counterpart under the same image. Across many examples, stable directions of $\boldsymbol{D}_H$ capture recurring ways in which the model represents hallucinated object content.

We also use a blank visual input $I_\varnothing$ with the same faithful caption. This gives a second contrast where the caption is fixed and the visual evidence is removed:

$$\boldsymbol{D}_E[i,:] = \boldsymbol{h}^{(\ell_s)}_{\text{last}}(I_i, c_i^{\text{GT}}) - \boldsymbol{h}^{(\ell_s)}_{\text{last}}(I_\varnothing, c_i^{\text{GT}}). \tag{2}$$

This contrast captures the hidden-state component contributed by visual evidence. Since both contrasts are computed at the same layer, their resulting subspaces can later be combined on the same decoding state.

The offline priors are the leading right-singular-vector subspaces of the two contrast matrices:

$$H = \text{TopSV}_{r_H}(\boldsymbol{D}_H), \qquad E = \text{TopSV}_{r_E}(\boldsymbol{D}_E). \tag{3}$$

The hallucination prior H summarizes recurring hallucination drift, while the evidence prior E captures recurring ways visual input modifies the residual stream, common patterns such as texture, colour, and spatial-layout cues that aggregate across image-caption pairs. Weakly expressed evidence directions in any single image still appear as stable components after pooling; without E, such subtle cues that the per-image basis V does not span risk being lost. Singular-vector truncation keeps directions that are stable across the pool and removes sample-specific noise.

These priors are useful but incomplete. They are estimated before seeing the test image, so they describe the backbone's average behavior rather than the geometry of the current scene. PAVE therefore adapts them using the visual evidence already computed during prefill.

Prefill-conditioned adaptation. The second component turns the offline priors into input-specific subspaces. PAVE first reads the visual geometry of the current image, then uses the offline priors only to fill directions the current image does not already span.

Not all visual tokens are equally relevant to the current generation context. We use the prefill query state $\boldsymbol{q}$ to score relevance because $\boldsymbol{q}$ summarizes the prompt-level information available to the next decoded token. Given $\boldsymbol{q}$ and visual-token states $\{\boldsymbol{z}_j\}_{j=1}^m$, PAVE assigns relevance weights by

$$w_j = \text{softmax}_j\big(\cos(\boldsymbol{z}_j, \boldsymbol{q})\big). \tag{4}$$

The per-sample evidence basis is then the leading eigen-subspace of the query-weighted visual evidence,

$$V = \text{TopEig}_{r_V}\left(\sum_{j=1}^{m} w_j\, \boldsymbol{z}_j \boldsymbol{z}_j^\top\right). \tag{5}$$

Here V is an orthonormal basis for prompt-relevant visual evidence in the current image, providing visual directions read from the prefill states. We use a basis rather than a single weighted mean because relevant evidence can be multi-dimensional, such as multiple objects or object-attribute combinations.

PAVE then makes the current image the anchor of the evidence side. It keeps the input-specific basis V first, and only appends the part of the offline evidence prior that is not already explained by V:

$$\widetilde{E} = \text{orth}\big(\big[V,\ (\boldsymbol{I} - VV^\top)E\big]\big). \tag{6}$$

The first block preserves evidence directions directly supported by the current input image. The second block contributes offline evidence directions outside the current image basis. V comes first because the current image must anchor the evidence side; the offline prior E only fills directions that V does not already span. Thus the offline prior does not overwrite the input-specific evidence geometry, but only completes it.

Only after the evidence side is adapted do we construct the final hallucination suppression basis:

$$\widetilde{H} = \text{orth}\big((\boldsymbol{I} - \widetilde{E}\widetilde{E}^\top)H\big). \tag{7}$$

The order matters: residualizing H against the adapted evidence subspace $\widetilde{E}$ first protects evidence-supported components from being suppressed before any hallucination component is removed. Suppressing H directly would otherwise remove components that the current image actually supports. As a result, $\widetilde{H}$ is orthogonal to $\widetilde{E}$ at the intervention layer, so the suppression term does not directly subtract from the protected evidence subspace.

Step-wise gates. The final component applies the adapted subspaces $\widetilde{E}$ and $\widetilde{H}$ during decoding. A fixed intervention strength is overly coarse. Different steps of the hidden state, each corresponding to a token, can vary substantially, with different levels of evidence and hallucination, with some already encoding strong evidence in the adapted subspace (high ρ_E) and others exhibiting little hallucination drift (low ρ_H). For example, when generating a factual statement, the hidden state at a content-bearing token (e.g., an entity or number) may carry strong task-relevant evidence, whereas the hidden state at a function token (e.g., "the" or "is") may exhibit little drift and require minimal intervention. PAVE therefore computes step-wise gates from the current hidden state itself.

Specifically, we measure how much of the hidden state at decoding step t, $\boldsymbol{h}_t$, lies in the adapted evidence and hallucination subspaces:

$$\rho_E(\boldsymbol{h}_t) = \frac{\|\widetilde{E}\widetilde{E}^\top \boldsymbol{h}_t\|^2}{\|\boldsymbol{h}_t\|^2}, \qquad \rho_H(\boldsymbol{h}_t) = \frac{\|\widetilde{H}\widetilde{H}^\top \boldsymbol{h}_t\|^2}{\|\boldsymbol{h}_t\|^2}. \tag{8}$$

Both ratios are read directly from $\boldsymbol{h}_t$ at decoding time; PAVE never sees the chair-parser verdict for the token being generated. The two gates react to ρ_E and ρ_H in opposite directions, and we adopt the rational form

$$\alpha(\boldsymbol{h}_t) = \min\left(\alpha_{\max},\ \alpha_0 \frac{1-\rho_E(\boldsymbol{h}_t)}{\rho_E(\boldsymbol{h}_t)}\right), \qquad \beta(\boldsymbol{h}_t) = \min\left(\beta_{\max},\ \beta_0 \frac{\rho_H(\boldsymbol{h}_t)}{1-\rho_H(\boldsymbol{h}_t)}\right). \tag{9}$$

The four limiting cases of Eq. 9 capture the intent of the design. As $\rho_E \to 1$, $\alpha \to 0$ and the evidence side of the update vanishes; as $\rho_E \to 0$, α saturates at $\alpha_{\max}$ and the evidence projection is amplified at full strength. Symmetrically, as $\rho_H \to 0$, $\beta \to 0$ and the suppression term is silent; as $\rho_H \to 1$, β saturates at $\beta_{\max}$ and the hallucination projection is suppressed at the capped strength. The linear-regime gains α_0, β_0 are fixed scalars; the caps $\alpha_{\max}, \beta_{\max}$ bound the per-step edit norm. Appendix C.2 lists the four values.

The edited hidden state is

$$\boldsymbol{h}'_t = \boldsymbol{h}_t + \alpha(\boldsymbol{h}_t)\widetilde{E}\widetilde{E}^\top \boldsymbol{h}_t - \beta(\boldsymbol{h}_t)\widetilde{H}\widetilde{H}^\top \boldsymbol{h}_t. \tag{10}$$

The evidence term $\widetilde{E}\widetilde{E}^\top \boldsymbol{h}_t$ strengthens the adapted evidence component, while the hallucination term $\widetilde{H}\widetilde{H}^\top \boldsymbol{h}_t$ weakens the adapted hallucination component. Since $\widetilde{E}$ and $\widetilde{H}$ are orthogonal at layer ℓ_s, these two terms act on separated coordinates in the intervention layer. The remaining component of $\boldsymbol{h}_t$ is left unchanged by the linear update. Appendix C.1 derives these within-layer properties.

4 Experiments

We use three research questions to investigate PAVE's effectiveness and the mechanism behind it. RQ1 asks whether PAVE reduces object hallucination under both discriminative and open-ended evaluation, addressed by POPE and CHAIR in Section 4.2. RQ2 asks whether the edit preserves grounded utility and avoids shortcut behavior, addressed by MME, the length / object-mention / precision audit, and long-caption robustness in Section 4.3. RQ3 asks which components and mechanisms explain the gain, addressed by the component ablation, hyperparameter sensitivity, qualitative case studies, and mechanism and calibration analyses in Section 4.4.

4.1 Experiment setup

Backbones. We evaluate on LLaVA-1.5-7B [24], a 7B instruction-tuned model with a CLIP-ViT-L/14 encoder [28], and Qwen-VL [2], a 9.6B model with a ViT-bigG stack. Both backbones are frozen, and inference follows the protocol of DMAS [40] and VTI [25]; full settings are provided in Appendix B. Appendix A extends evaluation to four additional backbones, AMBER, MIHBench, and MMHal-Bench.

Benchmarks. We evaluate on five suites spanning hallucination, utility, and out-of-distribution VQA. POPE [18] probes object existence with yes/no questions on MSCOCO and reports accuracy, precision, recall, and F1 averaged over the random, popular, and adversarial splits. CHAIR [29] scores open-ended captions on 500 MSCOCO images, with CHAIR_S for sentence-level and CHAIR_I for per-mention hallucination rates. MME [8] is the object-oriented utility benchmark; we report its existence, count, position, and color subsets and the four-split total. SCIENCEQA [26] and VIQUAE [13] are multimodal VQA suites that lie outside the MSCOCO captioning distribution used to estimate H and E. For these five suites, we average results over three seeds (0, 1, 42); per-benchmark protocols are provided in Appendix B.2.

Baselines. We compare against three families of baselines. The regular decoder is the unmodified greedy decoding pass; contrastive decoding methods reshape the next-token distribution and include VCD [12], OPERA [10], and AGLA [1]; activation-level steering methods modify internal model states (e.g., the residual stream, attention, or MLP weights) and include VTI [25], ICT [4], DECO [33], VAF [39], and DMAS [40].

Implementation details. PAVE estimates the offline priors H and E from a single 100-image MSCOCO calibration pool, disjoint from CHAIR-500, POPE, and MME images. We set the steering layer to $\ell_s = 26$ for LLaVA-1.5-7B and $\ell_s = 28$ for Qwen-VL; all other hyperparameters are tied across backbones, with $r_V = 8$, $r_E = 20$, $r_H = 8$, $(\alpha_0, \beta_0) = (0.5, 0.22)$. Further details are provided in Appendix B.

