From Heads to Neurons: Causal Attribution and Steering in Multi-Task Vision–Language Models

Qidong Wang[1], **Junjie Hu**[2], **Ming Jiang**[2*]
[1]Tongji University, [2]University of Wisconsin-Madison
wang_qidong@tongji.edu.cn, {junjie.hu, ming.jiang}@wisc.edu

Abstract

Recent work has increasingly explored neuron-level interpretation in vision-language models (VLMs) to identify neurons critical to final predictions. However, existing neuron analyses generally focus on single tasks, limiting the comparability of neuron importance across tasks. Moreover, ranking strategies tend to score neurons in isolation, overlooking how task-dependent information pathways shape the write-in effects of feed-forward network (FFN) neurons. This oversight can exacerbate neuron polysemanticity in multi-task settings, introducing noise into the identification and intervention of task-critical neurons. In this study, we propose HONES (**H**ead-**O**riented **N**euron **E**xplanation & **S**teering), a gradient-free framework for task-aware neuron attribution and steering in multi-task VLMs. HONES ranks FFN neurons by their causal write-in contributions conditioned on task-relevant attention heads, and further modulates salient neurons via lightweight scaling. Experiments on four diverse multimodal tasks and two popular VLMs show that HONES outperforms existing methods in identifying task-critical neurons and improves model performance after steering. Our source code is released at: https://github.com/petergit1/HONES.

1 Introduction

Large vision-language models (VLMs) have demonstrated strong multi-task capabilities across a wide range of vision-language applications, including visual question answering (VQA) (Liu et al., 2023; Dai et al., 2023), optical character recognition (OCR) (Ye et al., 2023; Hu et al., 2025), and image captioning (Li et al., 2023; Liu et al., 2025). Despite their impressive performance, the internal decision-making process of these models remains opaque, as multiple capabilities are entangled within shared parameters, hindering error attribution and limiting reliability and controllability in real deployments. To address this gap, recent studies have increasingly focused on model interpretability, particularly neuron-level analysis, which offers fine-grained and actionable insights for model diagnosis and editing (Lin et al., 2025; Shu et al., 2025; Meng et al., 2022).

Existing neuron analysis largely focuses on large language models (LLMs) (Zhao et al., 2024; Tang et al., 2024; Yu and Ananiadou, 2024). Recently, this line of work has begun to extend to multimodal settings (Huang et al., 2024; Huo et al., 2024; Pach et al., 2025). Research in this area primarily follows two strands: (1) ranking neurons to identify those most salient to model prediction (Wang et al., 2025a; Dang et al., 2024), and (2) analyzing the semantic information encoded by individual neurons (Pach et al., 2025; Sajjad et al., 2022).

Despite remarkable progress, existing neuron analysis methods face two key limitations. First, they typically focus on interpreting neurons within a single task (e.g., VQA), leaving the comparability of these interpretability methods across tasks underexplored. This issue becomes particularly pronounced for tasks with distinct characteristics and heterogeneous outputs, such as question answering versus image–text matching. Second, many approaches analyze neurons in isolation, which leads to high computational cost, especially in large VLMs with expansive FFN layers, and becomes even more prohibitive in long-context multimodal settings. Moreover, treating neurons independently while overlooking task-relevant routing context from attention heads may exacerbate polysemanticity in multi-task VLMs (Oikarinen and Weng, 2024; Haider et al., 2025). As a result, neurons participating in multiple routing paths can receive inflated importance scores, ultimately leading to noisier identification of truly task-relevant neurons.

To address the aforementioned issues, we propose **HONES** (Head-Oriented Neuron Explanation

*corresponding author

Findings of the Association for Computational Linguistics: ACL 2026, pages 36151–36175

& Steering), a unified, gradient-free framework for context-guided neuron attribution in multi-task VLMs, enabling consistent interpretability across heterogeneous readouts. Specifically, HONES builds upon a structured causal view of model computation, where attention heads select and route task-critical inputs, while downstream FFN neurons write the routed information into the residual stream for final prediction (Elhage et al., 2021). Following this consideration, we first localize task-critical attention heads via causal interventions and then use these routing signals to guide downstream neuron attribution. With our findings, we further introduce a lightweight steering method that freezes the backbone and learns only sparse scaling factors over the identified task-critical neurons, enabling controlled task-specific improvements.

We perform HONES on four diverse multimodal tasks (i.e., VQA, OCR, captioning, and image-to-text retrieval) and evaluate it on two representative VLMs (i.e., LLaVA and Qwen). Our key findings include: (1) HONES outperforms state-of-the-art neuron ranking methods in identifying task-critical neurons across all tasks and both VLMs; (2) task-critical neurons present task-dependent layer preferences; (3) key neurons shared across multiple tasks are more salient than task-specific ones, particularly those overlapping with VQA; and (4) our lightweight steering method effectively improves performance across all four tasks on both VLMs.

Overall, our main contributions are as follows:

- We propose **HONES**, a head-conditioned and gradient-free framework for neuron-level causal attribution and steering in multi-task VLMs, where FFN neurons are identified under task-relevant routing context rather than scored in isolation.
- We uncover a variety of novel insights into task-critical neuron patterns that advance the understanding of VLM mechanisms in cross-task generalization.
- We apply HONES to improve model performance via lightweight steering on the identified task-critical neurons. Experiments on four diverse multimodal tasks and two VLMs show consistent improvements.

2 Related Work

Large VLMs. Existing large VLMs typically combine a powerful visual encoder with an LLM. Popular training strategies include in-context multimodal conditioning like Flamingo (Alayrac et al., 2022) and visual instruction tuning for general-purpose task following, such as LLaVA (Liu et al., 2023) and InstructBLIP (Dai et al., 2023). These models, especially open-source ones, have made steady progress in supporting higher-resolution visual inputs and document-centric understanding, as demonstrated by InternVL (Chen et al., 2024), mPLUG-Owl2 (Ye et al., 2024), and Qwen2.5-VL (Bai et al., 2025). Despite offering similar functionality, they differ in how visual signals are injected and how multi-task behaviors are routed internally, motivating mechanistic analysis across backbones. In this work, we focus on two representative models—LLaVA-1.5-7B (Liu et al., 2023) and Qwen2.5-VL-7B (Bai et al., 2025) to evaluate the generalizability of our findings and steering method across different model architectures.

Neuron-level Interpretability. Existing neuron-level interpretability in VLMs largely inherits strategies from LLM-based studies. For example, prior work views FFN/MLP blocks as memory- and computation-like units that "write" into the residual stream, localize key neurons associated with facts/capabilities, and validate their causal roles through targeted interventions (Geva et al., 2021; Dai et al., 2022). Recently, dictionary learning and sparse autoencoders (SAEs) have highlighted that *features* (rather than individual neurons) offer stable analysis units, mitigating polysemanticity and enhancing interpretability (Bricken et al., 2023).

Building on this foundation, existing neuron-level studies for VLMs can be organized into two complementary streams: (1) causal analysis of model structure and (2) semantic analysis of latent representations. The former focuses on identifying which components causally drive the output behavior, and includes intervention-based analyses and activation-based neuron discovery followed by ablation or gating validation (e.g., domain-specific neurons (Huo et al., 2024), modality-specific neurons (Huang et al., 2024), and culture-sensitive neurons (Zhao et al., 2026)). It also covers readout-aligned neuron scoring, including prediction-probability-change methods (Yu and Ananiadou, 2024), as well as broader attribution methods that anchor neuron importance to output-side changes, yielding more direct and testable contribution mappings (Schwettmann et al., 2023; Pan et al., 2024; Fang et al., 2024; Wang et al., 2025a).

Relatedly, MultEdit (Basu et al., 2024) provides complementary mechanistic evidence by editing early causal MLP blocks in multimodal LLMs. The latter is mainly represented by SAE-based sparse dictionary learning, which learns sparse and more monosemantic feature factors to characterize how high-level semantics are organized in representation space, and has been extended to visual or vision-language representations to support interpretability analysis and steering (Pach et al., 2025; Lim et al., 2025).

Our work differs from existing methods in three aspects. First, unlike intervention-based analyses that rely on token-level objectives and counterfactual constructions, HONES provides a unified scoring interface across heterogeneous task readouts, which is particularly useful for open-ended generation and retrieval ranking. Second, unlike readout-aligned attribution methods, HONES conditions neuron scoring on localized evidence-routing attention heads, yielding cleaner and more comparable neuron sets under shared parameters. Third, HONES is gradient-free and scalable: once key routing heads are identified, it requires only a constant number of additional forward passes and computes all FFN neuron scores jointly in a vectorized manner, avoiding the per-unit patching bottleneck of fine-grained causal methods. Compared with feature-level approaches such as SAEs, HONES is also model-native and directly supports causal attribution and lightweight steering on the original backbone without additional feature learning.

