When Language Priors Dominate Visual Evidence: CoSMo for Hallucination Mitigation in VLMs

Anonymous ACL submission

Abstract

Vision Language Models (VLMs) achieve strong performance on multimodal tasks, but still suffer from hallucination, especially in object existence probing. Their responses are often dominated by language priors and lack stable use of visual evidence, leading to weak concept alignment during multimodal reasoning. To address this issue, we propose CoSMo, a concept- space modulation framework that improves visual grounding through concept guided feature modulation and lightweight latent modulation during decoding. Specifically, Concept Bottleneck Model (CBM) produce semantic concept activations that guide Feature-wise Linear Modulation (FiLM) on visual representations, encouraging more reliable extraction of task relevant visual semantics. During inference, latent correction directions estimated from hidden representations are further used to reduce systematic biases in visual evidence utilization.Experiments on the POPE hallucination benchmark demonstrate that CoSMo achieves state-of-the-art performance in the small-scale VLM regime and substantially reduces hallucination on precision-sensitive metrics, with precision improvements of up to 16.42% on the OneVision-0.5B model compared to baseline methods.

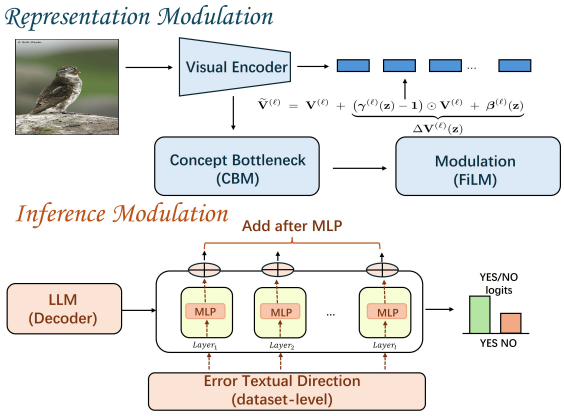

Figure 1: Illustration of the modulation mechanism in CoSMo. Visual representations are first conditioned by concept-aware Feature-wise Linear Modulation (FiLM) using concept logits predicted from the concept bottleneck module (CBM), improving concept-sensitive visual feature activation. The resulting dominant concepts are organized into separate concept-specific memory branches and fused into the decoder input. During decoding, layerwise error directions estimated from hidden-state statistics are further applied as inference-time latent modulation signals to suppress hallucination-prone activations and enhance grounded decoding.

1 Introduction

Vision Language Models (VLMs) have achieved remarkable progress in visual question answering, image captioning, and multimodal reasoning by coupling visual encoders with Large Language Models (LLMs) (Wang et al., 2024b). However, despite their strong generative capability, recent studies show that VLMs often suffer from hallucination, producing responses that are poorly grounded in visual evidence or overly influenced by language priors (Golovanevsky et al., 2025). Such hallucinations not only reduce prediction reliability, but also raise concerns about whether visual information is reliably utilized during multimodal reasoning. Recent studies suggest that hallucination is closely related to weak and unstable use of visual information in VLMs. Studies on token pruning show that removing a large number of visual tokens often causes only minor performance degradation (Cao et al., 2023). This observation suggests that informative visual signals may be hidden by redundant representations (Neo et al., 2025). Analyses of cross-modal attention further reveal that visual evidence is not always consistently propagated during autoregressive decoding, while visual tokens from deeper encoder layers often receive much lower attention weights than textual prompts (An et al., 2025). In addition,

multimodal training is commonly optimized toward language generation objectives, which leads to coarse image–text alignment and inference behaviors dominated by language priors (Jiang et al., 2026; Golovanevsky et al., 2025).

Existing approaches attempt to alleviate these issues through token pruning, multimodal alignment training, instruction tuning, or grounding-oriented optimization (Cao et al., 2023; Li et al., 2023a; Dai et al., 2023). However, most methods focus on only one aspect of efficiency, interaction, or alignment. As a result, they struggle to jointly improve representation organization and inference-time visual utilization within a unified and interpretable framework. Concept Bottleneck Model (CBM) provide a natural solution by organizing intermediate representations into human-interpretable concepts (Koh et al., 2020). Related studies further suggest that visual semantics often exhibit low-dimensional and factorized structures, enabling controllable feature organization and modulation along semantic dimensions (Rajendran et al., 2024).

Motivated by these observations, we aim to alleviate hallucination by jointly improving visual representation organization and inference-time visual utilization in VLMs. Specifically, we introduce concept-guided representation modulation to inject interpretable semantic structures into the visual interface, reducing redundancy and weak alignment. We further perform lightweight hidden-state modulation during decoding using error-conditioned statistical directions estimated from hidden representations. To maintain semantic consistency, the modulation process is constrained by geometric priors in the concept space.

To preserve the original VLM backbone as much as possible, we design an integrated and lightweight pipeline, as illustrated in Fig. 1. Conceptual semantics are first extracted through a CBM and injected into visual representations through Feature-wise Linear Modulation (FiLM) and residual modulation. Visual features are then organized along salient conceptual dimensions to construct compositional visual memories. Finally, hierarchical hidden-state directions estimated from correct and incorrect prediction subsets are injected into decoder layers during inference. This process enables lightweight distribution-level correction without modifying backbone weights.

To evaluate both fine-grained visual grounding and general-domain hallucination robustness, we conduct experiments on a newly constructed CUB-POPE benchmark and the standard COCO POPE benchmark. We further introduce concept-region visual perturbation analysis to measure prediction sensitivity when concept-related visual regions are disturbed, providing a direct evaluation of visual information utilization. Experimental results demonstrate that our method achieves state-of-the-art performance among comparable approaches, improving accuracy by up to 28% over baseline models while substantially reducing hallucination behaviors.

The main contributions of this paper are summarized as follows:

- **CoSMo: Concept-space modulation for VLMs.** We propose CoSMo, a lightweight concept-space modulation framework that improves visual grounding by organizing visual representations along semantic concept dimensions. Experiments on the POPE hallucination benchmark demonstrate state of the art performance, with precision improvements of up to 16.42% over baseline models in small scale VLM settings.

- **Concept guided feature modulation:** We introduce a CBM conditioned FiLM framework that injects interpretable concept semantics into visual representations, improving the organization and utilization of visual information.

- **Error guided latent modulation:** We estimate latent correction directions from correct and incorrect prediction subsets and apply lightweight hidden state modulation during decoding to reduce systematic biases in visual evidence utilization.

2 Related Work

Hallucination in VLMs Evaluation of VLMs has gradually shifted from correctness to faithfulness to visual evidence. Comprehensive benchmarks (Li et al., 2024b; Liu et al., 2024b) cover perception and reasoning from multiple dimensions, while POPE (Li et al., 2023d) formulates object existence as binary probes, revealing object hallucination driven by linguistic priors. Subsequent extensions such as H-POPE (Jiang et al., 2025) further characterize hierarchical and attribute-level error patterns. Mean-