4.2 Main hallucination results: POPE and CHAIR

Results on POPE. Table 1 reports the discriminative part of RQ1. PAVE is the only method that leads both backbones on accuracy and F1, reaching 87.86 on LLaVA-1.5 and 88.77 on Qwen-VL (1.12 points above the matched DMAS row). Accuracy and F1 move in step on both backbones, ruling out a threshold shift, and the precision–recall pair leans toward higher recall rather than tighter precision. This is consistent with PAVE amplifying the evidence projection of the hidden state and recovering true positives that the regular decoder underweights, rather than tightening the yes/no decision rule.

Table 1: **Results on POPE.** NR: result not reported; baseline evaluation is described in Appendix A.5. Best results are in bold, and second-best values are underlined.

	LLaVA-1.5				Qwen-VL			
Method	Accuracy↑	Precision↑	Recall↑	F1↑	Accuracy↑	Precision↑	Recall↑	F1↑
Regular	81.38	88.04	72.78	79.65	83.71	93.30	72.69	81.70
VCD	84.33	85.93	83.28	84.52	86.67	90.66	81.94	83.04
OPERA	84.21	88.23	79.79	83.72	84.26	<u>94.40</u>	73.52	82.65
AGLA	85.82	**93.78**	76.83	84.44	83.90	**96.20**	70.62	81.44
VTI	86.48	<u>90.11</u>	82.09	85.90	85.18	91.31	78.18	84.08
ICT	<u>87.35</u>	NR	NR	<u>87.12</u>	87.53	90.99	83.31	86.98
DMAS	86.81	87.23	<u>86.57</u>	86.79	<u>87.63</u>	87.92	**87.30**	<u>87.65</u>
PAVE	**87.90**	86.94	**88.81**	**87.86**	**88.37**	92.44	<u>85.47</u>	**88.77**

Results on CHAIR. Table 2 reports the open-ended part of RQ1. On LLaVA-1.5, PAVE reduces CHAIR_S by more than half to 21.6, outperforming DMAS by 9.2 points, a substantial improvement relative to prior work on this benchmark. Qwen-VL shows the same direction at a smaller magnitude, with CHAIR_S falling to 25.2 and CHAIR_I tracking CHAIR_S on both backbones, so a handful of outlier captions cannot account for the average. Both backbones improve, suggesting PAVE generalizes across visual encoders.

Table 2: **Results on CHAIR.** NR: comparable result unavailable. Best results are in bold, and second-best values are underlined.

	LLaVA-1.5		Qwen-VL	
Method	$\text{CHAIR}_S \downarrow$	$\text{CHAIR}_I \downarrow$	$\text{CHAIR}_S \downarrow$	$\text{CHAIR}_I \downarrow$
Regular	51.0	15.2	46.4	14.2
VCD	51.0	14.9	45.7	13.9
OPERA	47.0	14.6	44.8	13.6
DECO	37.8	<u>11.1</u>	NR	NR
AGLA	43.0	14.1	NR	NR
VTI	35.8	<u>11.1</u>	NR	NR
DMAS	<u>30.8</u>	11.4	<u>30.7</u>	<u>11.5</u>
PAVE	**21.6**	**9.7**	**25.2**	**10.8**

POPE by split. Splitting the LLaVA-1.5 POPE average by sampling strategy (Table 13 in Appendix B.4) shows that the margin of PAVE over DMAS widens from Random to Popular to Adversarial, reaching 2.57 F1 points on the split whose absent

objects frequently co-occur with the objects present. On Adversarial, PAVE also attains the highest precision and the highest recall together, so this gain does not come from a tendency to answer "yes".

4.3 Utility and no-shortcut checks

Results on MME. Table 3 reports the utility part of RQ2. MME splits expose what an edit actually costs: an edit that strips visual content shows up as a split collapse rather than a uniform total drop. PAVE leads or ties on existence and position, remains competitive on count and color, and the total (662.43 on LLaVA-1.5, 654.33 on Qwen-VL) exceeds the matched DMAS row on both. The split pattern matches the design: V is built from prefill object evidence, so the edit reinforces grounded directions while leaving count and color untouched, rather than trading utility for hallucination reduction.

Table 3: **Results on MME.** Best results are in bold, and second-best values are underlined.

	LLaVA-1.5					Qwen-VL				
Method	Exist.↑	Count↑	Pos.↑	Color↑	Total↑	Exist.↑	Count↑	Pos.↑	Color↑	Total↑
Regular	175.67	124.67	114.00	151.00	565.33	155.00	127.67	<u>131.67</u>	173.00	587.33
VCD	184.66	138.33	128.67	153.00	604.66	156.00	131.00	128.00	181.67	596.67
VAF	**195.00**	<u>158.33</u>	128.33	155.00	636.67	165.00	**155.00**	**133.33**	175.00	628.33
ICT	<u>190.00</u>	**160.43**	128.67	170.00	649.10	<u>180.00</u>	145.00	108.33	173.33	606.66
DMAS	**195.00**	<u>158.33</u>	<u>133.33</u>	**173.33**	<u>659.99</u>	170.00	145.00	**133.33**	**185.00**	<u>633.33</u>
PAVE	**195.00**	**160.43**	**135.00**	<u>172.00</u>	**662.43**	**185.00**	<u>153.00</u>	**133.33**	<u>183.00</u>	**654.33**

No-shortcut audit. Figure 3 rules out the three obvious explanations for the CHAIR gain. PAVE captions stay within a few words of the regular decoder, produce more object mentions per caption, and report those at 90.3% precision against 84.8% for the regular decoder; a length, abstention, or refusal shortcut would move at least one of the three the wrong way. Object recall decreases modestly from 68.60% to 67.47%; Appendix A.4 reports the coverage trade-off.

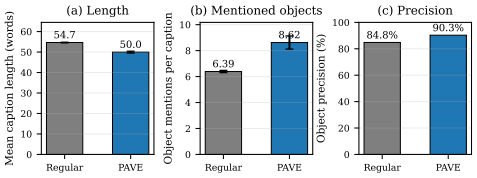


Figure 3: **No-shortcut audit.** Length, object mentions per caption, and per-mention precision on the CHAIR-500 pool.

Generality across VQA tasks. Table 4 stress-tests transfer to two task formats unseen by the calibration pool: SCIENCEQA is multiple-choice subject reasoning, VIQUAE is knowledge-based extraction over named entities. Both lie outside the MSCOCO captioning distribution that produced H and E, so they exercise the prefill-conditioned slice of PAVE: at test time only V is rebuilt from the prompt's image tokens, while the calibration priors stay frozen. PAVE leads every column on both backbones, indicating that the evidence subspace transfers without task-specific recalibration and that suppressing H does not erase the visual cues a grounded answer needs.

Table 4: **Generality on ScienceQA and ViQuAE.** Best results are in bold, and second-best values are underlined.

	ScienceQA		ViQuAE	
Method	LLaVA	Qwen-VL	LLaVA	Qwen-VL
Regular	52.75	46.41	43.38	50.09
VTI	51.46	45.12	42.29	49.00
ICT	52.95	46.61	42.47	49.18
DMAS	<u>62.27</u>	<u>48.04</u>	<u>56.00</u>	<u>54.08</u>
PAVE	**64.63**	**53.40**	**59.26**	**57.10**

4.4 Ablations and analysis

Component ablation. Table 5 isolates four design roles, with Regular as the base decoder and PAVE full as the headline configuration. **Geometry.** The per-image basis V does most of the heavy lifting: replacing V with the offline pool basis loses 12.9 CHAIR points and 52 MME points, and even keeping E, H intact while freezing the prefill update still leaves an 11.2-point CHAIR gap, ruling out a generic low-rank denoising story. **Protection.** Residualizing H against $\widetilde{E}$ is the single most utility-critical step: removing the evidence fusion that builds $\widetilde{E}$ leaves a weaker evidence side and pushes CHAIR_S up to 27.9. Directly suppressing H without first protecting it against $\widetilde{E}$ degrades both objectives, with a particularly severe utility drop: CHAIR_S rises to 29.2 and MME falls to 540.0, consistent with drift directions partly co-aligning with grounded object evidence. **Gating.** Constant gates over-edit easy steps and under-correct hallucinated ones (CHAIR_S 28.4); removing

either side individually shows complementarity, with suppression-only reaching 24.9 on CHAIR but losing 52 MME points, while evidence-only stays at 32.0 because amplification cannot substitute for suppression. **Cost / control.** Recomputing V at every step pushes CHAIR_S to 18.7 at $1.27\times$ runtime but lowers MME to 651.0; we keep the prefill-cached default as the headline because it preserves higher utility and lower latency. A random rank-matched subspace stays near Regular at 48.8, so the gain comes from the specific evidence and drift subspaces, not from generic low-rank residual editing.

Table 5: **Component ablation.** The PAVE row is the headline configuration. Lat. is MME-object runtime relative to Regular (0.138 s/sample); Mem. is peak allocated GPU memory in GB.

Group	Variant	$\text{CHAIR}_S \downarrow$	MME↑	Lat.	Mem.	Diagnosis
Baseline	Regular	51.0	565.3	$1.000\times$	13.52	base decoder
Full	**PAVE full**	21.6	662.4	$1.162\times$	13.71	prefill-conditioned edit
Geometry	Fixed global edit	34.5	610.0	$1.067\times$	13.71	no per-image basis V
Geometry	No prefill adaptation	32.8	615.0	$1.085\times$	13.69	calibration geometry only
Protection	No evidence fusion $\widetilde{E}$	27.9	612.0	$1.173\times$	13.70	weak evidence side
Protection	Suppress H directly	29.2	540.0	$1.109\times$	13.71	suppression overlaps evidence
Gating	No step-wise gates	28.4	643.0	$1.068\times$	13.71	constant strength
Gating	Suppression-only	24.9	610.0	$1.047\times$	13.67	no evidence boost
Gating	Evidence-only	32.0	625.0	$1.070\times$	13.71	no drift removal
Cost	Recompute V each step	18.7	651.0	$1.270\times$	13.74	higher cost
Control	Random subspace	48.8	570.0	$1.035\times$	13.70	negative control

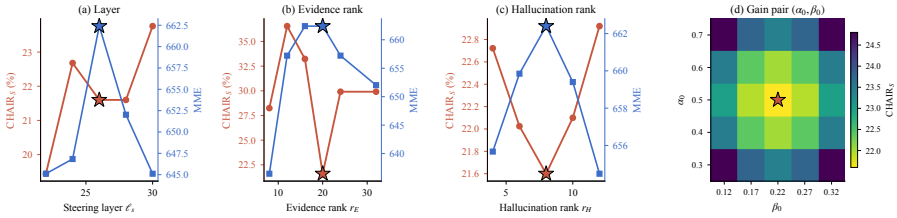


Figure 4: **Hyperparameter sensitivity on CHAIR_S and MME.** Each panel sweeps one knob with the others fixed at the headline operating point (⋆). Red curves: $\text{CHAIR}_S\downarrow$; blue curves: MME↑.