3 Preliminaries

Multi-task VLM. We consider a multi-task VLM with parameters θ over heterogeneous tasks $\mathcal{T} = \{T_{\text{vqa}}, T_{\text{ocr}}, T_{\text{cap}}, T_{\text{ret}}, \ldots\}$. For a task $t \in \mathcal{T}$, each example is a labeled pair $(x, y) \in \mathcal{D}_t$, where the input is a multimodal sequence $x = [x_v, x_t]$ (visual and textual tokens), and y is the task-specific ground-truth output. Given x, the model produces a task prediction $\hat{y}$ (e.g., short answers for VQA, character sequences for OCR, free-form captions for captioning, or decisions/rankings for retrieval). Note that our focus differs from multi-domain investigations, which typically study a fixed task across substantially different visual distributions (e.g., natural to medical imagery) (Huo et al., 2024). In contrast, we focus on heterogeneous tasks within a shared visual domain, enabling controlled cross-task comparison of neuron attribution without confounding from domain shift.

Neuron Definition. Following Huo et al. (2024), we focus on the two FFN layers at a Transformer layer ℓ, denoted as $\mathbf{W}_{\text{up}}^{(\ell)} \in \mathbb{R}^{d \times d_{\text{ff}}}$ and $\mathbf{W}_{\text{down}}^{(\ell)} \in \mathbb{R}^{d_{\text{ff}} \times d}$. Let $\mathbf{z}^{(\ell)} \in \mathbb{R}^{d_{\text{ff}}}$ be the intermediate activations of the last token in x after the first FFN at layer ℓ, which directly affects next-token prediction in autoregressive decoding. We then define the i-th neuron at layer ℓ as the i-th element of $\mathbf{z}^{(\ell)}$, denoted as $z_i^{(\ell)} \in \mathbb{R}$, which is associated with its input weight $\mathbf{W}_{\text{up}}^{(\ell)}[:, i]$ and output weight $\mathbf{W}_{\text{down}}^{(\ell)}[i, :]$. This definition ties a neuron to a concrete, manipulable computation in the model's FFN parameters. Formally, we denote the index set of all neurons in θ as $\mathcal{U} = \{(\ell, i) \mid \ell \in [1, L], i \in [1, d_{\text{ff}}]\}$.

Unified Dataset and Splits. We start from a unified multi-task dataset $\mathcal{D}$ constructed on a shared image set. For each task t, we derive the task-specific labeled set $\mathcal{D}_t \subseteq \mathcal{D}$ by keeping the same image x_v and only changing the task instruction x_t to form x with the corresponding label y. We split each $\mathcal{D}_t$ into three disjoint subsets: $\mathcal{D}_t^{\text{disc}}$, $\mathcal{D}_t^{\text{dev}}$, and $\mathcal{D}_t^{\text{test}}$, with no overlap in images across splits. We use $\mathcal{D}_t^{\text{disc}}$ for head/neuron discovery (§4.1, §4.2), $\mathcal{D}_t^{\text{dev}}$ for learning the scaling factors (§4.3), and use $\mathcal{D}_t^{\text{test}}$ only to *verify* causal importance by masking the discovered neurons and measuring the induced performance drop.

Instance-level Task Performance as a Scalar. We define an instance-level task performance scalar $\mathcal{P}_t(x, y; \theta) \in \mathbb{R}$ where higher is better. Concretely, $\mathcal{P}_t$ can be instantiated by the task-specific evaluation metric applied to the model prediction (e.g., VQAv2 accuracy, ANLS, BLEU-4, or NDCG@5 under a fixed evaluation protocol).

$$\mathcal{P}_t(x, y; \theta) \;=\; \text{Metric}_t\big(\hat{y}, y\big). \tag{1}$$

Problem Definition For any neuron subset $\mathcal{S} \subset \mathcal{U}$, let $\mathcal{I}_{\mathcal{S}}$ be a masking operator that suppresses activations of neurons in $\mathcal{S}$ at inference time, yielding an intervened model $\theta_{\mathcal{I}_{\mathcal{S}}}$. We quantify the causal importance of $\mathcal{S}$ on a labeled example (x, y) by the performance drop:

$$\Delta\mathcal{P}_t\big((x, y); \mathcal{S}\big) = \mathcal{P}_t(x, y; \theta) - \mathcal{P}_t\big(x, y; \theta_{\mathcal{I}_{\mathcal{S}}}\big).$$

Our goal is to identify, for each task $t \in \mathcal{T}$, a compact set $\mathcal{N}_t^*$ of K task-critical neurons whose causal intervention induces a large expected performance drop under the task utility $\mathcal{P}_t$ for data

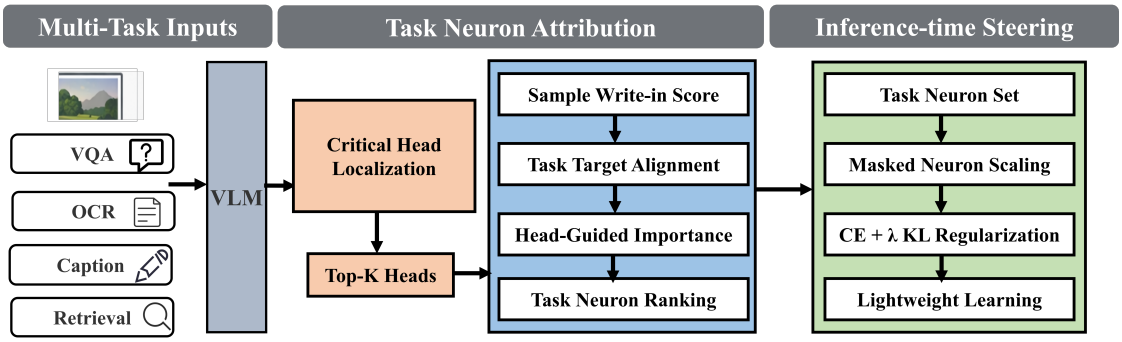


Figure 1: Overview of HONES. Left: Discovery task neurons via head-guided, readout-aligned write-in scoring. Right: Steering employs sparse activation scaling on a frozen backbone to enhance task performance.

in $\mathcal{D}_t^{\text{test}}$, i.e., masking these neurons should significantly reduce task performance. We discover such neurons on the discovery split $\mathcal{D}_t^{\text{disc}}$ and evaluate the induced performance drop $\Delta\mathcal{P}_t$ on the held-out test split $\mathcal{D}_t^{\text{test}}$ to assess generalization.

4 Method

Overview. Figure 1 overviews HONES, a two-stage framework for readout-aligned neuron discovery and causal steering in multi-task VLMs. For **discovery**, we localize a sparse set of task-critical attention heads $\mathcal{H}_t^*$ via causal head interventions (§4.1), then identify task-critical FFN neurons by measuring each neuron's *task-target-aligned* causal write-in contribution conditioned on $\mathcal{H}_t^*$ using *direct vocabulary projection* (§4.2). In **steering** (§4.3), we freeze the model θ and learn only sparse neuron-wise scaling factors on $\mathcal{N}_t^*$ with a KL-regularized objective, which both *verifies* causality and enables controllable task improvements.

4.1 Critical Head Localization

Our primary objective is to identify task-critical FFN neurons. Given that FFN write-ins are routed and aggregated via self-attention mechanisms, we first localize a sparse set of critical "routing nodes" (attention heads) prior to neuron-level attribution. This step is essential to constrain the search space and isolate effective computational pathways.

To this end, we adopt V-SEAM (Wang et al., 2025b), a state-of-the-art causal head localization method with a mean-replacement intervention operator. Let $\mathbf{o}^{(h)}(x)$ be the output of head h in a certain layer. The mean-replacement intervention $\mathcal{I}_h$ replaces this head's output with the mean of the remaining $H-1$ heads:

$$\tilde{\mathbf{o}}^{(h)}(x) = \frac{1}{H-1}\sum_{h'\neq h}\mathbf{o}^{(h')}(x). \tag{2}$$

We then update the target head output by setting $\mathbf{o}^{(h)}(x) \leftarrow \tilde{\mathbf{o}}^{(h)}(x)$ during the forward pass. Compared with hard zero-masking, mean-replacement reduces out-of-distribution artifacts.

Head Importance Score. A head's importance score is computed by the expected degradation in the task utility over the dataset $\mathcal{D}_t^{\text{disc}}$:

$$S_t(h) = \mathbb{E}_{(x,y)\sim\mathcal{D}_t^{\text{disc}}}\Big[\Delta\mathcal{P}_t\big((x,y);\mathcal{I}_h\big)\Big]. \tag{3}$$

Here $\Delta\mathcal{P}_t\big((x,y);\mathcal{I}_h\big) = \mathcal{P}_t(x,y;\theta) - \mathcal{P}_t\big(x,y;\theta_{\mathcal{I}_h}\big)$, where $\theta_{\mathcal{I}_h}$ denotes the model under intervention in Eq. (2). We select the Top-K_h heads sorted by $S_t(h)$ as $\mathcal{H}_t^*$.

4.2 Fine-grained Localization: Head-Guided Neuron Attribution

Motivation and Formulation. After localizing the critical routing nodes (the critical head set $\mathcal{H}_t^*$), we trace back and identify the FFN neurons that provide causally relevant information along these pathways. A common heuristic ranks neurons by activation magnitudes (Huang et al., 2024; Xu et al., 2025), but highly activated neurons can be polysemantic, inhibitory, or write information that is subsequently filtered out by attention routing, resulting in many false positives and unstable task-critical sets. Our intuition follows the *key–value memory* view of Transformer FFNs (Geva et al., 2021): once triggered, a neuron writes a content-bearing update into the residual stream via its down-projection direction. Formally, given $\mathcal{H}_t^*$ and the neuron full set $\mathcal{U}$, we aim to produce a globally ranked list and select the task-critical neuron set $\mathcal{N}_t^*$.