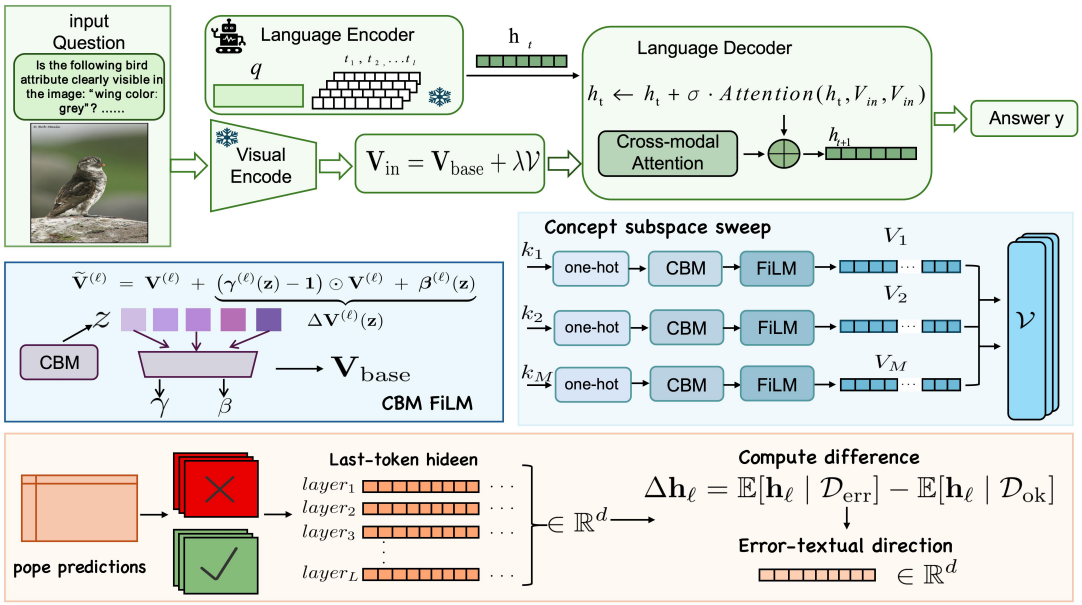

Figure 2: Overview of CoSMo. CoSMo combines concept bottleneck modulation, concept-specific visual memory fusion, and error-conditioned latent modulation for concept-aligned visual and latent representation control.

while, component-level analyses and surveys attribute hallucinations to factors such as attention routing, alignment strategies, and projection interfaces (Leng et al., 2024; Ghosh et al., 2025). Overall, hallucination is closely related to insufficient cross-modal alignment, attention reuse patterns, and shared static visual conditions, which motivates controllable modulation at the visual interface and test-time correction at the level of attention geometry.

Concept representations Concept Bottleneck Model (CBM) (Koh et al., 2020) introduce an explicit concept layer to obtain intervenable intermediate representations. Post-hoc construction and hierarchical extensions (Yuksekgonul et al., 2023) reduce annotation cost and mitigate information leakage, while auxiliary supervision and contrastive/sparse objectives are used to shape concept geometry (Vandenhirtz et al., 2024). Unlike classification settings, concepts in LVLMs must not only be interpretable but also compatible with dynamic decoding. The concept subspace perspective emphasizes the decomposability of semantic directions (Graziani et al., 2023), providing representational support for decomposing visual memory along concept axes and combining them downstream.

Feature modulation and controllable conditioning Recent studies improve LVLM reliability through feature-level modulation and controllable conditioning mechanisms. FiLM-based approaches (Perez et al., 2018; Strub et al., 2018; Wang and Zhu, 2025) demonstrate that feature-wise affine modulation effectively injects semantic priors into visual representations, while concept bottleneck models provide interpretable intermediate supervision for controllable prediction. Different from prior global conditioning strategies, CoSMo organizes modulation directly within concept-space by combining concept-guided FiLM modulation, concept-conditioned visual memories, and lightweight inference-time latent correction, enabling more reliable visual evidence utilization without modifying large backbone parameters.

3 Method

3.1 Overview

CoSMo in Fig 2 consists of three complementary components for concept-aware visual modulation and hallucination mitigation.

Given visual tokens $\mathbf{V}$ extracted by the vision encoder, we first introduce a concept bottleneck module that maps pooled visual representations to concept logits $\mathbf{z} \in \mathbb{R}^{K_{\text{all}}}$ and corresponding concept confidences $\mathbf{c} \in (0,1)^{K_{\text{all}}}$. These concept representations are used to condition FiLM-based feature modulation, producing concept-aware visual features $\widetilde{\mathbf{V}}$ while preserving compatibility with the original VLM interface.

Based on the dominant activated concepts se-

lected from $\mathbf{z}$, we further construct multiple concept-conditioned visual branches. Each branch produces a concept-specific visual memory $\mathbf{V}_k$, and the resulting memories are fused with the baseline visual representation $\mathbf{V}_{\text{base}}$ to form the final multimodal input $\mathbf{V}_{\text{in}}$ provided to the decoder.

Finally, during autoregressive decoding, we estimate layerwise latent correction directions $\{\Delta\mathbf{h}_\ell\}_{\ell=1}^{L}$ from hidden-state statistics of hallucination-prone predictions and apply lightweight directional modulation to decoder activations at inference time.

3.2 Concept bottleneck representation

We introduce a concept bottleneck that maps pooled visual representations into concept logits and corresponding concept confidences:

$$\begin{aligned} \mathbf{c} &= \sigma(\mathbf{z}) \in (0,1)^{K_{\text{all}}}, \\ \mathbf{z} &= g_{\text{cbm}}(\text{POOL}(\mathbf{V})) \in \mathbb{R}^{K_{\text{all}}}. \end{aligned} \tag{1}$$

Here, $\mathbf{V}$ denotes visual tokens, g_{cbm} is a lightweight MLP, K_{all} denotes the concept vocabulary size, and L denotes the number of vision encoder layers. $\text{POOL}(\cdot)$ takes the CLS token of $\mathbf{V}$ at the early anchor depth $\ell_{\text{early}} = \lfloor L/4 \rfloor$.

3.3 Concept-conditioned latent modulation

We formulate concept conditioning as a residual update around the identity mapping (Rebuffi et al., 2017):

$$\widetilde{\mathbf{V}}^{(\ell)} = \mathbf{V}^{(\ell)} + \underbrace{\left(\boldsymbol{\gamma}^{(\ell)}(\mathbf{z}) - \mathbf{1}\right) \odot \mathbf{V}^{(\ell)} + \boldsymbol{\beta}^{(\ell)}(\mathbf{z})}_{\Delta\mathbf{V}^{(\ell)}(\mathbf{z})}. \tag{2}$$

Here, $\boldsymbol{\gamma}^{(\ell)}, \boldsymbol{\beta}^{(\ell)} \in \mathbb{R}^d$ are broadcast across visual tokens. The residual term $\Delta\mathbf{V}^{(\ell)}(\mathbf{z})$ represents the concept-induced offset added to the original representation.

Eq. (2) is algebraically equivalent to the standard FiLM form $\boldsymbol{\gamma}^{(\ell)}(\mathbf{z}) \odot \mathbf{V}^{(\ell)} + \boldsymbol{\beta}^{(\ell)}(\mathbf{z})$.

The additive-only variant corresponds to $\boldsymbol{\gamma}^{(\ell)}(\mathbf{z}) = \mathbf{1}$:

$$\widetilde{\mathbf{V}}^{(\ell)} = \mathbf{V}^{(\ell)} + \boldsymbol{\beta}^{(\ell)}(\mathbf{z}), \qquad \Delta\mathbf{V}^{(\ell)}(\mathbf{z}) = \boldsymbol{\beta}^{(\ell)}(\mathbf{z}). \tag{3}$$

Finally, $\widetilde{\mathbf{V}}^{(\ell)}$ is projected and packed into the multimodal token sequence through the standard VLM interface.

Depth schedule. We use three fixed depth anchors: $\ell_{\text{early}} = \lfloor L/4 \rfloor$, $\ell_{\text{mid}} = \lfloor L/2 \rfloor$, and $\ell_{\text{late}} = L - 1$.

During CBM training, Eq. (2) is applied at all three anchors. When attached to a frozen VLM, modulation is applied at a single depth specified by the VLM configuration (Yao et al., 2023), after which the modulated tokens follow the standard projection and packing pipeline.

3.4 Concept subspace allocation and fusion

To reduce representation interference under a shared decoder, we allocate separate concept-conditioned branches for dominant concepts.

Given the concept logits $\mathbf{z}$, we select the Top-M activated concepts:

$$\mathcal{K} = \{k_1, \ldots, k_M\}, \qquad M \ll K_{\text{all}}. \tag{4}$$

For each selected concept $k \in \mathcal{K}$, we construct an axis-aligned concept vector:

$$\mathbf{z}_k = s\mathbf{e}_k, \tag{5}$$

where $\mathbf{e}_k$ denotes the one-hot basis vector associated with concept axis k and $s \gg 0$ is a fixed activation strength on the selected axis.