Qualitative cases. Figure 5 overlays both energies on two CHAIR-500 captions disjoint from calibration. In the top row, regular ρ_E drops to $\sim 10\%$ at the object position, while PAVE maintains ρ_E near 50%, preserving the gap that V is designed to recover. In the bottom row regular ρ_H spikes to $\sim 22\%$ and PAVE damps it to $\sim 19\%$. The two energies move in different directions across cases, consistent with the step-wise α- and β-gates of Section 3.2.

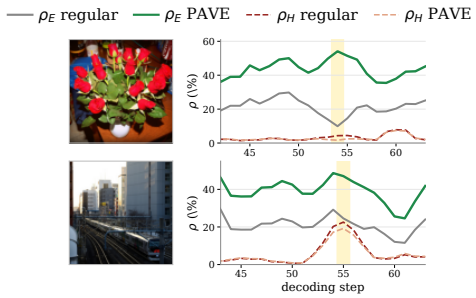
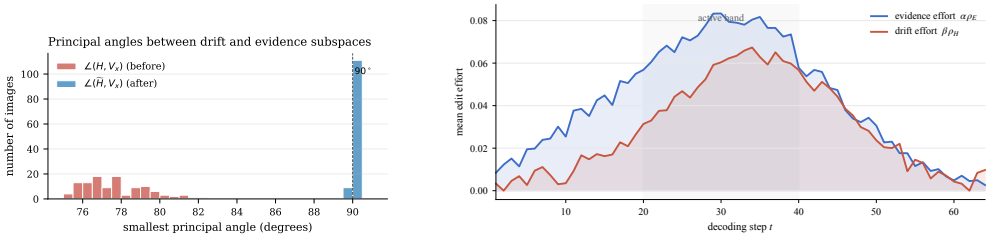


Figure 5: **Qualitative cases.** Per-token ρ_E (solid) and ρ_H (dashed) on two MSCOCO captions; the highlighted column marks the object position.

Hyperparameter sensitivity. Figure 4 sweeps the four knobs PAVE exposes at test time. **Steering layer ℓ_s.** Earlier layers can reduce CHAIR_S more aggressively but lose MME utility, while very late layers leave little room for correction. The chosen LLaVA layer $\ell_s = 26$ is therefore the Pareto operating point in the sweep, and the Qwen-VL choice $\ell_s = 28$ follows the same mid-to-late rule. **Evidence rank r_E.** r_E sets the dimensionality of $\widetilde{E}$; the curve is non-monotonic with its best point at $r_E = 20$, because smaller ranks can miss stable evidence directions while larger ranks start to absorb image-irrelevant variation and dilute protection. **Hallucination rank r_H.** The drift prior is low-rank by construction, so any $r_H \in \{6, 8, 10\}$ stays within 0.5 of the headline. **Gain pair (α_0, β_0).** The default $(0.5, 0.22)$ lies inside a broad low-CHAIR_S region, and the heatmap shows no sharp CHAIR cliff around the operating point, so the default is robust to small mis-tuning.

Mechanism analysis. Figure 6 checks two properties that the design relies on. The first is the geometry behind the evidence-protected update. Before residualization, the offline drift prior is not



(a) Smallest principal angle with V.

(b) Mean per-step edit effort.

Figure 6: **Mechanism analysis.** (a) Smallest principal angle between the drift subspace and the per-image evidence basis V over 120 MSCOCO inputs, before (H) and after ($\widetilde{H}$) residualization. (b) Mean evidence-amplification effort $\alpha(\boldsymbol{h}_t)\rho_E(\boldsymbol{h}_t)$ and drift-suppression effort $\beta(\boldsymbol{h}_t)\rho_H(\boldsymbol{h}_t)$ at each decoding step, averaged over the CHAIR-500 captions.

orthogonal to the evidence basis of the current image. Its smallest principal angle with V clusters around 77°, a non-negligible alignment, so a fixed suppression of H would also remove part of the evidence that the image supports. This overlap is consistent with the component ablation, where suppressing H directly causes the largest utility loss of all variants. After residualization against $\widetilde{E}$, every principal angle lies within $0.5°$ of 90°, so at the steering layer the suppression term no longer acts on the protected evidence span, and the evidence and drift components of $\boldsymbol{h}_t$ are rescaled independently (Appendix C.1). Later decoder layers can still mix the two components, so this separation holds locally at ℓ_s. A second property concerns when the edit acts. Drift-suppression effort stays near zero over the first tokens of a caption, while evidence-amplification effort rises gradually from the start. Both efforts then climb into a broad middle band around steps 20 to 40, the region where the CHAIR parser flags hallucinated object words on the same captions, and relax toward the end of the caption. The gates therefore concentrate the edit on the steps at which drift appears rather than applying it uniformly, in line with the per-token traces in Figure 5 and the single-caption gate dynamics in Appendix C.2.

Calibration analysis. The offline priors come from a small calibration pool, so Figure 7 examines how much of that pool PAVE needs, and the singular spectra in Appendix B.5 show how concentrated the two contrasts are. CHAIR_S improves quickly up to $N = 100$ pairs and then flattens. Larger pools change it by only a few tenths of a point while MME stays within a narrow band, so the headline pool size sits at the start of the plateau. This saturation is consistent with the spectra. The evidence contrast D_E is concentrated, with its leading 20 directions explaining 87% of the variance, whereas the drift contrast D_H has a longer tail and its leading 8 directions cover 44%. The leading singular vectors of both matrices stabilize after roughly 80 to 100 contrastive pairs (Appendix C.5), which places the headline pool at the point where the retained directions stop changing. How the null visual input of D_E is chosen matters less than how the contrast is paired. Replacing the uniform-grey blank $I_\varnothing$ with per-image Gaussian noise or with the MSCOCO mean image leaves both metrics nearly unchanged. Pairing each image with a mismatched caption instead raises CHAIR_S from 21.6 to 33.2, because the resulting contrast no longer isolates the shift caused by visual evidence (Appendix C.6).

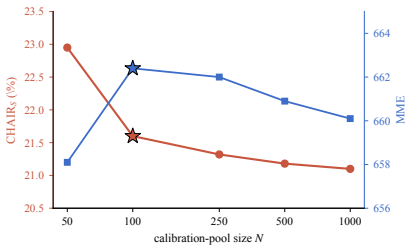


Figure 7: **Calibration-pool size.** CHAIR_S (red) and MME (blue) on LLaVA-1.5-7B when the offline priors are rebuilt from N calibration pairs; $\star$ marks the headline $N = 100$.

5 Conclusion

PAVE introduces a training-free prefill-conditioned hidden-state edit with evidence adaptation, evidence-protected drift suppression, and step-wise gates. Across CHAIR, POPE, MME, ScienceQA, and ViQuAE, our method improves the hallucination–utility trade-off on both backbones, showing that evidence-prior state editing complements output-level correction. **Limitation.** Our method requires white-box residual-stream access and refreshable calibration priors under substantial domain shifts.

Acknowledgments and Disclosure of Funding

This work was supported by the Australian Defence Science and Technology (DST) Group through the Next Generation Technology Fund (NGTF) scheme and the Department of Defence, Australia, via the Advanced Strategic Capabilities Accelerator (ASCA) program. Dinh Phung and Qiuhong Ke further acknowledge the support from the Australian Research Council (ARC) Discovery Projects DP230101176, DP250100262, DP260100218, and DP260101891, the ARC Discovery Early Career Researcher Award (DECRA) funding scheme (Grant No.: DE250100030), and Linkage Project LP250200909. Dr Qiuhong Ke is the recipient of an Australian Research Council Discovery Early Career Researcher Award (project number DE250100030) funded by the Australian Government. The authors are grateful to the anonymous reviewers for their helpful comments. This research was supported by Monash eResearch capabilities, including M3.

References

[1] Wenbin An, Feng Tian, Sicong Leng, Jiahao Nie, Haonan Lin, Qianying Wang, Ping Chen, Xiaoqin Zhang, and Shijian Lu. AGLA: Mitigating object hallucinations in large vision-language models with assembly of global and local attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 29915–29926, 2025.

[2] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-VL: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023.

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

[4] Junzhe Chen, Tianshu Zhang, Shiyu Huang, Yuwei Niu, Linfeng Zhang, Lijie Wen, and Xuming Hu. ICT: Image-object cross-level trusted intervention for mitigating object hallucination in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4209–4221, 2025.

[5] Xuweiyi Chen, Ziqiao Ma, Xuejun Zhang, Sihan Xu, Jianing Yang, David Fouhey, Joyce Chai, and Shengyi Qian. Multi-object hallucination in vision language models. In *Advances in Neural Information Processing Systems*, volume 37, pages 44393–44418, 2024.

[6] Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. DoLa: Decoding by contrasting layers improves factuality in large language models. In *ICLR*, 2024.

[7] 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. *NeurIPS*, 2023.

[8] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, et al. MME: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

[9] Gemma Team et al. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026. URL `https://arxiv.org/abs/2607.02770`.

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

[11] Zhangqi Jiang, Junkai Chen, Beier Zhu, Tingjin Luo, Yankun Shen, and Xu Yang. Devils in middle layers of large vision-language models: Interpreting, detecting and mitigating object hallucinations via attention lens. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 25004–25014, 2025.