Our key idea is to first compute a sample-level contribution score $c_{\ell,i}(x,y;\theta)$ of a neuron (ℓ,i) given a sample $(x,y)\in\mathcal{D}_t^{\text{disc}}$, then apply head-guided interventions using $h\in\mathcal{H}_t^*$ to quantify an overall importance score $I_{\ell,i}$ over all instances in

$\mathcal{D}_t^{\text{disc}}$, and lastly select the top K neurons by $I_{\ell,i}$ to form the task-critical neuron set $\mathcal{N}_t^*$. We detail these three steps below.

Sample-level Write-in Contribution $c_{\ell,i}(x, y; \theta)$. The goal here is to quantify the contribution of neuron (ℓ, i) in the model θ to the prediction of y given x. With the notations in §3, we consider an FFN neuron $(\ell, i) \in \mathcal{U}$ and obtain its scalar activation $z_i^{(\ell)}$ by a forward pass of the last token in x using the model θ. By downprojection in the second FFN, we compute the neuron's output vector written into the subsequent residual stream as:

$$\Delta \mathbf{r}_i^{(\ell)} = z_i^{(\ell)} \mathbf{W}_{\text{down}}^{(\ell)}[i, :] \in \mathbb{R}^d. \tag{4}$$

To quantify the write-in contribution of this residual increase $\Delta \mathbf{r}_i^{(\ell)}$ to the next-token prediction, we apply the *direct vocabulary projection* (DVP), which reuses the LM's prediction head to project the residual stream to the vocabulary space $\mathcal{V}$. Specifically, let $\mathbf{U} \in \mathbb{R}^{|\mathcal{V}| \times d}$ be the unembedding matrix in the LM's prediction head, where $\mathbf{u}_v = \mathbf{U}[v, :]$ is the unembedding vector of a token $v \in \mathcal{V}$. We then compute the neuron's contribution to affect the probability of predicting v by:

$$c_{\ell,i}(x, v; \theta) = \langle \Delta \mathbf{r}_i^{(\ell)}, \mathbf{u}_v \rangle \tag{5}$$

Since the ground-truth target y may contain a varying number of tokens, we need to aggregate its tokens to compute the unembedding vector $\mathbf{u}_y$ of y before computing $c_{\ell,i}(x, y; \theta)$. To this end, we propose to compute $\mathbf{u}_y$ for two categories of y.

1. *Fixed-set Targets* in tasks like VQA, OCR and retrieval where y is restricted to a fixed set of possible outputs. In this case, we compute the target vector $\mathbf{u}_y$ as the normalized mean unembedding vectors for all the tokens in y, i.e., $\mathbf{u}_y = \frac{\sum_{v \in y} \mathbf{u}_v}{\left\| \sum_{v \in y} \mathbf{u}_v \right\|_2}$.

2. *Open-ended Targets* in tasks like captioning where y can be any valid sentences. As not all tokens in y are equally important to estimate $\mathbf{u}_y$, we estimate the inverse document frequency (IDF) $\alpha(v)$ of each token v in y using the instances in $\mathcal{D}_t^{\text{disc}}$; see the IDF measure in Appendix D.1. We compute an IDF-weighted semantic center by:

$$\mathbf{u}_y = \frac{\sum_{v \in y} \alpha(v)\, \mathbf{u}_v}{\left\| \sum_{v \in y} \alpha(v)\, \mathbf{u}_v \right\|_2}. \tag{6}$$

Similar to Eq. (5), we can compute the sample-level write-in contribution as:

$$c_{\ell,i}(x, y; \theta) = \langle \Delta \mathbf{r}_i^{(\ell)}, \mathbf{u}_y \rangle. \tag{7}$$

Head-guided Importance Score $I_{\ell,i}$. To condition attribution on the critical routing pathways, for each critical head $h \in \mathcal{H}_t^*$ we apply a head intervention $\mathcal{I}_h$ (§4.1) during inference and recompute the neuron's contribution $c_{\ell,i}(x, y; \theta_{\mathcal{I}_h})$ using the intervened model $\theta_{\mathcal{I}_h}$. We measure the contribution drop due to head-guided intervention with respect to the original model θ as

$$\Delta c_{\ell,i}^{(h)}(x, y) = [c_{\ell,i}(x, y; \theta) - c_{\ell,i}(x, y; \theta_{\mathcal{I}_h})]_+,$$

where $[u]_+ = \max(u, 0)$ ensures to focus on neurons whose task-specific contribution drops when critical head routes are disrupted. We then aggregate the contribution drop across heads and data by a head-importance–weighted average:

$$\begin{aligned} I_{\ell,i} &= \sum_{h \in \mathcal{H}_t^*} w_h\, \mathbb{E}_{(x,y) \sim \mathcal{D}_t^{\text{disc}}} \big[\Delta c_{\ell,i}^{(h)}(x, y) \big], \\ w_h &= \frac{S_t(h)}{\sum_{h' \in \mathcal{H}_t^*} S_t(h')}, \qquad \sum_{h \in \mathcal{H}_t^*} w_h = 1, \end{aligned} \tag{8}$$

where $S_t(h)$ is the head importance in Eq. (3).

Final Ranking. We compute $I_{\ell,i}$ in Eq. (8) on the discovery split $\mathcal{D}_t^{\text{disc}}$ for all neurons $(\ell, i) \in \mathcal{U}$, rank them globally, and select the Top-K neurons to form the task-critical set:

$$\mathcal{N}_t^* = \text{TopK}\big(I_{\ell,i} \;:\; (\ell, i) \in \mathcal{U} \big). \tag{9}$$

4.3 Inference-time Steering

Neuron Mask and Scaling. To *verify* whether the neurons in $\mathcal{N}_t^*$ are causally responsible for task t, we learn to steer all the detected neurons in $\mathcal{N}_t^*$ using the validation split $\mathcal{D}_t^{\text{dev}}$, while keeping all backbone weights θ frozen. Specifically, we introduce a set of learnable scaling factors $\lambda_t = \{\lambda_{\ell,i} \mid (\ell, i) \in \mathcal{N}_t^*\}$ and steer the model by scaling only the detected task-critical neurons, i.e., $\tilde{z}_i^{(\ell)} = z_i^{(\ell)} \times \lambda_{\ell,i}, \forall (\ell, i) \in \mathcal{N}_t^*$. We denote the resulting neuron-scaled model for task t as θ_{λ_t}.

Task-specific Lightweight Learning. We learn λ_t *separately for each task* on $\mathcal{D}_t^{\text{dev}}$, while keeping all backbone weights in θ frozen. Let $p_\theta(\cdot|x)$ and $p_{\theta_{\lambda_t}}(\cdot|x)$ be the next-token distributions computed using the original model and the neuron-scaled model. We seek the optimal λ_t by minimizing the task-specific loss and the KL term that regularizes the scaled model to stay close to the original model.

$$\begin{aligned} \min_{\lambda_t} \; \mathbb{E}_{(x,y) \in \mathcal{D}_t^{\text{dev}}} \Big[& \mathcal{L}_t\big(x, y; \theta_{\lambda_t}\big) \\ & + \beta\, \text{KL}\Big(p_\theta(\cdot|x) \,\|\, p_{\theta_{\lambda_t}}(\cdot|x) \Big) \Big]. \end{aligned} \tag{10}$$

Strategy	LLaVA-1.5-7B					Qwen2.5-VL-7B				
	VQA	OCR	Caption	Retrieval	Avg	VQA	OCR	Caption	Retrieval	Avg
AP (Huang et al., 2024)	11.33	10.40	8.65	0.50	7.72	5.00	10.40	14.51	0.19	7.53
MA (Xu et al., 2025)	6.82	15.50	11.90	1.35	8.89	22.05	15.50	9.86	0.65	12.02
APE (Huo et al., 2024)	3.20	-1.87	12.20	0.90	3.61	-2.36	-1.87	6.20	0.08	0.51
HONES-NoHead	10.20	4.50	6.20	0.40	5.33	11.00	8.75	10.50	2.60	8.21
HONES-RandHead	9.70	6.30	10.00	1.50	6.88	13.00	11.50	18.00	3.75	11.56
HONES-Gaussian	4.50	3.60	9.45	0.95	4.63	5.50	9.00	12.50	0.80	6.95
RandNeuron	0.85	1.62	2.30	0.06	1.21	2.45	1.98	3.30	0.15	1.97
HONES (Ours)	**27.30**	**19.00**	**19.80**	**7.43**	**18.38**	**36.50**	**21.00**	**24.80**	**5.35**	**21.91**

Table 1: Relative performance drop (%) after masking the top-1% neurons selected by each ranking method. Higher values indicate greater causal importance (negative values indicate improvement).