Each $\mathbf{z}_k$ is independently passed through

$$\text{FiLM} \rightarrow \text{pack},$$

yielding a concept-specific visual memory $\mathbf{V}_k$ (without re-applying g_{cbm}). The resulting memories are stacked as

$$\mathcal{V} = \{\mathbf{V}_k\}_{k\in\mathcal{K}} \in \mathbb{R}^{M\times T_{\text{v}}\times D}. \tag{6}$$

We then fuse the baseline memory and concept-specific memories using normalized concept weights:

$$\mathbf{V}_{\text{in}} = \mathbf{V}_{\text{base}} + \lambda \sum_{k\in\mathcal{K}} \hat{w}_k \mathbf{V}_k, \tag{7}$$

where

$$\hat{w}_k = \frac{\exp(|z_k|)}{\sum_{j\in\mathcal{K}} \exp(|z_j|)}. \tag{8}$$

Here λ controls the contribution of concept-conditioned branches.

3.5 Error conditioned latent vector modulation

At inference time, we estimate latent correction directions from hidden state statistics and apply lightweight directional modulation during decoding (Li et al., 2023b).

Offline direction estimation. Under a fixed task setup, we partition an evaluation split into an error set $\mathcal{D}_{\text{err}}$ and a correct set $\mathcal{D}_{\text{ok}}$ according to model predictions.

For each sample, we extract the hidden state of the final token at every Transformer layer (Alain and Bengio, 2017). The layerwise error direction is computed as

$$\Delta \mathbf{h}_\ell = \mathbb{E}[\mathbf{h}_\ell \mid \mathcal{D}_{\text{err}}] - \mathbb{E}[\mathbf{h}_\ell \mid \mathcal{D}_{\text{ok}}], \qquad \ell = 1, \ldots, L. \tag{9}$$

This direction characterizes the average latent shift associated with hallucination-prone predictions (Kadavath et al., 2022).

Inference time hidden state modulation. During decoding, let $\mathbf{h}_\ell \in \mathbb{R}^D$ denote the post-MLP hidden state at layer ℓ. We first normalize the hidden state:

$$\widehat{\mathbf{h}}_\ell = \frac{\mathbf{h}_\ell}{\|\mathbf{h}_\ell\|_2}. \tag{10}$$

The modulated hidden state is then computed as

$$\mathbf{h}'_\ell = \|\mathbf{h}_\ell\|_2 \cdot \text{normalize}\left(\widehat{\mathbf{h}}_\ell - \eta \alpha_1 \widehat{\mathbf{d}}_\ell\right), \tag{11}$$

where

$$\widehat{\mathbf{d}}_\ell = \frac{\Delta \mathbf{h}_\ell}{\|\Delta \mathbf{h}_\ell\|_2}. \tag{12}$$

Here α_1 controls the modulation strength and η is a geometric mixing coefficient. By subtracting the normalized error direction, the decoder hidden state is shifted away from hallucination-prone regions in latent space.

This operation applies a lightweight directional correction to decoder activations without modifying model parameters (Huang et al., 2024; Li et al., 2023c).

Concept subspace constraint. To constrain modulation directions to semantically meaningful regions, we project $\Delta \mathbf{h}_\ell$ onto the subspace spanned by concept embeddings:

$$\widetilde{\Delta \mathbf{h}}_\ell = \Pi_{\mathcal{S}}(\Delta \mathbf{h}_\ell). \tag{13}$$

Here $\mathcal{S}$ denotes the low-dimensional semantic subspace induced by concept word embeddings. This projection suppresses off-subspace perturbations while preserving concept-aligned directions.

4 Experiments

4.1 Experimental Settings

Models. We evaluate our method on several representative LVLMs, including LLaVA-OneVision-0.5B (Li et al., 2024a), FastVLM-1.5B (Vasu et al., 2025), LLaVA-Next-7B (Liu et al., 2024a), Qwen2-VL-7B (Wang et al., 2024a), and Janus-Pro-7B (Chen et al., 2025). These models cover both relatively small and large vision tower configurations, allowing us to study the behavior of interface modulation under different vision–language capacity ratios.

Data. We evaluate on both concept-annotated and standard hallucination benchmarks.

CUB-200-2011-POPE (CUB-POPE) (Wah et al., 2011) We construct a POPE benchmark based on CUB-200-2011 using fine-grained bird attribute annotations as concept-level supervision for visual grounding evaluation. Positive questions are generated from annotated present attributes, while negative questions are generated from absent attributes, enabling evaluation of whether models rely on visual evidence or language priors.

We define two concept-selection settings based on CUB confidence annotations: *3/4-concept*, which includes attributes labeled as *likely* or *definite*, and *4-concept*, which retains only *definite* attributes. After filtering invalid samples and duplicates, the benchmark contains approximately 300K QA pairs.

COCO POPE. For evaluation on a general-domain benchmark, we adopt the public POPE benchmark constructed on MS-COCO `val2014`. The benchmark contains object presence queries under *random*, *popular*, and *adversarial* settings, each with approximately 3K samples. Unless otherwise specified, we report results on both the *popular* and *adversarial* splits. The adversarial setting is particularly challenging and more effectively exposes hallucinations driven by language priors, while the popular split evaluates robustness under high-frequency object co-occurrence biases.

Training and Inference. The CBM and FiLM modules are trained in a staged manner with the visual backbone frozen. After visual feature modulation, the decoder is evaluated under the POPE protocol either via generative Yes/No fine-tuning or within our proposed pipeline, and we explicitly indicate whether comparisons are made under inference-only settings.

Table 1: Main results on CUB-POPE and COCO-POPE (popular and adversarial). Acc denotes overall accuracy, P/R denote precision and recall, respectively, and Yes% indicates the ratio of affirmative responses. Higher precision generally reflects fewer hallucinated positive predictions and is therefore a particularly informative metric for evaluating hallucination behavior. We do not include VDGD in the COCO-POPE comparison because its decoding strategy relies on auxiliary image descriptions generated by a VLM and appended to the input prompt, which introduces additional textual grounding signals beyond the original question-answering setting.