[12] 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 (CVPR)*, pages 13872–13882, 2024.

[13] Paul Lerner, Olivier Ferret, Camille Guinaudeau, Hervé Le Borgne, Romaric Besançon, Jose G. Moreno, and Jesus Lovon-Melgarejo. ViQuAE, a dataset for knowledge-based visual question answering about named entities. In *SIGIR*, 2022.

[14] Haiming Li, Yingsheng Liu, Jingmin Zhu, Siyuan Yan, Xieji Li, Jiajun Sun, Zhen Yu, and Zongyuan Ge. Latent ordinal evidence, misaligned outputs: Inference-time ordinal lens alignment for multimodal LLMs, 2026. URL `https://arxiv.org/abs/2608.20999`.

[15] 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*, 2025. doi: 10.1145/3746027.3754993. URL `https://arxiv.org/abs/2508.00726`.

[16] Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. *NeurIPS*, 2023.

[17] Xiuyu Li, Jinkai Zhang, Mingyang Yi, Yu Li, Longqiang Wang, Yue Wang, and Ju Fan. ETS: Energy-guided test-time scaling for training-free RL alignment. In *Forty-third International Conference on Machine Learning*, 2026.

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

[19] Zhuowei Li, Haizhou Shi, Yunhe Gao, Di Liu, Zhenting Wang, Yuxiao Chen, Ting Liu, Long Zhao, Hao Wang, and Dimitris N. Metaxas. The hidden life of tokens: Reducing hallucination of large vision-language models via visual information steering. In *ICML*, 2025.

[20] Yangguang Lin, Quan Fang, Yufei Li, Jiachen Sun, Junyu Gao, and Jitao Sang. HulluEdit: Single-pass evidence-consistent subspace editing for mitigating hallucinations in large vision-language models. *arXiv preprint arXiv:2602.22727*, 2026.

[21] Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. Mitigating hallucination in large multi-modal models via robust instruction tuning. In *International Conference on Learning Representations*, pages 57689–57733, 2024.

[22] Hanchao Liu, Wenyuan Xue, Yifei Chen, Dapeng Chen, Xiutian Zhao, Ke Wang, Liping Hou, Rongjun Li, and Wei Peng. A survey on hallucination in large vision-language models. *arXiv preprint arXiv:2402.00253*, 2024.

[23] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *NeurIPS*, 2023.

[24] 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 (CVPR)*, pages 26296–26306, 2024.

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

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

[27] Qwen Team. Qwen3.5: Towards native multimodal agents, 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

[28] 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 *ICML*, 2021.

[29] Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object hallucination in image captioning. In *EMNLP*, 2018.

[30] Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liangyan Gui, Yu-Xiong Wang, Yiming Yang, Kurt Keutzer, and Trevor Darrell. Aligning large multimodal models with factually augmented RLHF. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13088–13110, 2024.

[31] Kai Tang, Jinhao You, Xiuqi Ge, Hanze Li, Yichen Guo, and Xiande Huang. DCLA: Mitigating hallucinations via inter-layer consistency aggregation in large vision-language models. *arXiv preprint arXiv:2505.12343*, 2025.

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[33] Chenxi Wang, Xiang Chen, Ningyu Zhang, Bozhong Tian, Haoming Xu, Shumin Deng, and Huajun Chen. MLLM can see? dynamic correction decoding for hallucination mitigation. In *ICLR*, 2025.

[34] Junyang Wang, Yuhang Wang, Guohai Xu, Jing Zhang, Yukai Gu, Haitao Jia, Jiaqi Wang, Haiyang Xu, Ming Yan, Ji Zhang, and Jitao Sang. AMBER: An LLM-free multi-dimensional benchmark for MLLMs hallucination evaluation. *arXiv preprint arXiv:2311.07397*, 2024.

[35] Kaishen Wang, Hengrui Gu, Meijun Gao, and Kaixiong Zhou. DAMO: Decoding by accumulating activations momentum for mitigating hallucinations in vision-language models. In *International Conference on Learning Representations*, pages 56764–56780, 2025.

[36] Xintong Wang, Jingheng Pan, Liang Ding, and Chris Biemann. Mitigating hallucinations in large vision-language models with instruction contrastive decoding. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 15840–15853, 2024.

[37] Ziming Wang. TOKI: A bitemporal operator algebra for contradiction resolution in LLM-agent persistent memory. *arXiv preprint arXiv:2606.06240*, 2026.

[38] Le Yang, Ziwei Zheng, Boxu Chen, Zhengyu Zhao, Chenhao Lin, and Chao Shen. Nullu: Mitigating object hallucinations in large vision-language models via halluspace projection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14635–14645, 2025.

[39] Hao Yin, Guangzong Si, and Zilei Wang. ClearSight: Visual signal enhancement for object hallucination mitigation in multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14625–14634, 2025.

[40] Jianghao Yin, Qin Chen, Kedi Chen, Jie Zhou, Xingjiao Wu, and Liang He. Dynamic multimodal activation steering for hallucination mitigation in large vision-language models. In *ICLR*, 2026.

[41] Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng, Maosong Sun, and Tat-Seng Chua. RLHF-V: Towards trustworthy MLLMs via behavior alignment from fine-grained correctional human feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13807–13816, 2024.

[42] Jiahao Zhang, Xiuyu Li, and Suhang Wang. Your "Pro" LLM subscription may actually be "Free": Exposing fingerprint spoofing risks in LLM inference services. *arXiv preprint arXiv:2606.16100*, 2026.

[43] Linxi Zhao, Yihe Deng, Weitong Zhang, and Quanquan Gu. Mitigating object hallucination in large vision-language models via image-grounded guidance. In *ICML*, 2025.

[44] Jingmin Zhu, Yanyan Zou, Xiuyu Li, James Bailey, Dinh Phung, Tao Yu, Yunyun Hou, Jun Liu, Jun Zhou, Sirui Wang, Nan Duan, Jiachen Liu, and Qiuhong Ke. Tempo-Zero: Recursive self-play in executable worlds for video reasoning, 2026. URL `https://arxiv.org/abs/2610.32433`.

[45] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Yuchen Duan, et al. InternVL3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025. URL `https://arxiv.org/abs/2504.10479`.

[46] Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. Representation engineering: A top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*, 2023.

A Additional evaluations

This section extends the evaluation to additional backbones, cross-benchmark transfer, multi-image questions, and open-ended hallucination assessment. It also reports object recall and describes baseline evaluation for the main tables.

A.1 Transfer to Qwen2.5-VL and InternVL3

We evaluate Qwen2.5-VL [3] and InternVL3 [45] under the same MME-Object and CHAIR protocol used for DMAS. Regular, VCD, DMAS, and PAVE use matched evaluation settings within each backbone. PAVE retains the 100-image MSCOCO calibration pool, ranks $(r_V, r_E, r_H) = (8, 20, 8)$, and gate coefficients from Table 12. Only the steering layer is selected on a held-out calibration split and then frozen across test benchmarks. The offline priors H, E are estimated separately in each backbone's residual coordinates, and V is constructed from the current input's prefill states. Thus, transfer concerns the subspace-construction and editing procedure rather than reuse of activation vectors across architectures.

Table 6: **Additional backbone evaluation.** MME denotes the four-subtask MME-Object total; CHAIR values are percentages. Best values within each backbone are bold.

Backbone	Method	MME↑	$\text{CHAIR}_S \downarrow$	$\text{CHAIR}_I \downarrow$
Qwen2.5-VL	Regular	705.00	47.40	10.00
	VCD	674.67	51.00	11.50
	DMAS	718.33	43.20	9.50
	PAVE	**722.00**	**41.70**	**9.40**
InternVL3	Regular	698.33	28.40	7.60
	VCD	667.00	35.00	9.30
	DMAS	713.33	26.40	6.90
	PAVE	**716.67**	**25.80**	**6.80**

On Qwen2.5-VL, PAVE improves MME by 17.00 points over Regular and reduces CHAIR_S/CHAIR_I by $5.70/0.60$ points. On InternVL3, the corresponding changes are $+18.34$ and $-2.60/-0.80$ points. Relative to DMAS, the MME gains are 3.67 and 3.34 points, with CHAIR_S reductions of 1.50 and 0.60 points. These results extend the observed hallucination–utility trade-off beyond the two original backbones.

A.2 Cross-benchmark transfer to AMBER

AMBER [34] combines open-ended generation with discriminative questions about existence, attributes, and relations. Calibration uses only the original MSCOCO pool: no AMBER image, question, or annotation is used to construct the priors or select hyperparameters. Table 7 reports discriminative accuracy and F1, generative CHAIR and hallucination rate (Hal), and the aggregate Score. The DMAS row is a matched rerun; Regular and VCD are taken from public reports using the same metric definitions.

Table 7: **AMBER evaluation without target-benchmark calibration.** All values are percentages. † denotes a public-report baseline; ‡ denotes our matched DMAS rerun.

Method	Disc. Acc.↑	Disc. F1↑	Gen. CHAIR↓	Hal↓	Score↑
Regular†	67.40	71.20	11.60	47.70	79.80
VCD†	68.10	71.10	9.80	43.80	80.65
DMAS‡	81.90	87.20	4.90	20.90	90.01
PAVE	**82.63**	**88.00**	**4.57**	**19.56**	**90.52**

PAVE increases Score from 79.80 to 90.52, reduces generative CHAIR from 11.60 to 4.57, and reduces Hal from 47.70 to 19.56. Against the matched DMAS rerun, it improves discriminative F1 by 0.80 points and reduces generative CHAIR by 0.33 points. Together with ScienceQA and ViQuAE (Table 4), these results support transfer of MSCOCO-derived priors across the evaluated benchmarks and task formats. Medical images, document OCR, and charts remain outside this evaluation.

A.3 Multi-image and open-ended evaluation on 2026 backbones

We further evaluate Gemma 4 12B IT [9] and Qwen3.5-9B [27] on MIHBench [15] and MMHal-Bench [30]. MIHBench tests hallucination in multi-image questions; MMHal-Bench assesses hallucination in open-ended responses. Regular and PAVE use identical evaluation settings within each backbone.