Method	LLaVA-1.5-7B	Qwen2.5-VL-7B
DLA (Elhage et al., 2021)	7.40	9.80
Group Patching (Zhang and Nanda, 2024)	18.00	20.05
QRNCA (Chen et al., 2025)	20.80	24.50
LLM-Knowledge (Yu and Ananiadou, 2024)	16.50	19.40
HONES (Ours)	**27.30**	**36.50**

Table 2: Comparison of neuron localization baselines on VQA, measured by the relative performance drop (%) after masking the top-1% neurons identified by each method. Larger drops indicate that the selected neurons are more causally important for task performance.

5 Experiments

5.1 Experimental Setting

Tasks and Data. We focus on four popular vision-language tasks: VQA, OCR, image captioning (Caption), and image-to-text retrieval (Retrieval). To ensure comparability across tasks, we construct a unified multi-task benchmark based on the `train2014` split of **MS COCO** (Lin et al., 2014). Specifically, we identify the shared images from **COCO-Text** (Veit et al., 2016), **MSCOCO Captions** (Chen et al., 2015), and **VQAv2** (Goyal et al., 2017), and curate a subset of 12,000 images, each of which contains human annotations for all tasks from the corresponding source benchmarks. Following the standard practice of prior work (Huo et al., 2024), we split the data into 7K for neuron analysis, 2K for model-steering parameter tuning, and 3K for testing (see details in Appendix A).

Base VLMs and Baselines. We employ two widely used VLMs as our base models: **LLaVA-1.5-7B** (Liu et al., 2023) and **Qwen2.5-VL-7B** (Bai et al., 2025). To evaluate the effectiveness of HONES in identifying key neurons, we compare our method against three state-of-the-art activation-based neuron ranking methods across all four tasks: (1) activation probability **(AP)** (Huang et al., 2024), (2) mean activation **(MA)** (Xu et al., 2025), and (3) activation probability entropy **(APE)** (Huo et al., 2024). These methods are naturally comparable across tasks because they rely only on activation statistics and do not require task-specific supervision signals or readout-specific surrogate objectives. In addition to activation-statistic rankings, we also compare HONES against other popularly-used neuron ranking strategies, ranging from logit attribution to causal tracing to gradient-based attribution. We conduct these comparisons exclusively on VQA, as it provides naturally defined token-level supervision. Extending the same setup to tasks such as OCR, captioning, and retrieval would require task-specific aggregation or surrogate objectives, thereby reducing cross-task comparability. Our additional baselines include (4) direct logit attribution **(DLA)** (Elhage et al., 2021), (5) activation patching with neuron groups **(Group Patching)** (Zhang and Nanda, 2024), (6) **QRNCA** (Chen et al., 2025), a gradient-based neuron ranking method, and (7) neuron-level knowledge attribution **(LLM-Knowledge)** (Yu and Ananiadou, 2024). To further assess the benefits of our neuron-level intervention strategy, we consider four control variants: random neuron masking **(RandNeuron)**, an ablation variant without head conditioning **(HONES–NoHead)**, HONES with random attention head masking **(HONES-RandHead)**, and HONES with Gaussian noise injection **(HONES-Gaussian)**.

Regarding model steering, we assess effectiveness and OOD generalizability by comparing HONES against matched-budget baselines: (1) fixed uniform two-fold amplification on HONES-identified neurons **(Fixed-Amp)**, (2) grid-searched and uniform amplification on these neurons **(Grid-Search)**, (3) learnable scaling on random neurons

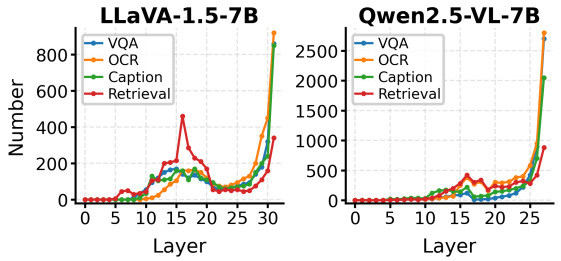


Figure 2: Layer-wise distribution of the top-1% task-critical neurons across four tasks (VQA/OCR/Caption/Retrieval) for both models.

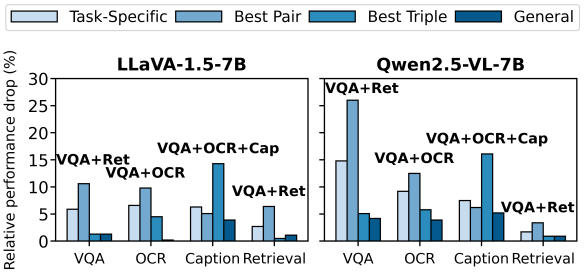


Figure 3: Cross-task neuron-group ablation results. Bars denote the relative performance drop (%) on each target task (x-axis) when masking the corresponding neuron group (legend): *task-specific*, the most damaging two-task overlap (*Best Pair*), the most damaging three-task overlap (*Best Triple*), and the four-task shared *General* group. Text labels indicate the identities of *Best Pair*/*Best Triple* for each target task.

(RandNeuron), and (4) HONES steering without KL regularization **(HONES w/o KL)**. We also tested LoRA but found it ineffective under our low-budget setting.

Metrics. We measure model performance on each task using standard metrics: accuracy for VQA, average normalized levenshtein similarity (ANLS) for OCR (Biten et al., 2019), BLEU-4 for Caption (Papineni et al., 2002), and NDCG@5 for Retrieval (Järvelin and Kekäläinen, 2002). Additional implementation details and prompt templates are provided in Appendix A.

5.2 Key Findings in Neuron Analysis

Superior localization of key neurons with HONES compared to activation-based methods. Table 1 and Table 2 show the relative performance drop induced by masking top-1% neurons selected by each interpretation method across four tasks on two base VLMs. Based on empirical validation (see Appendix D.2), HONES identifies key neurons by ranking them using the top 30 attention heads in LLaVA and the top 25 attention heads in Qwen, respectively. The results show that our method consistently outperforms all baselines in identifying critical neurons on both base models, inducing an average performance drop of 18.38% on LLaVA and 21.91% on Qwen. Notably, masking neurons ranked by APE leads to performance improvements on the VQA and OCR tasks, suggesting that entropy-based neuron ranking may capture task-irrelevant or interfering neurons. Further comparison of neuron intervention strategies shows that masking top-ranked attention heads is substantially more effective for identifying critical neurons across tasks—yielding over a 10% greater performance drop—than masking randomly ranked heads or applying Gaussian noise injection. Our results indicate that neuron importance is better captured by causally aligned information flow to the model's readout than by raw activation magnitude, which is often confounded by polysemantic activity. We also assess HONES' performance on a larger backbone, **LLaVA-1.5-13B**, under the same four-task setting. The available results show trends consistent with those of the 7B models, with HONES remaining the strongest localization method. Full results are provided in Appendix D.5. Beyond effectiveness, we further assess HONES in terms of localization efficiency on VQA against two representative baselines that are most relevant to our method, namely the gradient-based attribution method QRNCA and the intervention-based Group Patching. Our results show that HONES, as a gradient-free method, localizes key neurons substantially faster than its two counterparts (see Appendix D.6).

Key neurons exhibit task-dependent layer preferences. Inspired that VLMs perform different stages of multimodal information processing across layers, we further explore the layer-wise distribution of our identified top-1% neurons per task, aiming to examine the processing stages at which these neurons emerge and their relationship to task characteristics. Specifically, we aggregate the number of identified neurons within each layer. This layer-wise count allows us to pinpoint the specific network depths where task-critical computations are most concentrated. Fig. 2 displays the results for each base model. Overall, we observe that key neurons are predominantly concentrated in the middle layers (10–20) and deeper layers (>25). In contrast to the retrieval task, which tends to exhibit a peak in the middle layers, the other three tasks exhibit higher concentrations in deeper layers for both

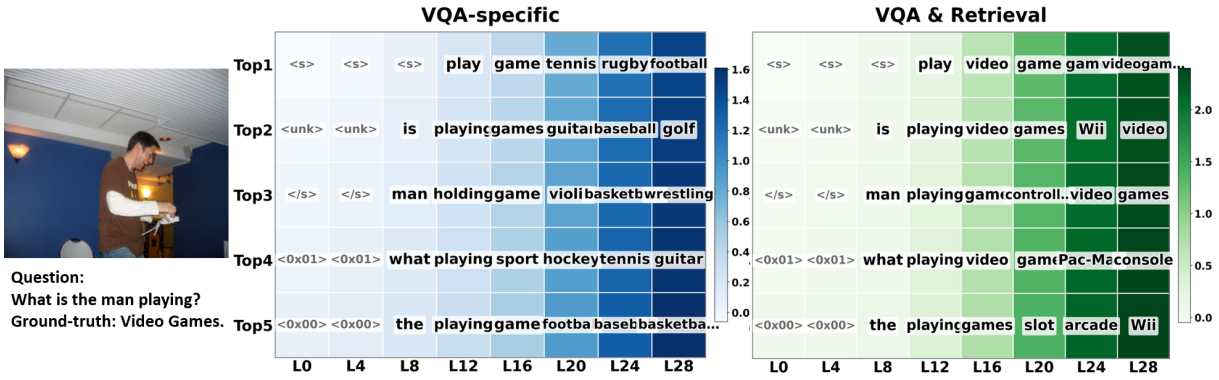


Figure 4: VQA Logit Lens case study in LLaVA-1.5. Rows show Top-5 tokens and columns are sampled every 4 layers; color indicates Δlogit (baseline−masked) for the task-specific group and the VQA&Retrieval shared group.