	3/4 Concept Setting				**4 Concept Setting**				**COCO POPE (Popular)**				**COCO POPE (Adversarial)**			
method	Acc	P	R	Yes%	Acc	P	R	Yes%	Acc	P	R	Yes%	Acc	P	R	Yes%
OneVision-0.5B																
Baseline	0.5483	0.6773	0.1847	0.1363	0.4260	**0.8552**	0.1881	0.1497	0.8857	0.9488	0.8153	0.4297	**0.8730**	0.9216	0.8153	0.4423
VCD	0.6427	0.6754	0.5493	0.4067	0.6103	0.8220	0.5453	0.4513	**0.8953**	0.9176	**0.8687**	0.4733	0.8690	0.8692	**0.8687**	0.4997
VTI	0.6340	0.6150	0.7167	0.5827	0.6677	0.7791	0.7139	0.6233	0.8867	0.9354	0.8307	0.4440	0.8693	0.9003	0.8307	0.4613
VDGD	0.5940	0.5559	**0.9353**	0.8413	0.5710	0.5393	**0.9740**	0.9030	/	/	/	/	/	/	/	/
Ours	**0.7150**	**0.8415**	0.7532	0.6553	**0.6863**	0.7972	0.7227	0.6167	0.8797	**0.9567**	0.7953	0.4157	0.8680	**0.9306**	0.7953	0.4273
FastVLM-1.5B																
Baseline	0.7276	0.7081	0.7744	0.5467	0.7453	0.8379	0.7758	0.6301	0.8450	0.9568	0.7227	0.3777	0.8370	0.9369	0.7227	0.3857
VCD	0.6973	0.6414	**0.8953**	0.6980	0.7520	0.7811	**0.8829**	0.7690	0.8563	0.9083	**0.7927**	0.4363	0.8400	0.8756	**0.7927**	0.4527
VTI	0.7283	0.6906	0.8273	0.5990	**0.7673**	0.8284	0.8300	0.6817	**0.8617**	0.9436	0.7693	0.4077	**0.8483**	0.9137	0.7693	0.4210
VDGD	0.7067	**0.7180**	0.6807	0.4740	0.7080	0.6919	0.7500	0.5420	/	/	/	/	/	/	/	/
Ours	**0.7315**	0.7124	0.7783	0.5474	0.7541	**0.8430**	0.7847	0.6333	0.8306	**0.9806**	0.6747	0.3441	0.8259	**0.9675**	0.6747	0.3488
LLaVA-Next-7B																
Baseline	0.6776	**0.7327**	0.5593	0.3813	0.6467	**0.8604**	0.5737	0.4537	0.8970	0.9032	0.8893	0.4923	0.8577	0.8364	0.8893	0.5317
VCD	0.7037	0.6721	0.7953	0.5917	0.7323	0.8142	0.7859	0.6567	0.8843	0.8552	**0.9253**	0.5410	0.8237	0.7690	**0.9253**	0.6017
VTI	**0.7083**	0.6697	**0.8220**	0.6137	**0.7527**	0.8118	**0.8285**	0.6943	0.8873	0.9080	0.8620	0.4747	0.8423	0.8294	0.8620	0.5197
VDGD	0.6050	0.6019	0.6200	0.5150	0.6370	0.6121	0.7048	0.6110	/	/	/	/	/	/	/	/
Ours	0.6800	0.7311	0.5693	0.3893	0.6470	0.8594	0.5752	0.4553	**0.8980**	**0.9240**	0.8673	0.4693	**0.8670**	**0.8668**	0.8673	0.5003
Qwen2-VL-7B																
Baseline	0.7173	0.7218	0.7073	0.4900	0.7220	**0.8523**	0.5737	0.4537	**0.8863**	0.9196	**0.8467**	0.4603	0.8680	0.8844	**0.8467**	0.4787
VCD	0.7100	0.6766	**0.8047**	0.5947	0.7323	0.8142	0.7859	0.6567	0.8837	0.9198	0.8407	0.4570	**0.8687**	0.8905	0.8407	0.4720
VTI	**0.7253**	0.7161	0.7467	0.5213	**0.7527**	0.8118	**0.8285**	0.6943	0.8660	0.9140	0.8080	0.4420	0.8513	0.8847	0.8080	0.4567
VDGD	0.6897	0.7072	0.6473	0.4577	0.6740	0.7430	0.5320	0.3580	/	/	/	/	/	/	/	/
Ours	0.7200	**0.7248**	0.7093	0.4893	0.7220	**0.8523**	0.7153	0.5710	0.8830	**0.9436**	0.8145	0.4315	0.8683	**0.9126**	0.8145	0.4461
Janus-Pro-7B																
Baseline	0.6280	0.5769	**0.9600**	0.8320	0.7403	0.7416	**0.9490**	0.8707	**0.7717**	**0.7439**	0.8287	0.5570	**0.7553**	0.7227	0.8287	0.5733
VCD	0.6413	0.5883	0.9420	0.8007	**0.7423**	0.7492	0.9339	0.8480	0.6827	0.6255	0.9107	0.7280	0.6823	0.6252	**0.9107**	0.7283
VTI	0.6880	0.6665	0.7527	0.5647	0.7283	**0.8115**	0.7825	0.6560	0.6467	0.6003	0.8780	0.7313	0.6383	0.5935	0.8780	0.7397
VDGD	**0.7037**	**0.6704**	0.8013	0.5977	0.6840	0.6592	0.7620	0.5780	/	/	/	/	/	/	/	/
Ours	0.6343	0.5824	0.9493	0.8150	0.7343	0.7379	0.9451	0.8713	0.7088	0.7317	**0.9404**	0.9255	0.7494	**0.7229**	0.8222	0.5782

4.2 POPE Binary Question Answering

POPE evaluates hallucination behavior through binary visual question answering. Since hallucinations often arise from insufficient grounding in visual evidence, the fine-grained attribute setting of CUB provides a suitable testbed for concept-sensitive reasoning.

Table 1 compares our method with **VCD** (Leng et al., 2024), **VTI** (Liu et al., 2025), and **VDGD** (Ghosh et al., 2025). Performance should be interpreted across multiple metrics, as some methods may exhibit unstable prediction behavior despite strong scores on individual metrics. Among them, Precision is particularly informative in POPE-style benchmarks because hallucinations mainly appear as false positive predictions on non-existent objects.

Experiments on the POPE hallucination benchmark demonstrate that CoSMo achieves state-of-the-art performance in the small-scale VLM regime and substantially reduces hallucination on precision-sensitive metrics, with precision improvements of up to 16.42% on OneVision-0.5B compared to baseline methods. In particular, CoSMo achieves the best Precision on the challenging COCO-POPE (Adversarial) setting, indicating stronger suppression of visually unsupported affirmative responses.

4.3 Performance under Visual Perturbations

Controlled visual degradation should reduce task performance if decisions use the image rather than language-only continuation. We compare baseline and CBM-FiLM under shared perturbation settings via absolute scores and relative drop w.r.t. clean inputs.

Global perturbations. On the same CUB split, we apply Gaussian noise, pixel dropout, and central block occlusion at matched strengths.

Path A (CBM / FiLM + classifier): Fixed concept labels; accuracy and relative drop on perturbed images (Fig. 3) probe feature-level sensitivity and FiLM modulation.

Path B (Part-mask POPE): For each at-

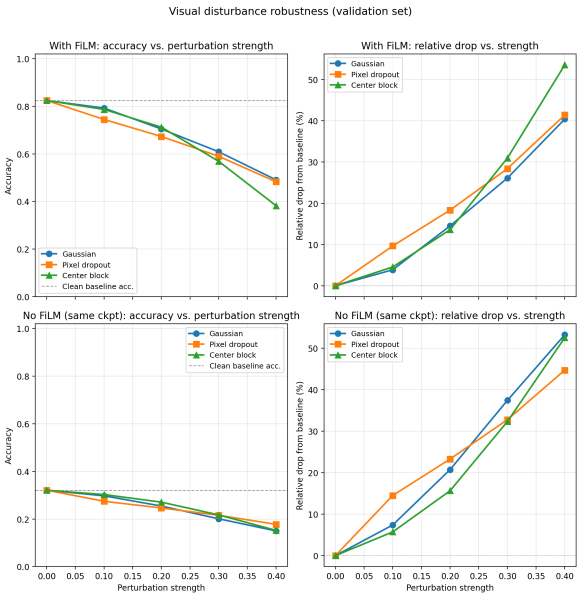

Figure 3: Global perturbations: CBM+FiLM vs. without FiLM (Janus backbone).

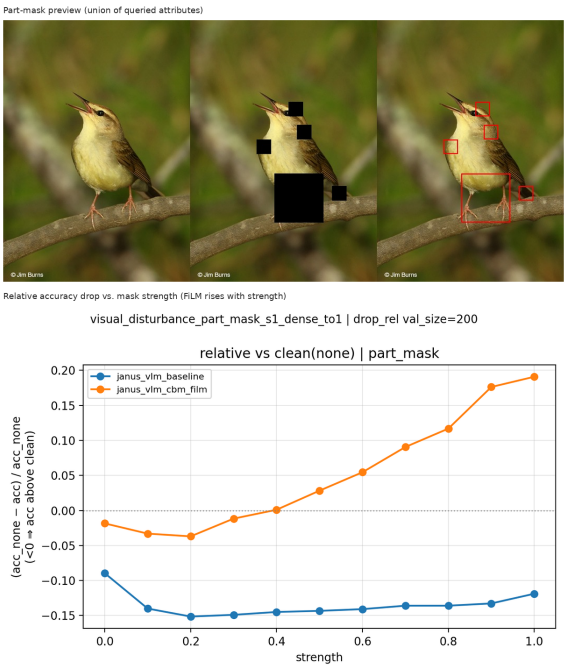

Figure 4: Part masking (POPE Yes/No): preview and relative drop vs. mask strength (baseline vs. CBM-FiLM,Janus backbone).

tribute, we mask CUB keypoint regions (union bbox, strength-dependent expansion) and run the same Yes/No protocol as Sec. 4.2 (N=200, 8 attrs/image, $s \in \{0, 0.1, \ldots, 1.0\}$). Fig. 4: CBM-FiLM degrades more with stronger masks; the baseline is less sensitive.