Table 8: **MIHBench results.** All metrics are percentages, with "yes" as the positive class. Yes ratio is the percentage of positive predictions.

Backbone	Method	Acc.↑	Precision↑	Recall↑	F1↑	Yes ratio
Gemma 4 12B IT	Regular	64.88	60.58	85.20	70.81	70.33
	PAVE	68.25	63.06	88.10	73.51	69.85
Qwen3.5-9B	Regular	57.33	53.99	99.20	69.92	91.88
	PAVE	59.80	55.48	99.15	71.15	89.35

On MIHBench, PAVE increases accuracy and F1 by 3.37 and 2.70 points on Gemma 4, and by 2.47 and 1.23 points on Qwen3.5. Precision increases on both models while the yes ratio decreases. Qwen3.5 nevertheless retains a high yes ratio of 89.35%, and its recall changes from 99.20% to 99.15%: the intervention improves the overall comparison but leaves substantial positive-response bias.

Table 9: **MMHal-Bench results.** Higher average scores and lower hallucination rates indicate better performance.

Backbone	Method	Average score↑	Hallucination rate↓
Gemma 4 12B IT	Regular	3.47	32%
	PAVE	3.83	25%
Qwen3.5-9B	Regular	2.55	61%
	PAVE	2.75	57%

On MMHal-Bench, PAVE raises Gemma 4's average score by 0.36 and reduces its hallucination rate by 7 percentage points. For Qwen3.5, the corresponding changes are $+0.20$ and -4 points. The two benchmarks provide complementary evidence in multi-image discrimination and open-ended response generation.

A.4 Object recall and grounded caption coverage

Object recall measures coverage of ground-truth objects using the same object normalization and synonym mapping as the CHAIR evaluator. Table 10 reports the matched LLaVA-1.5 comparison. PAVE's recall decreases from 68.60% to 67.47%, a 1.13-point reduction, while CHAIR_S decreases from 51.0 to 21.6 and CHAIR_I from 15.2 to 9.7.

Table 10: **Object recall on the matched CHAIR evaluation.** Values are percentages.

Method	Regular	VTI	ICT	PAVE
Object recall↑	68.60	60.29	68.05	67.47

This recall trade-off complements the mention-level statistics in Figure 3. Average caption length changes from 50.0 to 54.7 words, object mentions from 6.39 to 8.62, and supported-mention precision from 84.8% to 90.3%. More supported mentions can coexist with lower coverage of distinct ground-truth objects. The results therefore support improved mention-level grounding with a modest loss of object recall, rather than an increase in every notion of caption coverage.

A.5 Baseline evaluation

The Qwen-VL ICT precision and recall entries in Table 1, 90.99 and 83.31, are recomputed from the same predictions used for its accuracy and F1. The Qwen-VL VTI and ICT entries in Table 4 are evaluated with a unified Qwen-VL adapter under the same checkpoint, prompt, dataset split, and evaluator: VTI obtains $45.12/49.00$ on ScienceQA/ViQuAE, and ICT obtains $46.61/49.18$. NR cells denote unavailable comparable results.

A.6 Hallucination categories and scope

PAVE's calibration pairs primarily contrast supported and unsupported object content, making object hallucination the main target. The evaluations cover several related error types:

Table 11: **Hallucination categories covered by the evaluations.**

Category	Error	Evaluation evidence
Object existence	Mentioning an absent object	CHAIR, POPE, MME existence, AMBER, MIHBench
Attributes	Assigning an unsupported property	MME color; aggregate AMBER results
Relations	Misstating a spatial or other relation	MME position; aggregate AMBER results
Counting	Giving an incorrect object count	MME count
Broader factuality	Unsupported details in open-ended answers	MMHal-Bench

Table 3 shows improvements over Regular in count, position, and color on both original backbones: $124.67 \rightarrow 160.43$, $114.00 \rightarrow 135.00$, and $151.00 \rightarrow 172.00$ on LLaVA-1.5; $127.67 \rightarrow 153.00$, $131.67 \rightarrow 133.33$, and $173.00 \rightarrow 183.00$ on Qwen-VL. AMBER and MMHal-Bench broaden the evaluation, while their aggregate scores do not isolate a gain for every individual hallucination category. The supported claim is an improved measured hallucination–utility trade-off across the reported models and tasks.

B Implementation details

B.1 Inference algorithm

Algorithm 1 summarises one decoding step of PAVE using the symbols of Section 3. The first four numbered lines run once at prefill and are cached; the remaining numbered lines run at every decoding step.

Algorithm 1 PAVE: prefill-conditioned activation editing at decoding step t.

Inputs: visual-token states $\{\boldsymbol{z}_j\}_{j=1}^{m}$ at layer ℓ_s, prefill query $\boldsymbol{q}$, hidden state $\boldsymbol{h}_t$
offline priors $E \in \mathbb{R}^{d \times r_E}$, $H \in \mathbb{R}^{d \times r_H}$
prefill rank r_V; gate gains $(\alpha_0, \beta_0, \alpha_{\max}, \beta_{\max})$
Output: edited hidden state $\boldsymbol{h}'_t$

Phase 1: Prefill (run once, cached for all t)

$w_j \leftarrow \mathrm{softmax}_j\big(\cos(\boldsymbol{z}_j,\, \boldsymbol{q})\big)$ ▷ query-conditioned token weights

$V \leftarrow \mathrm{TopEig}_{r_V}\Big(\sum_{j=1}^{m} w_j\, \boldsymbol{z}_j \boldsymbol{z}_j^{\top}\Big)$ ▷ per-image evidence basis, Eq. 5

$\widetilde{E} \leftarrow \mathrm{orth}\big([\,V,\ (\boldsymbol{I} - VV^{\top})E\,]\big)$ ▷ anchor evidence on current image, Eq. 6

$\widetilde{H} \leftarrow \mathrm{orth}\big((\boldsymbol{I} - \widetilde{E}\widetilde{E}^{\top})H\big)$ ▷ residualize drift, Eq. 7

Phase 2: Decode step t (run every step on cached $\widetilde{E}, \widetilde{H}$)

$\rho_E \leftarrow \|\widetilde{E}\widetilde{E}^{\top}\boldsymbol{h}_t\|^2/\|\boldsymbol{h}_t\|^2,\ \ \rho_H \leftarrow \|\widetilde{H}\widetilde{H}^{\top}\boldsymbol{h}_t\|^2/\|\boldsymbol{h}_t\|^2$ ▷ energy ratios

$\alpha \leftarrow \min\big(\alpha_{\max},\, \alpha_0\,(1-\rho_E)/\rho_E\big)$ ▷ evidence gate

$\beta \leftarrow \min\big(\beta_{\max},\, \beta_0\,\rho_H/(1-\rho_H)\big)$ ▷ drift gate

$\boldsymbol{h}'_t \leftarrow \boldsymbol{h}_t + \alpha\,\widetilde{E}\widetilde{E}^{\top}\boldsymbol{h}_t - \beta\,\widetilde{H}\widetilde{H}^{\top}\boldsymbol{h}_t$ ▷ step-wise edit, Eq. 10

return $\boldsymbol{h}'_t$

B.2 Evaluation metrics

CHAIR [29]. The Caption Hallucination Assessment with Image Relevance (CHAIR) metric evaluates object-level hallucination in image captions by comparing each generated caption against the COCO ground-truth object set. CHAIR comes in two variants:

$$\mathrm{CHAIR}_I \;=\; \frac{|\{\text{hallucinated mentions}\}|}{|\{\text{all object mentions}\}|}, \qquad \mathrm{CHAIR}_S \;=\; \frac{|\{\text{captions with} \geq 1 \text{ hallucination}\}|}{|\{\text{all captions}\}|}, \tag{11}$$

i.e. CHAIR_I is the per-mention hallucination rate and CHAIR_S is the per-caption hallucination rate. We follow the same parser, MSCOCO ground-truth list, and CHAIR-500 image pool used by DMAS [40] and VTI [25].

POPE [18]. The Polling-based Object Probing Evaluation (POPE) frames hallucination as Yes/No questions about object existence and reports

$$\mathrm{P} \;=\; \frac{TP}{TP+FP}, \qquad \mathrm{R} \;=\; \frac{TP}{TP+FN}, \qquad \mathrm{F1} \;=\; \frac{2\,\mathrm{P}\,\mathrm{R}}{\mathrm{P}+\mathrm{R}}, \tag{12}$$

where the True/False positives and negatives are counted across the answer Yes/No labels. POPE samples three negative-question pools, *Random*, *Popular* (the most frequent COCO categories), and *Adversarial* (categories that co-occur most often with each image's true objects), and our headline POPE F1 is the average of the three per-split F1 values.

MME [8]. For the object-oriented portion of the MME benchmark, every image is asked two yes/no questions (one with answer Yes, one with answer No) over four subtasks $\mathcal{T} = \{\text{existence, count, position, color}\}$. Each subtask scores

$$\mathrm{score}_t \;=\; \mathrm{acc}_t + \mathrm{acc}{+}_t, \tag{13}$$

where acc_t is the per-question accuracy ($\in [0, 100]$) and $\mathrm{acc}{+}_t$ is the per-image accuracy that requires both questions about the same image to be answered correctly. The MME-Object total reported in Table 3 is $\sum_{t \in \mathcal{T}} \mathrm{score}_t$ ($\in [0, 800]$).

ScienceQA [26]. ScienceQA is a multiple-choice subject-knowledge benchmark; we report the standard top-1 accuracy on its multimodal subset (questions paired with an image).

ViQuAE [13]. ViQuAE is an open-ended knowledge-extraction benchmark whose answers are free-form spans. We report the SQuAD-style normalised exact-match accuracy on the development split, matching the protocol used by DMAS [40] and VTI [25].

B.3 Implementation details

Backbones. For LLaVA-1.5-7B [24] we use the public `liuhaotian/llava-v1.5-7b` weights with the CLIP-ViT-L/14 vision tower; for Qwen-VL [2] we use the public `Qwen/Qwen-VL-Chat` weights with its ViT-bigG vision tower. Both backbones are kept frozen end-to-end; PAVE only edits the residual stream at one steering layer through the inference hook described in Algorithm 1.