Model	Method	VQA	OCR	Caption	Retrieval	Avg
LLaVA-1.5	Base	0.6520	0.5760	0.1285	0.9328	0.5723
	Fixed-Amp	0.6610	0.5820	0.1290	0.9370	0.5773
	Grid-Search	0.6660	0.5944	0.1315	0.9560	0.5870
	RandNeuron	0.6530	0.5735	0.1260	0.9345	0.5718
	Ours w/o KL	0.6580	0.5790	0.1385	0.9365	0.5780
	Ours	**0.6733**	**0.6019**	**0.1407**	**0.9626**	**0.5946**
Qwen2.5-VL	Base	0.6750	0.6580	0.2240	0.9464	0.6259
	Fixed-Amp	0.6773	0.6620	0.2280	0.9492	0.6291
	Grid-Search	0.6803	0.6650	0.2290	0.9500	0.6311
	RandNeuron	0.6760	0.6560	0.2210	0.9480	0.6253
	Ours w/o KL	0.6790	0.6610	0.2250	0.9475	0.6281
	Ours	**0.6907**	**0.6790**	**0.2335**	**0.9580**	**0.6403**

Table 3: Effectiveness of neuron steering. Fixed-Amp and Grid-Search are train-free amplifications; RandNeuron learns scaling on a random neuron set; Ours w/o KL removes KL regularization; **Ours** learns sparse scaling with KL. Metrics: Acc. (VQA), ANLS (OCR), BLEU-4 (Caption), NDCG@5 (Retrieval).

VLMs. Given that middle layers are generally associated with visual–text alignment, whereas deeper layers are more involved in answer decoding (Yu and Lee, 2025), these results suggest that retrieval relies more heavily on visual–text alignment, while text-based answer generation plays a larger role in the remaining three generation-oriented tasks.

VQA-shared neurons dominate cross-task saliency. Examining key neurons across tasks, we find that a substantial number of these neurons are shared among multiple tasks. This motivates us to explore how the influence of these multi-task neurons compares to task-specific neurons across tasks, providing insight into the mechanisms underlying cross-task generalization and model sparsity.

Given the top-1% neurons identified by HONES per task, we group neurons based on the tasks across which they are shared. For each target task, we then calculate the relative performance drop resulting from masking each neuron group. Fig. 3 presents the results for both base models. To conserve space, we only show the shared group that induces the largest drop when neurons are shared by two and three tasks, respectively (see full comparisons in Appendix E.1). Interestingly, multi-task shared neurons exhibit higher saliency than task-specific ones, particularly those overlapping with VQA, suggesting a hub effect whereby VQA-related neurons support a broad range of vision–language tasks. Pairwise shared neurons demonstrate dominant causal saliency across tasks, with the exception of captioning. We hypothesize that this deviation arises because captioning relies on holistic image understanding, while textual cues present in our curated images may bias the most salient neuron group toward OCR-related neurons. To verify our hypothesis, we repeat the analysis on the same images after removing textual cues. Following Appendix E.2, the most salient neuron group for the captioning task shifts back to the VQA–Caption pair. We further examine patterns of key-neuron sharing across tasks on out-of-distribution datasets and observe the same qualitative trend, suggesting that the identified routing-and-neuron structures generalize across diverse data distributions (Appendix E.3).

Furthermore, token-level Logit Lens analysis (Neo et al., 2025) provides semantic corroboration. Specifically, we quantify the causal effect of a neuron group by the logit drop on target tokens, $\Delta\text{logit} = \text{logit}_{\text{base}} - \text{logit}_{\text{masked}}$. We compute this layer-wise by projecting the MLP layer outputs to the vocabulary space. By comparing the projected logits before and after masking the target neuron group, we inspect which candidate tokens lose the most support under intervention. As illustrated in Fig. 4, VQA-specific neurons mainly strengthen coarse answer priors

Model	Method	GQA	TextVQA	$\text{Flickr}_{\text{Cap}}$	$\text{Flickr}_{\text{Ret}}$	Avg
LLaVA-1.5	Base	0.610±0.001	0.470±0.001	0.097±0.001	0.863±0.002	0.510±0.001
	Ours (zero-shot)	0.616±0.001	0.474±0.002	0.099±0.002	0.870±0.001	0.515±0.001
	RandNeuron	0.610±0.001	0.469±0.002	0.097±0.001	0.864±0.002	0.510±0.001
	Ours (tuned)	**0.628±0.002**	**0.488±0.003**	**0.106±0.002**	**0.883±0.002**	**0.526±0.001**
Qwen2.5-VL	Base	0.632±0.001	0.596±0.002	0.163±0.002	0.888±0.002	0.570±0.001
	Ours (zero-shot)	0.636±0.002	0.601±0.001	0.165±0.002	0.892±0.002	0.573±0.001
	RandNeuron	0.632±0.001	0.596±0.002	0.163±0.001	0.889±0.001	0.570±0.001
	Ours (tuned)	**0.641±0.002**	**0.607±0.002**	**0.169±0.002**	**0.898±0.002**	**0.579±0.001**

Table 4: OOD steering results (Accuracy on GQA, ANLS on TextVQA, BLEU-4 on FlickrCap, and NDCG@5 on FlickrRet). **Ours (tuned)** tunes scalars on the identified task-critical neurons; *RandNeuron* tunes scalars on randomly selected neurons with a matched budget; *Ours (zero-shot)* directly transfers in-domain scales.

(e.g., generic category-level candidates), whereas the VQA&Retrieval shared group selectively amplifies fine-grained, visually grounded tokens, effectively shifting predictions toward the ground truth. We observe consistent trends in OCR, captioning, and retrieval (Appendix E.4).

5.3 Results of Neuron-level Model Steering

Effectiveness. Table 3 displays the performance of HONES in neuron-level model steering compared against a variety of baselines. To improve steering efficiency, we restrict steering to the dominant salient neuron group per task identified in Sec. 5.2, rather than intervening on the entire set of top-1% neurons. Overall, HONES consistently outperforms all baselines across tasks on both backbone models. The gap to **RandNeuron** provides interventional evidence that task capability is concentrated in a small subset of neurons, while the gain over **Grid-Search** indicates that using neuron-wise modulation, rather than a single uniform amplifier, is necessary to fully exploit these causal pathways in practice.

OOD Generalization. To test the generalizability of our neuron steering method, we further evaluate the steered models on three out-of-distribution benchmarks for four tasks: GQA (VQA), TextVQA (OCR), and Flickr30k (Caption & Retrieval). Similarly, we focus here exclusively on neurons belonging to the dominant salient group for each task. As shown in Table 4, directly transferring the in-domain learned scaling factors (*Ours (zero-shot)*) already yields consistent gains. Moreover, without re-localizing attention heads or re-identifying neurons, and using only 20% OOD samples to learn scaling with evaluation on the remaining 80% (averaged over 10 random splits; mean±std), our method consistently outperforms the random-neuron control. This indicates that task capabilities are indeed concentrated and anchored in those sparse neurons from the dominant salient neuron group. Additional settings and statistical details are provided in Appendix F.3.

6 Conclusion

We present **HONES**, a head-oriented, readout-aligned causal interpretability framework for multi-task VLMs. By localizing task-critical routing heads and ranking MLP neurons via Causal Write-in Effect, HONES provides a unified and verifiable notion of neuron importance across heterogeneous task readouts. Through a systematic analysis of LLaVA-1.5-7B and Qwen2.5-VL-7B, we find that activation statistics are often decoupled from causal criticality, task-critical computation follows a stable rise–valley–surge depth profile with task-dependent anchoring, and cross-task behaviors are mediated by a sparse VQA-centric *Hub-and-Bridge* sharing structure. Our learning-based steering further operationalizes these discoveries by learning only a small set of neuron-wise scaling factors, enabling effective and stable control of task behaviors while keeping all backbone weights frozen.

Limitations

Despite providing new insights into interpretability and controllable interventions for large VLMs, our study has several limitations.

1. **Model scale and architectural coverage.** To enable exhaustive layer-wise and neuron-level scanning, we focus on 7B dense backbones (LLaVA-1.5-7B and Qwen2.5-VL-7B). While both exhibit a consistent *VQA-centric hub* pattern, it remains unclear whether this structure generalizes to substantially larger models (e.g., 70B+) or mixture-of-experts (MoE) architectures. As capacity increases, functional modularization may emerge differently or

routing may become more distributed, which we leave for future work with more substantial compute budgets.

2. **Task definition granularity.** For a unified cross-task analysis, we study four coarse task categories (VQA, OCR, Caption, Retrieval). This abstraction may obscure finer-grained variations within each category. For instance, VQA covers diverse sub-skills such as counting, spatial reasoning, and commonsense QA, which may rely on richer and more distinct neuron-level mechanisms. Future work can adopt a finer task taxonomy to reveal sub-task-level sharing and conflict patterns.

3. **Computational cost of causal analysis.** Our coarse-to-fine pipeline requires multiple forward passes, including targeted head ablations and verification through neuron-group and cross-task ablations. Scaling to more layers, more neurons, or larger datasets can therefore incur substantial computational overhead. Future work may explore more efficient candidate screening and approximate evaluation strategies to reduce inference-time cost.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L. Menick, Sebastian Borgeaud, and 8 others. 2022. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736.