4.4 Vision Mass Analysis

When prompting VLMs for image attribute description, we analyze the self-attention in the prefill stage (Abnar and Zuidema, 2020). For each layer, we take the attention distribution from the last query position over all tokens, and compute the normalized attention mass assigned to visual placeholder tokens relative to the full sequence. This yields a layer-wise metric termed vision mass (comparable with text mass), which quantifies how much attention budget is allocated to visual slots before token generation begins.

Empirically, after introducing concept-interface modulation, the vision mass curve is generally elevated compared to the baseline, as shown in Fig. 5. Moreover, FiLM with residual connections and layer-wise injection shows a more pronounced increase than standard FiLM, indicating improved utilization of visual information in our model.

4.5 Ablation Studies

4.5.1 Modulation strength and stability

Both feature modulation and latent modulation require explicit control of modulation strength for stable decoding across different VLM configurations. In practice, the language and vision branches are not always well balanced, and excessive modulation can disturb the visual representations expected by the decoder.

Table 2: Effect of the FiLM scaling factor α_2 on POPE using LLaVA-Next-7B under the 3/4 concept-setting benchmark. Only the CBM+FiLM branch is enabled for controlled ablation.

α_2	Acc	P	R	F1	Yes	n_{unparsed}
0.00	0.6930	0.8979	0.6472	0.7522	0.5190	0
0.25	**0.7130**	0.8972	**0.6792**	**0.7731**	0.5450	0
0.50	0.6830	0.8990	0.6306	0.7412	0.5050	0
0.75	0.5430	0.9071	0.4069	0.5618	0.3230	0
1.00	0.4074	0.8690	0.2026	0.3286	0.1669	131

Table 2 shows the effect of the FiLM scaling factor α_2. Moderate modulation strength achieves the best overall performance, while overly strong modulation significantly degrades decoding stability and increases unparsed or biased generations. We refer to such instability as *modulation collapse*, including phenomena such as severe accuracy degradation, invalid outputs, or extreme Yes/No response bias under excessively strong perturbations (formal criteria are provided in Appendix C.2). Setting $\alpha_2 = 0$ disables FiLM modulation and corresponds to the baseline model.

We further study concept-space projection for

Table 3: Ablation study quantifying the contribution of each module. All runs use LLaVA-0.5B and are evaluated on both CUB concept tier splits (c3 and c4).

Setting (matches our scripts)	c3 acc / F1	c4 acc / F1
Pure baseline	0.5483 / 0.2902	0.4260 / 0.3084
Baseline + CBM + FiLM	0.5690 / 0.3609	0.4620 / 0.3891
Baseline + CBM + FiLM + concept space	0.6173 / 0.5035	0.5423 / 0.5369
Baseline + error textual	0.6383 / 0.6352 (α=0.8)	0.6986 / 0.7812 (α=0.8)
Baseline + CBM + FiLM + concept space + error textual	0.6537 / 0.6760 (α=0.8)	0.7050 / 0.7882 (α=1.0)
Baseline + CBM + FiLM + concept space + error textual (projection)	0.6110 / 0.5107 (α=1.0)	0.5623 / 0.5760 (α=1.0)

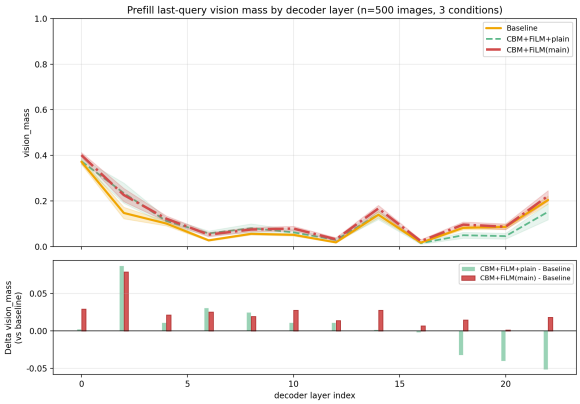

Figure 5: Prefill vision-mass by LM layer on CUB (n=500, LLaVA-OneVision-0.5B). Baseline vs. CBM+FiLM (plain / main checkpoints); modulation at the last vision layer before the LM. Bottom: Δ vs. Baseline.

latent modulation. Projecting modulation directions into the concept subspace improves stability and reduces collapse-like failures under strong modulation. However, this constraint can slightly limit peak performance in some settings, particularly when the language and vision branches exhibit larger capacity mismatch. We therefore treat concept-space projection as an optional regularization strategy rather than a universal default choice.

4.5.2 Module-wise structural ablation

The ablation results in Table 3 are consistent with the roles of each component in CoSMo. The CBM+FiLM module improves concept-aware visual modulation and strengthens alignment between visual representations and the decoder. The concept conditioned memory branch further reduces interference caused by relying on a single static visual memory during decoding. In addition, error-guided latent modulation provides a lightweight correction mechanism that improves the utilization of visual evidence without additional training. The effect of concept-space projection is also consistent with the modulation stability analysis, improving decoding stability under strong modulation while slightly limiting peak performance in some settings.

5 Conclusion

This paper studies object hallucination in LVLM from the perspective of concept-space organization and controllable visual exploitation. To this end, we propose CoSMo, a concept-space-centric modulation framework that systematically improves how visual evidence is represented, conditioned, and utilized during autoregressive decoding. Specifically, CoSMo integrates concept bottleneck based visual modulation, multi-branch concept-conditioned visual memory construction, and lightweight inference-time latent correction within a unified framework. Rather than modifying large backbone parameters, CoSMo enhances the reliability of visual grounding through structured concept-space guidance and controllable feature modulation.

Experiments on both CUB-POPE and COCO-POPE demonstrate that CoSMo achieves consistent improvements in hallucination-sensitive metrics across several representative LVLMs, particularly on Precision under challenging settings. Additional perturbation studies and visual attention analyses further verify improved robustness under corrupted visual inputs and more stable allocation of attention to visual evidence during decoding. Overall, our findings suggest that concept-space-centric representation organization and inference-time modulation provide an interpretable, lightweight, and effective direction for mitigating hallucination in LVLMs.

Limitations

CoSMo is most beneficial when the backbone already provides usable visual evidence. Our re-

sults indicate that concept-guided FiLM modulation tends to help more on compact LVLMs with relatively stronger vision towers (higher effective vision contribution), whereas gains are smaller on some large language-dominated backbones where decoding is less sensitive to interface-level visual reorganization. Practitioners should therefore calibrate FiLM strength (e.g., α_2 in Table 2) rather than applying identical modulation schedules across all model scales.

For inference-time latent correction, projecting error directions into a concept subspace substantially reduces *modulation collapse*, including unparsed generations and extreme Yes/No bias (e.g., nearly all-Yes or all-No responses) observed under overly strong unconstrained perturbations. This safety benefit trades off with peak accuracy: subspace projection can prevent the optimizer from reaching the best unconstrained operating point in exchange for more stable decoding. CoSMo should thus be viewed as prioritizing reliable visual utilization over maximal score chasing, especially when hallucination-sensitive metrics (Precision, Yes%) matter more than raw accuracy alone.

Finally, CBM-FiLM modules are trained on CUB attribute annotations, multi-branch memory increases inference cost linearly in Top-M, and error-direction estimation requires a held-out calibration split disjoint from the reported test set.

References

Samira Abnar and Willem Zuidema. 2020. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4190–4197.

Guillaume Alain and Yoshua Bengio. 2017. Understanding intermediate layers using linear classifier probes.

Wenbin An, Feng Tian, Sicong Leng, Jiahao Nie, Haonan Lin, Qianying Wang, Ping Chen, Xiaoqin Zhang, and Shijian Lu. 2025. 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*.

Qingqing Cao, Bhargavi Paranjape, and Hannaneh Hajishirzi. 2023. PuMer: Pruning and merging tokens for efficient vision language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12890–12903, Toronto, Canada. Association for Computational Linguistics.