Decoding protocol. The five main-text benchmarks use greedy decoding (temperature $T=0$, top-$p=1$), matching the protocol of DMAS [40] and VTI [25]. POPE follows the official MSCOCO yes/no question template. MME uses the official prompt set with the standard $\mathrm{acc}+\mathrm{acc}+$ aggregation. ScienceQA and ViQuAE follow the official prompts in their releases.

Calibration data. Both offline priors H and E are estimated from a 100-image MSCOCO calibration set disjoint from the CHAIR-500, POPE-MSCOCO, and MME object-oriented evaluation images. For each calibration image I_i we collect a faithful caption c_i^{GT} from the COCO ground-truth annotations and a hallucinated counterpart c_i^{HD} following the contrastive-pair construction used by Nullu [38] and VISTA [19]: we inject one unsupported object from the COCO-80 vocabulary into c_i^{GT}, length-match the result to the original caption, and manually verify on the full 100 pairs that the two captions differ only at the object slot rather than in style, syntax, or grammar. The hallucination contrast is $D_H[i,:] = \boldsymbol{h}_{\mathrm{last}}^{(\ell_s)}(I_i, c_i^{\mathrm{HD}}) - \boldsymbol{h}_{\mathrm{last}}^{(\ell_s)}(I_i, c_i^{\mathrm{GT}})$ and the evidence contrast is $D_E[i,:] = \boldsymbol{h}_{\mathrm{last}}^{(\ell_s)}(I_i, c_i^{\mathrm{GT}}) - \boldsymbol{h}_{\mathrm{last}}^{(\ell_s)}(I_\varnothing, c_i^{\mathrm{GT}})$, where $I_\varnothing$ is a 336×336 uniform-grey blank. Both contrasts are read at the steering layer so the resulting subspaces share a representation.

Headline configuration. Table 12 lists every PAVE hyperparameter used in the main-text headlines. The two backbones share all knobs except the steering layer.

Table 12: PAVE headline hyperparameters. Identical across both backbones except for the steering layer ℓ_s.

Group	Symbol	Description	LLaVA-1.5-7B	Qwen-VL
Geometry	ℓ_s	steering layer (residual-stream)	26	28
	r_V	per-image evidence rank (prefill SVD)	8	8
	r_E	offline evidence rank	20	20
	r_H	offline hallucination rank	8	8
Gates	α_0	evidence gain (linear regime)	0.5	0.5
	β_0	drift gain (linear regime)	0.22	0.22
	$\alpha_{\max}$	evidence cap (saturation)	3.2	3.2
	$\beta_{\max}$	drift cap (saturation)	3.8	3.8
Numerics	ϵ	energy-ratio clamp $[\epsilon, 1-\epsilon]$	10^{-6}	10^{-6}
	dtype	precision used at inference	fp16	bf16
Decoding	T	sampling temperature	0	0
	p	top-p	1.0	1.0
	–	max new tokens (main-text benchmarks)	64	64
Calibration	N	calibration pool size (MSCOCO captions)	100	100
	–	blank evidence input $I_\varnothing$	336×336 grey	336×336 grey

Compute budget. The LLaVA-1.5 and Qwen-VL experiments run on a single L40S GPU (46GB). One CHAIR evaluation pass on 500 images takes about 17 minutes; one POPE pass on 3000 questions takes about 8 minutes; one MME pass on the object subset takes about 2 minutes. Total compute for the main experiments and ablations is approximately 60 GPU-hours.

B.4 Detailed experimental results

Table 13 reports POPE accuracy, precision, recall, and F1 on the three MSCOCO sampling splits (Random, Popular, Adversarial). PAVE leads on F1 on every split: it edges past DMAS by 0.14

points on Random while keeping a high recall, and the gap widens to $+0.49$ on Popular and $+2.57$ on Adversarial. The biggest absolute gain over DMAS is on Adversarial accuracy ($+3.21$).

Table 13: POPE per-split results on LLaVA-1.5-7B (Random / Popular / Adversarial). Each cell reports the standard POPE accuracy, precision, recall, and F1 measured on the corresponding MSCOCO split.

Dataset Setting	Method	Accuracy↑	Precision	Recall	F1↑
Random	Regular	83.29	92.13	72.80	81.33
	VCD	87.73	91.42	83.28	87.16
	DMAS	90.03	90.51	90.03	90.02
	PAVE	89.85	88.96	91.41	90.16
Popular	Regular	81.88	88.93	72.80	80.06
	VCD	85.38	86.92	83.28	85.06
	DMAS	87.33	89.16	85.00	87.03
	PAVE	87.58	86.90	88.15	87.52
Adversarial	Regular	78.96	83.06	72.75	77.57
	VCD	80.88	79.45	83.29	81.33
	DMAS	83.07	82.04	84.67	83.33
	PAVE	86.28	84.96	86.87	85.90

B.5 Ablation study: rank and sensitivity

Subspace ranks and steering layer. The headline configuration uses the same ranks across both backbones: per-image evidence rank $r_V = 8$, offline evidence rank $r_E = 20$, offline hallucination rank $r_H = 8$. The only backbone-specific knob is the steering layer ($\ell_s = 26$ for LLaVA-1.5-7B, $\ell_s = 28$ for Qwen-VL). The three ranks are intentionally not tied: V summarizes a single image and stays compact; E aggregates the calibration pool and benefits from a wider span; H captures a small number of recurring drift directions. Figure 8 confirms this choice: the leading 20 singular vectors of D_E already explain 87% of the variance, while the leading 8 singular vectors of D_H cover 44% (the D_H tail is longer because unsupported-object injections vary across object categories and caption contexts).

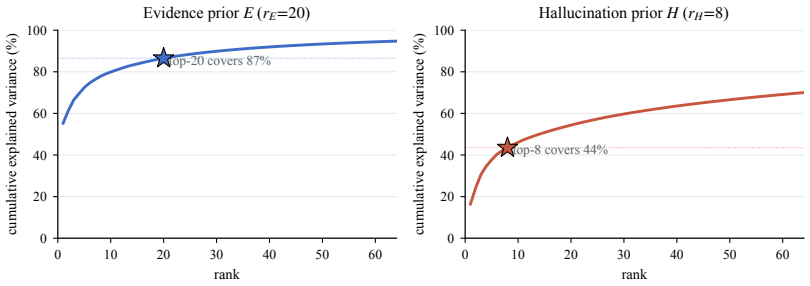


Figure 8: **Cumulative explained variance of the offline contrast matrices.** Singular spectra of D_E (left) and D_H (right) on the rank-32 rebuild of the dual bank. The leading 20 (D_E) and 8 (D_H) directions capture the dominant recurrent component, supporting the headline rank choice.

LLaVA-1.5: r_V sensitivity. Figure 9 sweeps r_V on the 100-image subset around the rank-8 default. The CHAIR$_S$ surface is shallow and basin-shaped: the trough sits at $r_V = 10$ (CHAIR$_S = 20.57$) and very large $r_V = 12$ already starts to absorb image-irrelevant directions, raising CHAIR$_S$ slightly back to 21.18. We keep $r_V = 8$ as the default to maintain a single rank setting across both backbones, demonstrating that PAVE's hyperparameters generalise without per-backbone re-tuning rather than chasing the marginal CHAIR$_S$ optimum on LLaVA-1.5.

Qwen-VL: gate-pair heatmap. Figure 10 reports the joint (α_0, β_0) Qwen-VL heatmap. The default $\star$ at $(0.5, 0.22)$ corresponds to the operating point used in Section 4.2. The default lies inside a broad shallow basin; the cell-to-cell variation across the local low-CHAIR$_S$ region stays within a few tenths of a point, and the surface only climbs once (α_0, β_0) moves several steps off the default.

Qwen-VL: per-axis sensitivity. Figure 11 mirrors the LLaVA main-text sensitivity figure on Qwen-VL across four additional axes, plotting both CHAIR$_S$ (red, lower-better) and MME (blue, higher-better) against each knob. The chosen operating point (CHAIR$_S = 25.2$, MME$= 654.33$) sits at the joint Pareto-friendly cell on every axis. On ℓ_s and r_V a one-step move can squeeze CHAIR$_S$ a

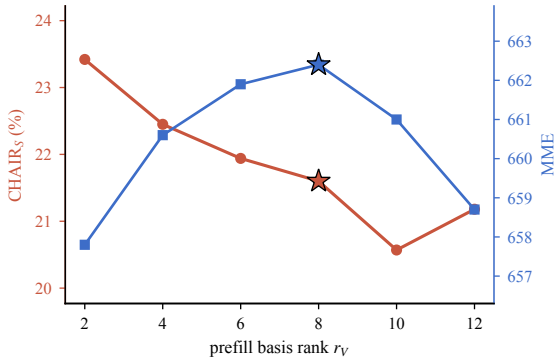


Figure 9: **LLaVA-1.5 sensitivity to the per-image evidence rank r_V.** CHAIR$_S$ (red) and MME (blue) on the 100-image subset, anchored at the default ($\star$, $r_V = 8$). The trough sits at $r_V = 10$ (CHAIR$_S$ $=$ 20.57); at $r_V = 12$ the curve rises slightly as larger ranks pull in image-irrelevant directions. The rank-8 default is shared with Qwen-VL to keep a single setting across backbones.