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, and 8 others. 2025. Qwen2.5-vl technical report. *Preprint*, arXiv:2502.13923.

Samyadeep Basu, Martin Grayson, Cecily Morrison, Besmira Nushi, Soheil Feizi, and Daniela Massiceti. 2024. Understanding information storage and transfer in multi-modal large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pages 7400–7426.

Ali Furkan Biten, Ruben Tito, Andres Mafla, Lluis Gomez, Marcal Rusinol, Ernest Valveny, C. V. Jawahar, and Dimosthenis Karatzas. 2019. Scene text visual question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4291–4301.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nicholas L. Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Alex Tamkin, Karina Nguyen, Brayden McLean, and 5 others. 2023. Towards monosemanticity: Decomposing language models with dictionary learning. Transformer Circuits (Anthropic).

Lihu Chen, Adam Dejl, and Francesca Toni. 2025. Identifying query-relevant neurons in large language models for long-form texts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 23595–23604.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. 2015. Microsoft COCO captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. 2024. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24185–24198.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502.

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. In *Advances in Neural Information Processing Systems*, volume 36, pages 49250–49267.

Yunkai Dang, Kaichen Huang, Jiahao Huo, Yibo Yan, Sirui Huang, Dongrui Liu, Mengxi Gao, Jie Zhang, Chen Qian, Kun Wang, Yong Liu, Jing Shao, Hui Xiong, and Xuming Hu. 2024. Explainable and interpretable multimodal large language models: A comprehensive survey. *CoRR*, abs/2412.02104.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, and 6 others. 2021. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 1(1):12.

Junfeng Fang, Zongze Bi, Ruipeng Wang, Houcheng Jiang, Yuan Gao, Kun Wang, An Zhang, Jie Shi, Xiang Wang, and Tat-Seng Chua. 2024. Towards neuron attributions in multi-modal large language models. In *Advances in Neural Information Processing Systems*, volume 37, pages 122867–122890. Curran Associates, Inc.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6325–6334.

Muhammad Umair Haider, Hammad Rizwan, Hassan Sajjad, Peizhong Ju, and A. B. Siddique. 2025. Neurons speak in ranges: Breaking free from discrete neuronal attribution. In *Mechanistic Interpretability Workshop at NeurIPS 2025*.

Anwen Hu, Haiyang Xu, Liang Zhang, Jiabo Ye, Ming Yan, Ji Zhang, Qin Jin, Fei Huang, and Jingren Zhou. 2025. mPLUG-DocOwl2: High-resolution compressing for OCR-free multi-page document understanding. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5817–5834, Vienna, Austria. Association for Computational Linguistics.

Kaichen Huang, Jiahao Huo, Yibo Yan, Kun Wang, Yutao Yue, and Xuming Hu. 2024. MINER: Mining the underlying pattern of modality-specific neurons in multimodal large language models. *arXiv preprint arXiv:2410.04819*.

Drew A. Hudson and Christopher D. Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6700–6709.

Jiahao Huo, Yibo Yan, Boren Hu, Yutao Yue, and Xuming Hu. 2024. MMNeuron: Discovering neuron-level domain-specific interpretation in multimodal large language model. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6801–6816, Miami, Florida, USA. Association for Computational Linguistics.

Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 19730–19742. PMLR.

Hyesu Lim, Jinho Choi, Jaegul Choo, and Steffen Schneider. 2025. Sparse autoencoders reveal selective remapping of visual concepts during adaptation. In *International Conference on Learning Representations (ICLR)*. Poster.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, pages 740–755.

Zihao Lin, Samyadeep Basu, Mohammad Beigi, Varun Manjunatha, Ryan A. Rossi, Zichao Wang, Yufan Zhou, Sriram Balasubramanian, Arman Zarei, Keivan Rezaei, Ying Shen, Barry Menglong Yao, Zhiyang Xu, Qin Liu, Yuxiang Zhang, Yan Sun, Shilong Liu, Li Shen, Hongxuan Li, and 2 others. 2025. A survey on mechanistic interpretability for multi-modal foundation models. *arXiv preprint arXiv:2502.17516*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pages 34892–34916.

Zhihang Liu, Chen-Wei Xie, Bin Wen, Feiwu Yu, Jixuan Chen, Pandeng Li, Boqiang Zhang, Nianzu Yang, Yinglu Li, Zuan Gao, Yun Zheng, and Hongtao Xie. 2025. Capability: A comprehensive visual caption benchmark for evaluating both correctness and thoroughness. *arXiv preprint arXiv:2502.14914*.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. In *Advances in Neural Information Processing Systems*, pages 17359–17372.

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 (ICLR)*.

Tuomas Oikarinen and Tsui-Wei Weng. 2024. Linear explanations for individual neurons. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 38639–38662. PMLR.

Mateusz Pach, Shyamgopal Karthik, Quentin Bouniot, Serge Belongie, and Zeynep Akata. 2025. Sparse autoencoders learn monosemantic features in vision-language models. *arXiv preprint arXiv:2504.02821*.

Haowen Pan, Yixin Cao, Xiaozhi Wang, Xun Yang, and Meng Wang. 2024. Finding and editing multi-modal neurons in pre-trained transformers. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1012–1037, Bangkok, Thailand. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318.

Hassan Sajjad, Nadir Durrani, and Fahim Dalvi. 2022. Neuron-level interpretation of deep NLP models: A survey. *Transactions of the Association for Computational Linguistics*, 10:1285–1303.

Sarah Schwettmann, Neil Chowdhury, Samuel Klein, David Bau, and Antonio Torralba. 2023. Multimodal neurons in pretrained text-only transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 2862–2867.

Dong Shu, Xuansheng Wu, Haiyan Zhao, Daking Rai, Ziyu Yao, Ninghao Liu, and Mengnan Du. 2025. A survey on sparse autoencoders: Interpreting the internal mechanisms of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 1690–1712, Suzhou, China. Association for Computational Linguistics.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. 2019. Towards vqa models that can read. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8309–8318.

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Xin Zhao, Furu Wei, and Ji-Rong Wen. 2024. Language-specific neurons: The key to multilingual capabilities in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5701–5715, Bangkok, Thailand. Association for Computational Linguistics.

Andreas Veit, Tomas Matera, Lukas Neumann, Jiri Matas, and Serge Belongie. 2016. COCO-text: Dataset and benchmark for text detection and recognition in natural images. *arXiv preprint arXiv:1601.07140*.

Feiyu Wang, Ziran Zhao, Dong Yu, and Pengyuan Liu. 2025a. Attribution and application of multiple neurons in multimodal large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 11649–11662.

Qidong Wang, Junjie Hu, and Ming Jiang. 2025b. V-SEAM: visual semantic editing and attention modulating for causal interpretability of vision-language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 17396–17420, Suzhou, China. Association for Computational Linguistics.

Jiaqi Xu, Cuiling Lan, and Yan Lu. 2025. Deciphering functions of neurons in vision-language models. In *Proceedings of the 33rd ACM International Conference on Multimedia*, page 3173–3181.

Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Guohai Xu, Chenliang Li, Junfeng Tian, Qi Qian, Ji Zhang, Qin Jin, Liang He, Xin Lin, and Fei Huang. 2023. UReader: Universal OCR-free visually-situated language understanding with multimodal large language model. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2841–2858. Association for Computational Linguistics.

Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, and Fei Huang. 2024. mplug-owi2: Revolutionizing multi-modal large language model with modality collaboration. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13040–13051.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.

Zeping Yu and Sophia Ananiadou. 2024. Neuron-level knowledge attribution in large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3267–3280, Miami, Florida, USA. Association for Computational Linguistics.

Zhuoran Yu and Yong Jae Lee. 2025. How multimodal llms solve image tasks: A lens on visual grounding, task reasoning, and answer decoding. *Preprint*, arXiv:2508.20279.

Fred Zhang and Neel Nanda. 2024. Towards best practices of activation patching in language models: Metrics and methods. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. 2024. Explainability for large language models: A survey. *ACM Trans. Intell. Syst. Technol.*, 15(2):1–38.

Xiutian Zhao, Rochelle Choenni, Rohit Saxena, and Ivan Titov. 2026. Finding culture-sensitive neurons in vision-language models. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2026 - Volume 1: Long Papers, Rabat, Morocco, March 24-29, 2026*, pages 3366–3381. Association for Computational Linguistics.

A Benchmark Details

In this section, we provide comprehensive details regarding the construction and specific configurations of our unified multi-task benchmark. We first outline the dataset composition and our strict splitting strategy to prevent information leakage. Next, we elaborate on task-specific implementation

Split	#Images	#Instances per Task	Primary Usage
Discovery	7,000	7,000 (VQA/OCR/Caption/Retrieval)	Head localization and neuron identification.
Development	2,000	2,000 (VQA/OCR/Caption/Retrieval)	Hyperparameter selection (e.g., scaling factors).
Test	3,000	3,000 (VQA/OCR/Caption/Retrieval)	Final ablation verification and steering evaluation.
Total	12,000	12,000 (VQA/OCR/Caption/Retrieval)	–

Table 5: Unified benchmark split statistics. Each image yields one instance for each task via prompt switching, hence equal per-task counts in every split.

details, including visual prompting for OCR and candidate mining for retrieval. Finally, we specify the evaluation metrics used to quantify model performance across all tasks.