Xiaokang Chen and 1 others. 2025. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36:49250–49267.

Sreyan Ghosh, Chandra Kiran Reddy Evuru, Sonal Kumar, Utkarsh Tyagi, Oriol Nieto, Zeyu Jin, and Dinesh Manocha. 2025. Visual description grounding reduces hallucinations and boosts reasoning in lvlms. In *International Conference on Learning Representations*.

Michal Golovanevsky, William Rudman, Michael A. Lepori, Amir Bar, Ritambhara Singh, and Carsten Eickhoff. 2025. Pixels versus priors: Controlling knowledge priors in vision-language models through visual counterfacts. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 24837–24852, Suzhou, China.

Mara Graziani, Laura O'Mahony, An-phi Nguyen, Henning Müller, and Vincent Andrearczyk. 2023. Uncovering Unique Concept Vectors Through Latent Space Decomposition. *Transactions on Machine Learning Research*.

Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. 2024. 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*, pages 13418–13427.

Jiachen Jiang, Jinxin Zhou, Bo Peng, Xia Ning, and Zhihui Zhu. 2026. Analyzing fine-grained alignment and enhancing vision understanding in multimodal language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Zhangqi Jiang, Junkai Chen, Beier Zhu, Tingjin Luo, Yankun Shen, and Xu Yang. 2025. 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*, pages 25004–25014.

Saurav Kadavath, Lawrence Tang, and et al. 2022. Language models (mostly) know what they know. *Transactions of the Association for Computational Linguistics*, 10:909–922.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. 2020. Concept bottleneck models. In *International Conference on Machine Learning*, pages 5338–5348. PMLR.

Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. 2024. 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*, pages 13872–13882.

Bo Li, Yuanhan Zhang, and 1 others. 2024a. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*.

Bohao Li, Yuying Ge, Yixiao Ge, Guangzhi Wang, Rui Wang, Ruimao Zhang, and Ying Shan. 2024b. Seed-bench: Benchmarking multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13299–13308.

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

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2023b. Inference-time intervention: Eliciting truthful answers from a language model. In *Advances in Neural Information Processing Systems*, volume 36, pages 41451–41530. Curran Associates, Inc.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori B Hashimoto, Luke Zettlemoyer, and Mike Lewis. 2023c. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12286–12312.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. 2023d. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 292–305.

Haotian Liu and 1 others. 2024a. Llava-next: Improved reasoning, ocr, and world knowledge. *arXiv preprint arXiv:2401.03374*.

Sheng Liu, Haotian Ye, and James Y Zou. 2025. Reducing hallucinations in large vision-language models via latent space steering. In *International Conference on Learning Representations*.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. 2024b. Mmbench: Is your multi-modal model an all-around player? In *European Conference on Computer Vision*, pages 216–233.

Clement Neo, Luke Ong, Philip Torr, Mor Geva, David Krueger, and Fazl Barez. 2025. Towards interpreting visual information processing in vision-language models. In *International Conference on Learning Representations*.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. 2018. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Goutham Rajendran, Simon Buchholz, Bryon Aragam, Bernhard Schölkopf, and Pradeep Ravikumar. 2024. From causal to concept-based representation learning. In *Advances in Neural Information Processing Systems*, volume 37, pages 101250–101296. Curran Associates, Inc.

Sylvestre-Alvise Rebuffi, Hakan Bilen, and Andrea Vedaldi. 2017. Learning multiple visual domains with residual adapters. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Florian Strub, Mathieu Seurin, Ethan Perez, Harm de Vries, Jeremie Mary, Philippe Preux, and Aaron CourvilleOlivier Pietquin. 2018. Visual reasoning with multi-hop feature modulation. In *The European Conference on Computer Vision (ECCV)*.

Moritz Vandenhirtz, Sonia Laguna, Ričards Marcinkevičs, and Julia E Vogt. 2024. Stochastic concept bottleneck models. *Advances in Neural Information Processing Systems*, 37:51787–51810.

Pavan Kumar Anasosalu Vasu, Fartash Faghri, Chun-Liang Li, Cem Koc, Nate True, Albert Antony, Gokul Santhanam, James Gabriel, Peter Grasch, Oncel Tuzel, and Hadi Pouransari. 2025. Fastvlm: Efficient vision encoding for vision language models. In *CVPR*.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. 2011. The caltech-ucsd birds-200-2011 dataset. Technical report, California Institute of Technology.

Peng Wang and 1 others. 2024a. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Qun Wang and Feng Zhu. 2025. A simple yet efficient multimodal feature modulation adapter for visual grounding. In *2025 7th International Conference on Electronics and Communication, Network and Computer Technology (ECNCT)*, pages 306–310.

Weihan Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang, Lei Zhao, Xixuan Song, and et al. 2024b. Cogvlm: Visual expert for pretrained language models. *Advances in Neural Information Processing Systems*, 37:121475–121499.

Hantao Yao, Rui Zhang, and Changsheng Xu. 2023. Visual-language prompt tuning with knowledge-guided context optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6757–6767.

Mert Yuksekgonul, Maggie Wang, and James Zou. 2023. Post-hoc concept bottleneck models. In *The Eleventh International Conference on Learning Representations*.

A Training

A.1 Data Splits and Evaluation Sets

- **CBM/FiLM training:** CUB-200-2011 attribute annotations; validation split is class-stratified (30% validation ratio).
- **Error-direction estimation:** directions $\{\Delta \mathbf{h}_\ell\}$ are estimated on *independent* training splits for each difficulty tier (C3/C4), with *zero overlap* with the final reported test sets.
- **Reported test sets:** C3 holdout contains 3000 samples (balanced yes/no); C4 uses a fixed 3000-sample subset; COCO POPE is an out-of-domain adversarial set with approximately 3K samples.
- **Construction pool:** approximately 300K CUB QA pairs cover classes and attributes; main-text POPE results use the fixed 3K evaluation subset described above.

A.2 Three-Stage CBM Training

The concept bottleneck (CBM) and modulation network are trained in three stages; the visual backbone is aligned with the vision-tower weights of the base VLM used in downstream evaluation.

Stage 1 optimizes only g_{cbm} (masked BCE on $\mathbf{c}$). Gradient clipping uses $\|\nabla\|_2 \leq 1$. The learning rate is multiplied by 0.5 when the validation metric plateaus (patience 5), with early-stopping patience 12. The concept-positive threshold is τ_c=0.5 after sigmoid. The final layers of the modulation head are zero-initialized so that $\boldsymbol{\beta}^{(\ell)}(\mathbf{z}) \equiv \mathbf{0}$ at initialization and $\Delta\mathbf{V}^{(\ell)}(\mathbf{z})$ is near zero (Sec. 3.3).

Table 4 summarizes epochs, batch size, and learning rates (LLaVA-Mistral-7B with CUB C4 concept annotations).

Table 4: Three-stage CBM training hyperparameters (LLaVA-Mistral-7B with CUB C4 concept annotations).

Stage	Epochs	Batch	Learning Rate
Concept BCE (backbone frozen by default)	100	8	10^{-4}
Modulation + classifier CE	100	8	10^{-4}
Joint finetuning (λ_{concept}=0.5)	10	8	10^{-5}

A.3 Error-Direction Estimation and POPE Evaluation Protocol

Error directions are estimated independently on the training split of each tier:

$$\Delta\mathbf{h}_\ell = \mathbb{E}[\mathbf{h}_\ell \mid \mathcal{D}_{\text{err}}] - \mathbb{E}[\mathbf{h}_\ell \mid \mathcal{D}_{\text{ok}}], \quad \ell = 1, \ldots, L. \tag{14}$$

where $\mathbf{h}_\ell$ denotes the post-MLP hidden state of the **final token** at decoder layer ℓ (embedding layer excluded), consistent with Sec. 3.5.