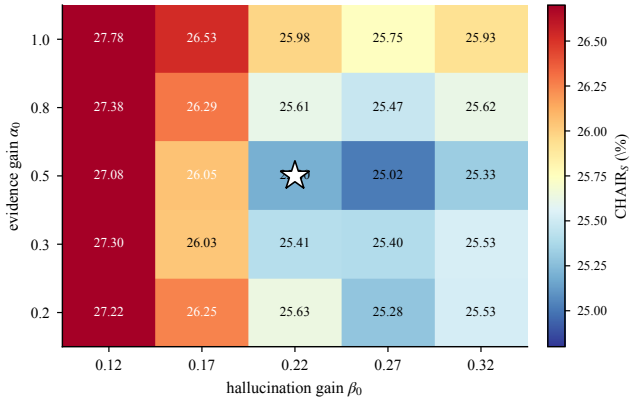


Figure 10: **Qwen-VL hyperparameter sensitivity, gate-pair view.** CHAIR$_S$ on the $\alpha_0 \times \beta_0$ grid; default $\star$ at $(0.5, 0.22)$. The basin is shallow and centred on the default, mirroring the LLaVA gate-pair landscape.

fraction lower but always at the cost of a measurable MME drop. On r_E and r_H the default is itself the CHAIR$_S$ minimum: shifting to $r_E = 24$ or $r_H = 10$ raises CHAIR$_S$ to 25.4 and 25.3 respectively, and other settings are worse still.

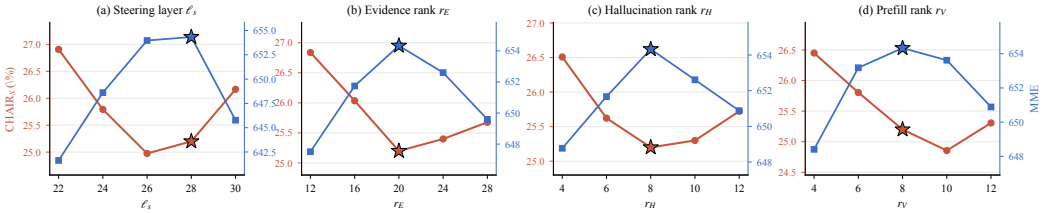


Figure 11: **Qwen-VL hyperparameter sensitivity, per-axis view (CHAIR$_S$ + MME).** (a) Steering layer ℓ_s, (b) evidence rank r_E, (c) hallucination rank r_H, (d) prefill rank r_V. Red curves: CHAIR$_S \downarrow$ on the left axis; blue curves: MME$\uparrow$ on the right. Default $\star$ is the joint Pareto-friendly cell across both metrics on every axis.

C Analysis

C.1 Local effect of the evidence-protected update

The two calibration contrasts isolate different changes: D_H holds the image fixed while changing unsupported object content, whereas D_E holds the faithful caption fixed while removing visual input. Their leading singular directions provide separate low-rank priors for drift suppression and evidence

amplification. Prefill adaptation anchors the evidence span on the current image, and residualization removes this protected span from the suppression basis.

Let $P_E = \widetilde{E}\widetilde{E}^\top$ and $P_H = \widetilde{H}\widetilde{H}^\top$. Equation 7 gives $P_E P_H = P_H P_E = 0$. Applying these projectors to Equation 10 yields

$$P_E \boldsymbol{h}'_t = (1+\alpha_t) P_E \boldsymbol{h}_t, \qquad P_H \boldsymbol{h}'_t = (1-\beta_t) P_H \boldsymbol{h}_t, \tag{14}$$

$$(\boldsymbol{I} - P_E - P_H)\boldsymbol{h}'_t = (\boldsymbol{I} - P_E - P_H)\boldsymbol{h}_t, \tag{15}$$

where $\alpha_t = \alpha(\boldsymbol{h}_t)$ and $\beta_t = \beta(\boldsymbol{h}_t)$. Thus the evidence-component norm scales by $1+\alpha_t$, the drift-component norm scales by $|1-\beta_t|$, and the orthogonal remainder is unchanged. Evidence is amplified for $\alpha_t > 0$, and the drift component contracts for $0 < \beta_t < 2$. These identities hold at the steering layer; subsequent nonlinear decoder layers can mix the components.

The energy gates allocate the update according to the current state's projections. The evidence gate decreases as ρ_E grows, while the drift gate increases with ρ_H. Orthogonality also gives the edit norm exactly:

$$\frac{\|\boldsymbol{h}'_t - \boldsymbol{h}_t\|^2}{\|\boldsymbol{h}_t\|^2} = \alpha_t^2 \rho_E(\boldsymbol{h}_t) + \beta_t^2 \rho_H(\boldsymbol{h}_t) \le \max\{\alpha_{\max}^2, \beta_{\max}^2\}. \tag{16}$$

The caps bound the intervention magnitude. The default $\beta_{\max} = 3.8$ permits values outside the contraction interval, so the exact scaling above, together with the component ablations in Table 5, describes the local mechanism and its measured effects.

C.2 Per-step gate dynamics

Gating form. The step-wise gates take the rational form of Equation 9 in the main text,

$$\alpha(\boldsymbol{h}_t) = \min\left(\alpha_{\max},\ \alpha_0 \frac{1-\rho_E(\boldsymbol{h}_t)}{\rho_E(\boldsymbol{h}_t)}\right), \qquad \beta(\boldsymbol{h}_t) = \min\left(\beta_{\max},\ \beta_0 \frac{\rho_H(\boldsymbol{h}_t)}{1-\rho_H(\boldsymbol{h}_t)}\right),$$

where ρ_E, ρ_H are the energy ratios in the adapted evidence and hallucination subspaces, α_0, β_0 are linear-regime gains, and $\alpha_{\max}, \beta_{\max}$ are caps that bound the per-step edit norm.

Default values. The headline gains are $\alpha_0 = 0.5$ and $\beta_0 = 0.22$, with caps $\alpha_{\max} = 3.2$ and $\beta_{\max} = 3.8$. The two gains are picked once on the 100-image MSCOCO calibration split, disjoint from CHAIR-500, POPE-MSCOCO, and MME object images, and then frozen across both backbones and all benchmarks. In implementation, ρ_E and ρ_H are clamped to $[\epsilon, 1-\epsilon]$ with $\epsilon = 10^{-6}$ before substituting into Eq. 9.

Stability. Figure 12 shows that on a generated caption both gates fire in the neighbourhood of object-token positions where the residual stream carries the object signal; the proxy margin $\rho_E - \rho_H$ moves in the expected direction after the edit.

C.3 Geometry verification

The Method (Section 3) constructs the residualized hallucination subspace $\widetilde{H}$ inside the orthogonal complement of the adapted evidence subspace. As an independent verification we measure the principal angles between the offline prior H and the per-image evidence basis V_x before and after this residualization. Figure 6a shows the distribution: without residualization the smallest principal angle sits near 77° on average, leaving a non-negligible alignment that a fixed suppression would exploit; after residualization every principal angle is within 0.5° of 90°. The Frobenius cross-product norm $\|V_x^\top H\|_F^2 / r_V$ falls from a mean of $9{\times}10^{-3}$ to $2.4{\times}10^{-15}$ across 120 MSCOCO inputs, twelve orders of magnitude. At the steering layer, the linear suppression term therefore has negligible projection onto the protected evidence subspace; downstream decoder layers may still remix activations, so this guarantee is local to ℓ_s.

C.4 Complexity and runtime breakdown

PAVE adds a tiny amount of constant-rank algebra on top of the regular decoder pass. We summarise per-token compute and memory in Table 14. All numbers are measured on a single L40S GPU at the headline LLaVA-1.5-7B setting ($\ell_s = 26$, $r_V = 8$, $r_E = 20$, $r_H = 8$).

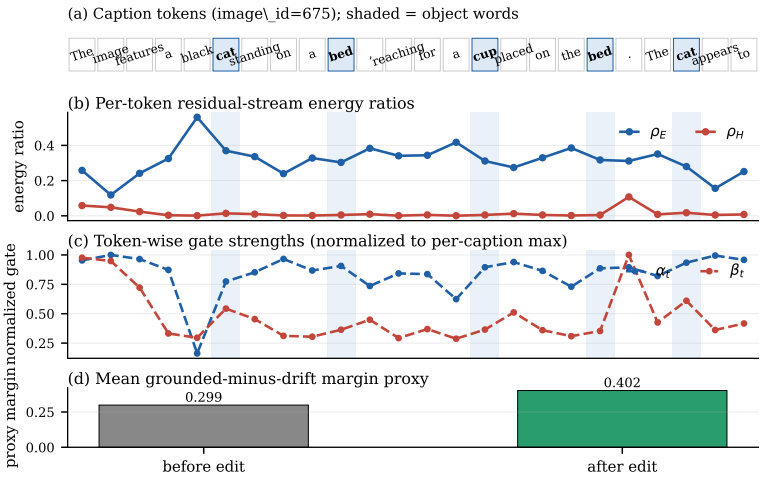


Figure 12: **Per-step gate dynamics on a single caption.** (a) Caption tokens; shaded boxes mark COCO-object words. (b) Per-step residual-stream energy ratios ρ_E and ρ_H at the steering layer ℓ_s. (c) Step-wise gates α_t and β_t, normalised to the per-caption maximum to expose the firing pattern. (d) Mean grounded-minus-drift margin proxy $\rho_E - \rho_H$ before and after the edit.

Per-token compute. The added cost at every decoding step is two projection matmuls of size $(d \times r)$ for ρ_E, ρ_H readout, two scalar gate evaluations, and one rank-$(r_V{+}r_E)$ plus one rank-r_H projector update of $\boldsymbol{h}_t \in \mathbb{R}^d$:

$$\text{FLOPs}_{\text{step}}^{\text{PAVE}} \;=\; 4\,d\,(\widetilde{r}_E + \widetilde{r}_H) \;+\; \mathcal{O}(\widetilde{r}_E + \widetilde{r}_H), \qquad \widetilde{r}_E = r_V + r_E,\; \widetilde{r}_H = r_H. \tag{17}$$

With $d{=}4096$ on LLaVA-1.5-7B and $\widetilde{r}_E{=}28, \widetilde{r}_H{=}8$, this works out to roughly $5.9{\times}10^5$ FLOPs per step. The vanilla decoder forward pass uses $\sim 1.4{\times}10^{10}$ FLOPs per step, so the PAVE overhead is below $5{\times}10^{-5}$ of the decoder's compute.

Table 14: Per-step compute and memory of PAVE on top of the regular LLaVA-1.5-7B decoder. The edit operates on tensors of width $d{=}4096$ at one steering layer.