A.1 Benchmark Composition & Splitting

We base our experiments on the `train2014` split of **MS COCO** (Lin et al., 2014). We first collect candidate images by taking the image overlap among:

- **COCO-Text** (Veit et al., 2016) (scene-text annotations on COCO images, providing text bounding boxes and transcriptions)
- **MSCOCO Captions** (Chen et al., 2015) (human-written image captions; we use the standard 5-reference setting)
- **VQAv2** (Goyal et al., 2017) (a balanced VQA benchmark with open-ended questions and human answers)

We then filter the candidates to retain only images with complete annotations required by our four-task setup, resulting in exactly **12,000** images. Crucially, for each image, we keep the visual input fixed and derive four task inputs (VQA/OCR/Caption/Retrieval) by *only* changing the task instruction prompts. All questions and annotations are directly derived from the official datasets; we do not synthesize new labels.

Following prior neuron-analysis practice (Huo et al., 2024) that separates the data used for neuron identification from held-out evaluation, we strictly partition the 12,000 images into three disjoint splits: **7,000** for *Discovery* (head localization and neuron identification), **2,000** for *Development* (steering hyperparameter selection), and **3,000** for *Test* (final reporting, ablations, and steering evaluation). All splits are image-disjoint to prevent leakage. Table 5 details the statistics and specific usage of each split.

A.2 Task Implementation Details

OCR (Visual Prompting). We leverage the scene-text bounding box annotations provided by **COCO-Text**. Using the annotated box coordinates, we overlay blue bounding boxes onto the original images to indicate the target text region, without cropping or changing image resolution. This visual prompt guides the model to the intended text instance for transcription in multi-text scenes.

Retrieval (Pairwise Re-ranking). We evaluate *image-to-text* retrieval (I→T): given an image query, the model selects the most relevant caption from a candidate pool. Since direct listwise ranking over a large candidate set is challenging for LVLMs, we reformulate I→T retrieval as *pairwise* image–text verification, where the model outputs "Yes"/"No" for each candidate text conditioned on the image, and we rank candidates by the probability of the token "Yes". To balance reliability and computational cost, we use a candidate pool size of 50 per query (1 positive + 49 negatives). For discovery-stage retrieval attribution, we construct candidates within the 7,000-image Discovery split as follows:

- **Query Set:** We use 2,000 images as queries. For each query image, we randomly sample *one* caption from its five ground-truth captions as the positive candidate.
- **Hard Negative Pool:** We collect all captions from the remaining 5,000 images (5 captions per image; 25,000 captions in total) as a global negative pool, ensuring it is disjoint from the query set. Instead of randomly sampling negatives, we rank all pool captions by their *semantic similarity* to the query (measured in a pretrained embedding space with cosine similarity), and select the top-49 most similar captions as negatives. This yields substantially harder and more realistic confounders for retrieval attribution.

A.3 Evaluation Metrics

We adopt standard metrics for each task:

Task	Image Example	Question / Instruction	GT / Target
VQA		What kind of animal is this?	cat
OCR		Transcribe the text inside the blue box. Output text only.	MERRELL
Caption		Describe the image in one sentence.	A cat standing on top of a shoe on the floor.
Retrieval (I→T)		Does the candidate correctly describe the image? (Yes/No) Text: [Candidate 1] A cat standing on top of a shoe. Text: [Candidate 2] A dog running on the grass. . . . Text: [Candidate N] A group of people sitting in the room.	Yes No . . . No

Table 6: Task-specific examples in our unified benchmark. For OCR, the image is overlaid with a blue bounding box (from COCO-Text annotations) to indicate the target text region. Retrieval is evaluated as image-to-text (I→T) pairwise verification and re-ranking.

- **VQA: VQAv2 Accuracy** computed with the official evaluation protocol (answer normalization and consensus-based soft scoring over 10 human answers).
- **OCR: ANLS** (Biten et al., 2019), defined by normalized Levenshtein similarity with a threshold $\tau = 0.5$.
- **Captioning: BLEU-4** (Papineni et al., 2002) using the standard COCO caption evaluation implementation with *multiple references* (5 reference captions per image).
- **Retrieval: NDCG@5** (Järvelin and Kekäläinen, 2002) over the ranked captions for each image query:

$$\mathrm{NDCG@5} = \mathrm{DCG@5}/\mathrm{IDCG@5},$$

$$\mathrm{DCG@5} = \sum_{k=1}^{5} \frac{2^{\mathrm{rel}_k} - 1}{\log_2(k+1)}.$$

where rel_k denotes the relevance of the caption ranked at position k, and IDCG@5 is the DCG@5 of the ideal (perfect) ranking.

A.4 Task-Specific Examples

Table 6 demonstrates representative inputs and targets for each task in our benchmark.

B Prompt Template

B.1 Prompt Templates for Evaluation

To ensure consistent and interpretable input formatting across tasks and models, we design task-specific prompt templates for both **LLaVA-1.5-7B** and **Qwen2.5-VL-7B**. All templates follow the same principle: we keep the visual input fixed and *only* vary the task instruction prompt to induce task-specific readouts. Below, `<Image>` denotes the image input, and `{Question}`/`{CandidateText}` are placeholders filled from the benchmark.

C Task-Critical Attention Heads

In this section, we visualize the spatial distribution of task-critical attention heads to elucidate

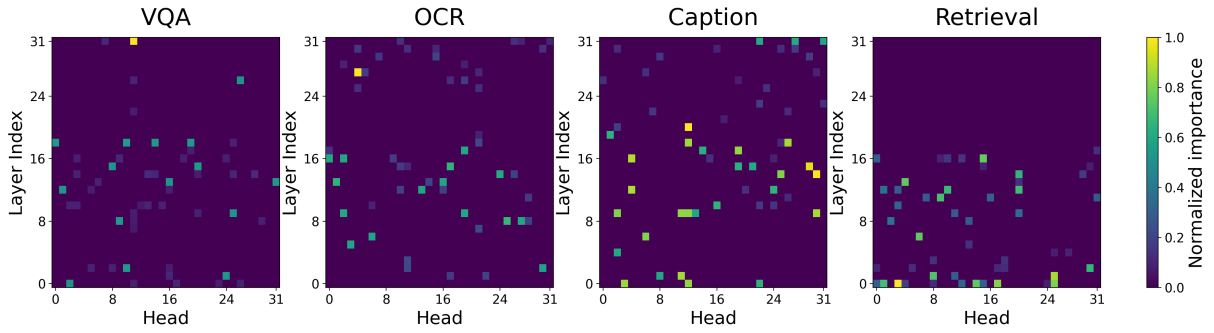


Figure 5: Attention head importance heatmaps for LLaVA-1.5-7B. Rows denote layer indices and columns denote head indices; colors indicate normalized head importance, with brighter colors meaning higher importance.

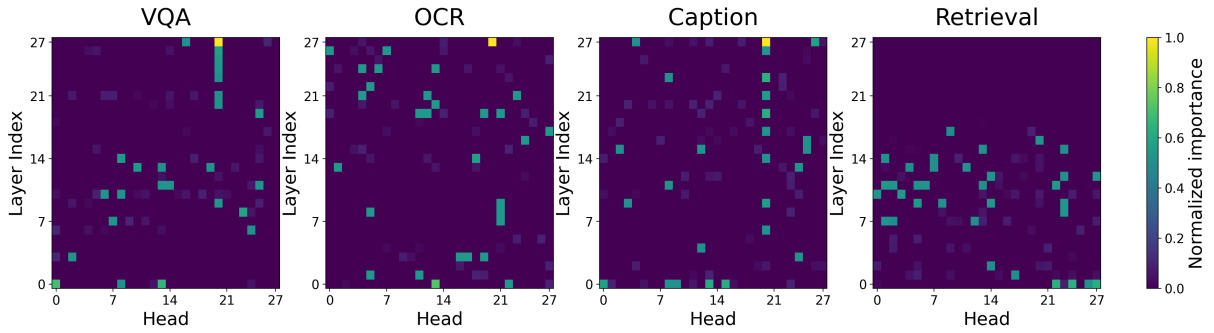


Figure 6: Attention head importance heatmaps for Qwen2.5-VL-7B. Rows denote layer indices and columns denote head indices; colors indicate normalized head importance, with brighter colors meaning higher importance.

the routing mechanisms within VLMs. Using the head-level causal localization procedure detailed in Sec. 4.1 (referencing V-SEAM (Wang et al., 2025b)), we compute the normalized importance of each head across four tasks (VQA, OCR, Captioning, Retrieval) for both LLaVA-1.5-7B and Qwen2.5-VL-7B. The resulting heatmaps are presented in Fig. 5 and Fig. 6.