On C3/C4 we report binary concept-QA accuracy and yes ratio. Compared settings are: (A) unmodulated VLM backbone; (B) CBM visual modulation without concept-subspace fusion; (C) (B) plus error-conditioned latent modulation. Default inference scalings are α_{film}=0.25 (visual offset) and $\alpha_1 \in \{0.125, 0.25, 0.5, 0.8, 0.9, 1.0\}$ with η=0.1 (textual direction). Stability of full-dimensional versus concept-projected injection under this α_1 sweep is reported in Table 5.

B Algorithmic Details

B.1 CBM Concept Head and Backbone Extraction

The vision encoder yields layerwise tokens $\mathbf{V}^{(\ell)} \in \mathbb{R}^{(1+N)\times d}$. At the early anchor $\ell_{\text{early}} = \lfloor L/4 \rfloor$ (Sec. 3.3), concept logits and confidences are obtained as in Sec. 3.2:

$$\begin{aligned} \mathbf{z} &= g_{\text{cbm}}\big(\text{POOL}(\mathbf{V}^{(\ell_{\text{early}})})\big) \in \mathbb{R}^{K_{\text{all}}}, \\ \mathbf{c} &= \sigma(\mathbf{z}) \in (0,1)^{K_{\text{all}}}. \end{aligned} \tag{15}$$

In our implementation, $\text{POOL}(\cdot)$ uses the CLS token of $\mathbf{V}^{(\ell_{\text{early}})}$. The head g_{cbm} is a two-stage MLP ($512 \to 256$, ReLU, dropout 0.2). Stage 1 trains only g_{cbm}; Stages 2/3 additionally attach the modulation network and a linear classifier on $\mathbf{z}$ (Table 4).

B.2 Residual Semantic Modulation (CBM Training)

Following Eq. (2), the modulation network maps $\mathbf{z}$ to channel-wise FiLM parameters at ℓ_{early}, $\ell_{\text{mid}} = \lfloor L/2 \rfloor$, and $\ell_{\text{late}} = L-1$. Under the additive-only training setting, $\boldsymbol{\gamma}^{(\ell)}(\mathbf{z}) \equiv \mathbf{1}$ and injection is applied to patch tokens:

$$\widetilde{\mathbf{V}}^{(\ell)}_{1:N} = \mathbf{V}^{(\ell)}_{1:N} + \boldsymbol{\beta}^{(\ell)}(\mathbf{z}) = \mathbf{V}^{(\ell)}_{1:N} + \Delta\mathbf{V}^{(\ell)}(\mathbf{z}). \tag{16}$$

Concept logits are read out from ℓ_{early}; modulation is never applied before that layer. Zero initialization of the modulation head yields $\boldsymbol{\beta}^{(\ell)}(\mathbf{z}) \equiv \mathbf{0}$ at the start of training.

B.3 Visual FiLM During VLM Inference

At evaluation, the same CBM produces concept logits $\mathbf{z}$ and predicts $(\boldsymbol{\gamma}^{(\ell)}(\mathbf{z}), \boldsymbol{\beta}^{(\ell)}(\mathbf{z}))$ for the configured visual depth(s). On patch features $\mathbf{V} \in \mathbb{R}^{N\times D}$ (CLS removed),

$$\widetilde{\mathbf{V}} = \mathbf{V} + \big(\boldsymbol{\gamma}^{(\ell)}(\mathbf{z}) - \mathbf{1}\big) \odot \mathbf{V} + \boldsymbol{\beta}^{(\ell)}(\mathbf{z}) = \mathbf{V} + \Delta\mathbf{V}^{(\ell)}(\mathbf{z}), \tag{17}$$

which is equivalent to Eq. (2). Optional inference scaling applies only to the offset (Sec. C.1, Eq. (28)).

The modulated visual tokens are then projected and packed into the baseline visual memory $\mathbf{V}_{\text{base}}$ fed to the multimodal decoder (Sec. 3.4).

B.4 Concept-Axis Visual Memory and Fusion

Let $\mathbf{V}_{\text{base}}$ denote the packed baseline memory obtained from the full concept logits $\mathbf{z}$ after a single CBM $\rightarrow$ FiLM $\rightarrow$ pack pass.

We select the Top-M activated concepts (implementation: M=32, $M \ll K_{\text{all}}$):

$$\mathcal{K} = \{k_1, \ldots, k_M\}. \tag{18}$$

For each $k \in \mathcal{K}$, we construct the axis-aligned concept vector as in Sec. 3.4:

$$\mathbf{z}_k = s\,\mathbf{e}_k, \tag{19}$$

where $\mathbf{e}_k$ is the one-hot basis vector along concept axis k and $s \gg 0$ is a fixed activation strength (not the raw logit z_k).

Each $\mathbf{z}_k$ is independently passed through

$$\text{FiLM} \rightarrow \text{pack},$$

without re-applying g_{cbm}, yielding $\mathbf{V}_k$. Memories are stacked as

$$\mathcal{V} = \{\mathbf{V}_k\}_{k\in\mathcal{K}} \in \mathbb{R}^{M\times T_{\text{v}}\times D}. \tag{20}$$

Fusion uses normalized concept weights and residual mixing coefficient λ (Sec. 3.4):

$$\begin{aligned} \hat{w}_k &= \frac{\exp(|z_k|)}{\sum_{j\in\mathcal{K}} \exp(|z_j|)}, \\ \mathbf{V}_{\text{in}} &= \mathbf{V}_{\text{base}} + \lambda \sum_{k\in\mathcal{K}} \hat{w}_k\, \mathbf{V}_k. \end{aligned} \tag{21}$$

Default λ=0.5 in our experiments. Alternatively, $\mathcal{V}$ may be passed to layerwise concept-scheduling fusion without forming $\mathbf{V}_{\text{in}}$ explicitly.

B.5 Error-Conditioned Latent Modulation

Offline direction estimation. On the calibration split, samples are partitioned into $\mathcal{D}_{\text{err}}$ and $\mathcal{D}_{\text{ok}}$ according to POPE binary labels. For each sample, multimodal prefilling is performed and final-token hidden states are aggregated:

$$\Delta\mathbf{h}_\ell = \mathbb{E}[\mathbf{h}_\ell \mid \mathcal{D}_{\text{err}}] - \mathbb{E}[\mathbf{h}_\ell \mid \mathcal{D}_{\text{ok}}], \qquad \Delta\mathbf{h}_\ell \in \mathbb{R}^D. \tag{22}$$

Inference-time modulation. At decoder layer ℓ, modulation is applied after the feed-forward (MLP) submodule. Let $\mathbf{h}_\ell \in \mathbb{R}^D$ denote the post-MLP hidden state of the final token, with $n_\ell = \|\mathbf{h}_\ell\|_2$. We normalize

$$\widehat{\mathbf{h}}_\ell = \frac{\mathbf{h}_\ell}{\|\mathbf{h}_\ell\|_2}. \tag{23}$$

Optionally, we project the offline direction onto the concept-embedding subspace:

$$\widetilde{\Delta\mathbf{h}}_\ell = \Pi_{\mathcal{S}}(\Delta\mathbf{h}_\ell) = (\Delta\mathbf{h}_\ell\,\mathbf{Q})\,\mathbf{Q}^\top, \tag{24}$$

and define

$$\widehat{\mathbf{d}}_\ell = \frac{\widetilde{\Delta\mathbf{h}}_\ell}{\|\widetilde{\Delta\mathbf{h}}_\ell\|_2}$$

(25)

The modulated state matches Sec. 3.5:

$$\begin{aligned} \mathbf{h}'_\ell &= n_\ell \cdot \text{normalize}\Big(\widehat{\mathbf{h}}_\ell - \eta\,\alpha_1\,\widehat{\mathbf{d}}_\ell\Big), \\ \eta &= 0.1 \qquad \text{(default)}. \end{aligned} \tag{26}$$

This preserves $\|\mathbf{h}'_\ell\|_2 = n_\ell$ and shifts the unit-norm state *away* from the hallucination-prone direction $\widehat{\mathbf{d}}_\ell$. The pathway acts only on the language decoder and is independent of visual FiLM.