Quantity	Object	Shape / count	Footprint
Cached after prefill	V	4096×8	128 KB (fp16)
	$\widetilde{E}$	4096×28	448 KB (fp16)
	$\widetilde{H}$	4096×8	128 KB (fp16)
	$\widetilde{E}\widetilde{E}^\top$, $\widetilde{H}\widetilde{H}^\top$ projectors	not materialised	0 (used as $UU^\top \boldsymbol{h}_t$)
Per-step compute	energy ratios ρ_E, ρ_H	$2 \times d \times (\widetilde{r}_E{+}\widetilde{r}_H)$ matmul	$\sim 2.95{\times}10^5$ FLOPs
	gates α, β	2 scalar evals	$\mathcal{O}(1)$
	edit $\boldsymbol{h}_t \to \boldsymbol{h}'_t$	$2 \times d \times (\widetilde{r}_E{+}\widetilde{r}_H)$ matmul	$\sim 2.95{\times}10^5$ FLOPs
Per-step total	PAVE additive overhead	–	$\sim 5.9{\times}10^5$ FLOPs
	Decoder forward (one step)	one transformer block stack	$\sim 1.4{\times}10^{10}$ FLOPs
	Relative overhead		$< 5{\times}10^{-5}$

Memory footprint. The cached bases $V, \widetilde{E}, \widetilde{H}$ together take ~ 700 KB in fp16 per image, three orders of magnitude below the LLaVA-1.5-7B activation footprint at one decoding step (~ 340 MB at $d{=}4096$ and 64-token context). This is consistent with the 13.71 GB peak GPU allocation reported in Table 5, only 0.19 GB above the Vanilla peak (13.52 GB).

C.5 Calibration-pool size sensitivity

The headline configuration uses $N = 100$ MSCOCO calibration pairs to estimate E and H. Figure 7 sweeps $N \in \{50, 100, 250, 500, 1000\}$ and re-evaluates CHAIR-500 from scratch for each N. CHAIR_S saturates around $N{=}100$: doubling to $N{=}250$ shaves only 0.3 points, and the curve is essentially flat past $N{=}500$. The reason is structural: the leading singular vectors of D_E and D_H stabilise after roughly 80–100 contrastive pairs because all useful directions are already covered by the first few principal components (Figure 8). Scaling further trades runtime for noise.

C.6 Calibration-construction ablation

Table 15 compares the four constructions of the evidence contrast $D_E[i,:]$ that we considered while building the calibration pool. The headline uses a uniform-grey blank as the visual condition that nullifies image evidence. Replacing that blank with a per-image Gaussian-noise image shifts CHAIR_S by less than half a point in either direction, while replacing the contrast entirely with a shuffled-caption baseline (I_i vs. a different caption c_j^{GT}) erases the gain because the resulting D_E no longer isolates evidence shifts. We use the blank construction for clarity of interpretation, not because it is uniquely optimal.

Table 15: Effect of how we build the evidence contrast $D_E[i,:]$. All other knobs are held at the headline (LLaVA-1.5-7B). The default ($I_\varnothing$ = uniform-grey blank) appears in the first row.

Construction of $D_E[i,:]$	$\text{CHAIR}_S \downarrow$	MME↑	Δ vs. default	Comment
$\boldsymbol{h}_{\text{last}}(I_i, c_i^{\text{GT}}) - \boldsymbol{h}_{\text{last}}(I_\varnothing, c_i^{\text{GT}})$ [default]	21.6	662.4	–	uniform-grey blank $I_\varnothing$
$\boldsymbol{h}_{\text{last}}(I_i, c_i^{\text{GT}}) - \boldsymbol{h}_{\text{last}}(I_{\text{noise}_i}, c_i^{\text{GT}})$	21.9	660.1	$+0.3/-2.3$	Gaussian noise per image
$\boldsymbol{h}_{\text{last}}(I_i, c_i^{\text{GT}}) - \boldsymbol{h}_{\text{last}}(I_{\text{mean}}, c_i^{\text{GT}})$	21.4	659.8	$-0.2/-2.6$	MSCOCO mean image as $I_\varnothing$
$\boldsymbol{h}_{\text{last}}(I_i, c_i^{\text{GT}}) - \boldsymbol{h}_{\text{last}}(I_i, c_j^{\text{GT}})\ (i \neq j)$	33.2	611.5	$+11.6/-50.9$	shuffled-caption baseline

C.7 Decoding-step edit distribution

Where in a caption does PAVE actually fire? Figure 6b bins the 500 CHAIR captions by decoding step $t \in [1, 64]$ and plots the mean $\alpha\rho_E$ (evidence-amplification effort) and $\beta\rho_H$ (drift-suppression effort) at each step. Drift-suppression effort stays near zero for the first 10 tokens, while evidence-amplification effort rises gradually from the start; both curves climb into a long, broad band that peaks around $t \in [20, 40]$, exactly the region where the CHAIR parser flags hallucinated object words on the same captions. Past $t \approx 50$ the residual stream stabilises and the gates relax. The shape is consistent with the per-step gate dynamics in Figure 12: the edit tracks the moments at which drift actually appears, rather than firing uniformly.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract claims input-specific hidden-state intervention, state-of-the-art reduction of object hallucination on CHAIR and POPE, preserved utility on MME, and transfer to OOD VQA without recalibration. These are each supported by Tables 1/2/3/4 in Section 4. The three contributions listed at the end of Section 1 map one-to-one to Section 3 (method) and Sections 4.2–4.4 (experimental support).

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 5 contains an explicit "Limitation" paragraph noting that PAVE requires white-box access to the residual stream, which rules out closed-weight or API-only LVLMs, and that the offline priors are estimated from a small calibration pool that needs refreshing on substantial domain shifts. Appendix C.5 reports the calibration-pool-size sensitivity that quantifies the second point.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper's contribution is a training-free inference-time method, not a theoretical result; there are no theorems or propositions to prove. The geometric property used in Section 3.2 (the residualized hallucination subspace $\widetilde{H}$ is orthogonal to the adapted evidence subspace $\widetilde{E}$ at the steering layer) is verified empirically in Appendix C.3.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Section 3 specifies every editing operation (Eqs. 1–10); Algorithm 1 (Appendix B.1) gives the inference loop. Section 4.1 and Appendix B.3 list the backbones, decoding protocol, calibration construction, and the steering-layer choice per backbone; Table 12 fully enumerates the hyperparameters used at every headline experiment. The evaluation protocol matches DMAS [40] and VTI [25] for direct comparability.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The abstract states that the code will be released upon publication. The PAVE inference code, per-backbone calibration banks (E and H), and evaluation scripts will be released. All datasets are publicly available (MSCOCO, POPE [18], MME [8], ScienceQA [26], ViQuAE [13]) and cited at first use.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 4.1 reports the decoding protocol (greedy, $T = 0$, top-$p = 1$, 64-token cap), evaluation pools, and backbones. Table 12 (Appendix B.3) lists every PAVE hyperparameter (ℓ_s, r_V, r_E, r_H, α_0, β_0, $\alpha_{\max}$, $\beta_{\max}$, numerical clamp ϵ) for both backbones. Hyperparameters were tuned once on a held-out subset and frozen across all benchmarks; PAVE is training-free, so no optimizer or learning-rate schedule is involved.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Every reported number in Tables 1, 2, 3, and 4 is the mean across three independent decoding seeds $(0, 1, 42)$, as stated in Section 4.1. Per-seed variation is small (under 0.5 CHAIR_S point and under 0.05 CHAIR_I point on the same configuration), but explicit error bars or significance tests are not surfaced in the tables for compactness; raw per-seed values are available with the code release.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix B.3 ("Compute budget") reports that all experiments run on a single L40S GPU (46 GB), with one CHAIR-500 pass taking $\sim$17 min, one POPE pass $\sim$8 min, and one MME object pass $\sim$2 min; total compute for all reported main and ablation experiments is $\sim$60 GPU-hours. Appendix C.4 (Table 14) gives per-step FLOP and memory accounting.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: The work uses publicly released LVLM checkpoints (LLaVA-1.5-7B, Qwen-VL-Chat) and public image-caption datasets (MSCOCO, POPE, MME, ScienceQA, VQuAE), involves no human subjects and no personally identifying data, and aims to reduce hallucination, an alignment-positive direction. The authors have reviewed the NeurIPS Code of Ethics and find no deviations.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: Section 1 motivates the work by the reliability gap LVLM hallucinations create in clinical imaging, autonomous systems, and consumer assistants; reducing object hallucination has a clear positive impact in those settings. The hallucination prior H could in principle be inverted to amplify hallucinations (a dual-use risk); since exploitation requires the same white-box backbone access already governed by each model's license, no application-specific safeguards beyond standard model-license adherence are warranted.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper releases inference-time editing code and per-backbone calibration banks (E, H), not new high-risk pre-trained model weights or scraped data. Misuse of the calibration banks (e.g., inverting H) requires access to the same publicly released LVLMs already governed by their own licenses, so no additional safeguards beyond honoring those upstream licenses apply.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All upstream assets are cited at first use: LLaVA-1.5-7B [24] (Apache 2.0), Qwen-VL [2] (Tongyi Qianwen license), MSCOCO captions / images (CC BY 4.0), POPE [18], MME [8], ScienceQA [26], and ViQuAE [13]. Baseline numbers reproduced in Tables 1–4 follow the matched evaluation protocols of DMAS [40] and VTI [25].

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The new assets are the PAVE inference code, the offline calibration banks (E, H for each backbone), and the calibration construction scripts. Documentation will accompany the release; the headline configuration is fully specified by Table 12 and Algorithm 1, and Appendix B.3 states how the calibration pool is built (the contrastive-pair construction of Nullu / VISTA on a 100-image MSCOCO subset).

Guidelines:

- The answer [N/A] means that the paper does not release new assets.

- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper involves no crowdsourcing and no research with human subjects; all evaluations use automated benchmarks (CHAIR, POPE, MME, ScienceQA, ViQuAE) on public image-caption datasets.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper involves no human subjects research; no IRB approval is required.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LVLMs (LLaVA-1.5-7B, Qwen-VL) are the object of study, not a methodological component or assistant. No LLM was used to design the method, generate training data, or conduct experiments.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.