Concept-subspace projection. Concept embeddings $\mathbf{e}_k \in \mathbb{R}^D$ are built from the concept vocabulary via the input embedding layer (attention-mask-weighted averaging per concept name). Reduced QR on $[\mathbf{e}_1; \ldots; \mathbf{e}_M]^\top$ yields an orthonormal basis $\mathbf{Q} \in \mathbb{R}^{D\times R}$ spanning subspace $\mathcal{S}$ (Sec. 3.5). Quantitative stability with versus without $\Pi_{\mathcal{S}}$ is given in Table 5.

B.6 Linear Readout of Cross-Modal Attention

Under the value-only assumption,

$$\mathrm{Attn}(\mathbf{h}_t, \widetilde{\mathbf{V}}) = \mathrm{Attn}(\mathbf{h}_t, \mathbf{V}_{\mathrm{in}}) + \sum_k c_k \, \mathbf{A}_t \mathbf{u}_k \mathbf{W}_V. \tag{27}$$

With key–value coupling, the expression is only a first-order interpretation around the unmodulated operating point.

C Concept Subspace and Modulation

C.1 Inference-Time Scaling

Visual-side scaling (α_{film}). Let $\Delta\mathbf{V}^{(\ell)}(\mathbf{z})$ be the concept-induced offset in Eq. (2). At inference we optionally scale only the offset:

$$\widetilde{\mathbf{V}}^{(\ell)} = \mathbf{V}^{(\ell)} + \alpha_{\mathrm{film}} \, \Delta\mathbf{V}^{(\ell)}(\mathbf{z}). \tag{28}$$

When $\boldsymbol{\gamma}^{(\ell)}(\mathbf{z}) \neq \mathbf{1}$, α_{film} is applied identically to the residual channel-scale and bias terms in Eq. (2). α_{film}=0 yields $\widetilde{\mathbf{V}}^{(\ell)} = \mathbf{V}^{(\ell)}$; α_{film}=1 applies full CBM modulation.

Text-side scaling (α_1). This is the scalar α_1 in Sec. 3.5 and does not pass through visual FiLM. Without concept projection, smaller α_1 is generally preferred (Table 5).

C.2 Concept-Subspace Constraints and Modulation Failure

Motivation. Using full-dimensional $\Delta\mathbf{h}_\ell$ with large α_1 often causes parsing failures and extreme yes ratios (*modulation failure*). Projection $\widetilde{\Delta\mathbf{h}}_\ell = \Pi_{\mathcal{S}}(\Delta\mathbf{h}_\ell)$ suppresses off-subspace components. On C4, the Frobenius norm of the direction matrix decreases from 20.45 to 1.70 (about 8.3% at α_1=1.0).

Failure criteria. Relative to the unmodulated baseline on the same split, a run fails if any of the following holds: (1) accuracy drops by at least 15 percentage points; (2) yes ratio < 0.05 with recall ≈ 0; (3) yes ratio > 0.95 with severe precision degradation.

Table 5 reports paired comparisons under this protocol (LLaVA-v1.6-Mistral-7B; η=0.1; directions from 800 err/ok pairs, seed 42). Bold entries indicate failure (accuracy drop $\geq$15 pp vs. unmodulated baseline on the same split).

Full-dimensional vs. concept projection. *Full-dimensional:* $\Delta\mathbf{h}_\ell$ is estimated and injected in the full hidden space; at most 800 error/correct samples per subset are used when constructing directions. *Concept projection:* $\Delta\mathbf{h}_\ell$ is replaced by $\widetilde{\Delta\mathbf{h}}_\ell$ before injection; directions use all available error/correct samples from the training split (M=32 concept axes). As shown in Table 5, full-dimensional injection fails under large α_1, whereas concept projection remains stable across all six settings.

C.3 Two Forms of Concept Subspace Construction

Visual-axis stacking. For Top-M concepts, each axis uses $\mathbf{z}_k = s\mathbf{e}_k$ with fixed $s \gg 0$. A single vision-tower forward pass supplies features; each axis independently undergoes FiLM $\rightarrow$ pack (no second CBM readout), yielding $\mathcal{V} \in \mathbb{R}^{M \times T_{\mathrm{v}} \times D}$ and fusion via $\hat{w}_k \propto \exp(|z_k|)$ and λ.

Textual concept-embedding subspace. The concept vocabulary induces embeddings $\mathbf{e}_k$ and basis $\mathbf{Q}$, defining $\mathcal{S}$ used to constrain $\Delta\mathbf{h}_\ell$ before forming $\widehat{\mathbf{d}}_\ell$ (Table 5).

D Supplementary Experimental Results

D.1 Full-Dimensional vs. Concept Projection

Table 5 compares the unmodulated baseline, CBM+FiLM, and error-conditioned modulation under a unified protocol (α_{film}=0.25, η=0.1, LLaVA-v1.6-Mistral-7B). Bold entries denote failures under Sec. C.2. In the full-dimensional setting, large α_1 often drives the yes ratio toward zero. Concept projection avoids failure across all six α_1 settings, keeping accuracy and yes ratio close to the baseline.

With α_{film}=0.25, the full-dimensional setting fails in 4 of 6 α_1 configurations on C3 and 5 of 6 on C4, whereas concept projection exhibits no failures (Table 5). When additionally scanning $\alpha_{\mathrm{film}} \in \{0.5, 0.75, 1.0\}$ (24 full-dimensional configurations), failures increase to 18/24 on C3 and 22/24 on C4.

For full-dimensional runs with $\alpha_1 \geq 0.5$ on C3 (or larger α_1 on C4), the yes ratio typically collapses toward zero and accuracy drops near chance, indicating language-prior dominance over visual evidence. After projection, accuracy varies within $\pm$1.5 percentage points of the unmodulated baseline across all six α_1 settings, with yes-ratio deviation below 3 percentage points.

D.2 Multiple Backbones and Ablation Summary

Multi-backbone POPE comparisons and module-wise ablations are reported in the main exper-

Table 5: Unmodulated baseline and error-textual modulation (LLaVA-v1.6-Mistral-7B; η=0.1; 800 err/ok pairs, seed 42). **Full**: no $\Pi_{\mathcal{S}}$; **Proj**: α_{film}=0.25, K=32 with $\Pi_{\mathcal{S}}$. Acc / Yes%; bold = failure ($\geq$15 pp acc drop vs. baseline on the same split).

Split	Setting	α_{film}	Acc / Yes% at α_{err}					
			0.125	0.25	0.5	0.8	0.9	1.0
C3	Baseline	0	0.676 / 40.9%					
	Full (no $\Pi_{\mathcal{S}}$)	0.25	0.613 / 21.8%	0.556 / 9.7%	**0.488 / 0.5%**	**0.483 / 0.0%**	**0.483 / 0.0%**	**0.483 / 0.0%**
	Proj ($\Pi_{\mathcal{S}}$)	0.25	0.672 / 40.9%	0.669 / 39.8%	0.665 / 38.6%	0.668 / 38.7%	0.662 / 39.1%	0.660 / 39.7%
C4	Baseline	0	0.662 / 45.4%					
	Full (no $\Pi_{\mathcal{S}}$)	0.25	0.565 / 30.3%	**0.455 / 15.1%**	**0.324 / 1.0%**	**0.314 / 0.0%**	**0.314 / 0.0%**	**0.314 / 0.0%**
	Proj ($\Pi_{\mathcal{S}}$)	0.25	0.650 / 44.2%	0.646 / 43.2%	0.629 / 40.5%	0.621 / 38.7%	0.619 / 38.9%	0.620 / 39.4%

iments section. Smaller models with stronger visual reliance (e.g., OneVision-0.5B) tend to show larger gains in precision and F1 under the full CoSMo pipeline. Progressive ablations on LLaVA-0.5B indicate gains from CBM+FiLM, concept-subspace branches ($\mathcal{V}$, $\hat{w}_k$, λ), and error-conditioned modulation ($\Delta\mathbf{h}_\ell$, α_1). After concept projection, peak accuracy may decrease slightly while α_1 stability matches Table 5. On LLaVA-Next-7B, excessively large α_{film} can cause unparsable Yes/No responses, a visual-side analogue of over-modulation.