Across models, we observe three consistent patterns. (i) **Sparsity:** Important heads are highly sparse, indicating that task-relevant information flow is dominated by a small set of routing hubs. (ii) **A shared mid-layer backbone:** Both models exhibit a higher density of critical heads in mid layers, suggesting a shared backbone where cross-modal interactions are most concentrated. (iii) **Task-dependent depth profiles:** Beyond the shared backbone, different tasks exhibit distinct depth patterns: Retrieval tends to emphasize early-to-mid layers, while OCR and Captioning assign more weight to mid-to-late layers; VQA shows a more distributed profile across depth.

These observations motivate the *coarse-to-fine* design of **HONES**: since computation is routed through sparse, task-specific pathways, we condition neuron attribution on the localized routing heads (see §4.2).

D Additional Implementation Details and Results for Neuron Localization

In this section, we provide supplementary details regarding the neuron localization experiments presented in the main text (§5.2), including the hyperparameter selection for attention heads, formal definitions of baseline strategies, and the absolute performance metrics corresponding to the relative drops reported in Table 1, as well as additional larger-backbone validation and efficiency analysis.

D.1 IDF Weight.

We compute $\alpha(\tau)$ on the discovery split. Let N be the number of instances in $\mathcal{D}_t^{\text{disc}}$ and $\mathrm{df}(\tau)$ be the number of instances whose references contain token τ. We use the smoothed IDF:

$$\alpha(\tau) = \log \frac{N+1}{\mathrm{df}(\tau)+1} + 1. \tag{11}$$

Given $\mathbf{u}_{\text{tgt}}(x, y)$, we define the *sample-level* write-in score for run ρ as

$$c_{\ell,i}^{\rho}(x, y) = \left\langle \Delta \mathbf{r}_i^{(\ell),\rho}(x),\, \mathbf{u}_{\text{tgt}}(x, y) \right\rangle. \tag{12}$$

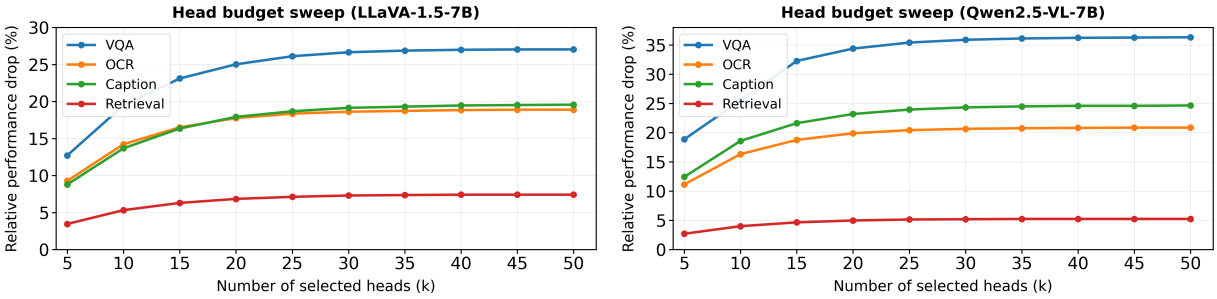


Figure 7: Attention head budget sweep. Relative performance drop (%) after masking the top-K localized heads across four tasks. The curves exhibit clear saturation points, motivating K=30 for LLaVA-1.5-7B and K=25 for Qwen2.5-VL-7B to capture the sparse routing backbone with diminishing returns beyond the elbow.

D.2 Ablation Study on Attention Head Budget

To determine the optimal head budget K_h for constraining the neuron search space in HONES, we conduct a sensitivity sweep over $K_h \in \{5, 10, \ldots, 50\}$ by measuring the relative performance drop induced by masking the top-K_h localized heads for each task. Our selection criterion is based on identifying the saturation point (or "elbow") of the performance curves: a steep initial rise indicates that the top-ranked heads capture sparse routing hubs effectively, whereas a subsequent plateau suggests diminishing returns—including additional heads beyond this point contributes only marginally to task routing and may introduce noise into the downstream neuron attribution.

Fig. 7 summarizes the trends for both models. For **LLaVA-1.5-7B**, the curves increase rapidly and then saturate at task-dependent elbows: VQA saturates around K_h=28, OCR around K_h=26, Caption around K_h=30, and Retrieval around K_h=28. To ensure consistent coverage across tasks while staying near the saturation regime, we adopt a unified budget of $\mathbf{K_h}$**=30** for LLaVA-1.5-7B. For **Qwen2.5-VL-7B**, saturation happens earlier: VQA around K_h=24, OCR around K_h=23, Caption around K_h=25, and Retrieval around K_h=24, thus we set $\mathbf{K_h}$**=25** for Qwen.

Finally, note that masking far more heads (e.g., $K_h \gg 50$) would likely continue to degrade performance and may eventually cause collapse; however, our goal here is *not* to maximize degradation, but to identify a *sparse* and *causally-relevant* routing backbone that stabilizes the neuron search space. Selecting K_h at the plateau balances two factors: (i) capturing the dominant information flow and (ii) avoiding the inclusion of weakly-related heads that could dilute head-constrained neuron attribution.

D.3 Definitions of Baseline Strategies

To rigorously evaluate the efficacy of HONES, we compare it against three widely used activation-based heuristics and a random baseline. Let $a_{ij}^l(x)$ denote the post-activation output (e.g., after GeLU) of neuron j at layer l on token position i for input x. Since VLM inputs contain multiple tokens (visual and textual), we first aggregate activations over positions to obtain a single scalar per neuron:

$$\tilde{a}_j^l(x) \;=\; \max_{i \in \mathcal{I}(x)} a_{ij}^l(x),$$

where $\mathcal{I}(x)$ denotes the set of token positions for x. All statistics below are computed on the same discovery split used for neuron selection, and we always select the top-B neurons under each strategy with an identical budget B (Top-1% of all MLP neurons).

Activation Probability (AP; "activation frequency"). This heuristic ranks neurons by how often they are activated across the dataset, a common proxy used in modality-/domain-sensitive neuron discovery. Formally, given a task-specific dataset $\mathcal{D}$ and an activation threshold τ (we set $\tau = 0$), we define

$$p_j^l \;=\; \frac{1}{|\mathcal{D}|} \sum_{x \in \mathcal{D}} \mathbb{I}\Big[\tilde{a}_j^l(x) > \tau\Big] .$$

Neurons are then ranked by p_j^l in descending order, and the top-B are selected. This baseline is aligned with activation-probability-based neuron mining in multimodal settings (e.g., Huang et al., 2024).

Mean Activation Magnitude (MA; "mean activation rank"). Instead of frequency, this baseline ranks neurons by their average activation strength, motivated by the intuition that consistently large

Strategy	LLaVA-1.5-7B					Qwen2.5-VL-7B				
	VQA	OCR	Caption	Retrieval	Avg	VQA	OCR	Caption	Retrieval	Avg
Base VLM (unmasked)	0.6520	0.5760	0.1285	0.9328	0.5723	0.6750	0.6580	0.2240	0.9464	0.6259
AP (Huang et al., 2024)	0.5780	0.5161	0.1174	0.9281	0.5349	0.6410	0.5896	0.1915	0.9446	0.5917
MA (Xu et al., 2025)	0.6073	0.4867	0.1132	0.9202	0.5319	0.5260	0.5560	0.2019	0.9402	0.5561
APE (Huo et al., 2024)	0.6310	0.5868	0.1128	0.9244	0.5638	0.6907	0.6703	0.2101	0.9456	0.6292
HONES-RandHead	0.5887	0.5397	0.1157	0.9188	0.5407	0.5873	0.5823	0.1837	0.9109	0.5660
HONES-Gaussian	0.6227	0.5553	0.1164	0.9239	0.5546	0.6377	0.5988	0.1960	0.9388	0.5929
RandNeuron	0.6463	0.5667	0.1255	0.9322	0.5677	0.6587	0.6450	0.2166	0.9450	0.6163
HONES (Ours)	0.4740	0.4666	0.1031	0.8635	0.4768	0.4287	0.5198	0.1684	0.8958	0.5032

Table 7: Absolute performance after masking top-1% neurons. Avg is the arithmetic mean of the four task scores.

Method	VQA	OCR	Caption	Retrieval	Avg
AP (Huang et al., 2024)	12.45	9.30	9.50	0.70	7.99
MA (Xu et al., 2025)	5.35	16.25	13.40	1.65	9.16
APE (Huo et al., 2024)	3.90	2.80	12.90	1.40	5.25
HONES-RandHead	13.25	7.60	11.40	1.60	8.46
HONES-Gaussian	7.10	4.50	9.90	1.10	5.65
RandNeuron	0.95	1.80	2.05	0.15	1.24
HONES (Ours)	**24.15**	**21.30**	**26.00**	**5.80**	**19.31**

Table 8: Additional scale validation on **LLaVA-1.5-13B**. Relative performance drop (%) after masking the top-1% neurons selected by each ranking method. Higher values indicate greater causal importance (negative values indicate improvement). Avg denotes the arithmetic mean across the four tasks.

activations may indicate frequent usage. We compute

$$\mu_j^l \;=\; \frac{1}{|\mathcal{D}|}\sum_{x\in\mathcal{D}}\left|\tilde{a}_j^l(x)\right|,$$

and rank neurons by μ_j^l in descending order. This style of magnitude-based scoring is commonly adopted when probing functional neurons in VLMs (e.g., Xu et al., 2025).

Activation Probability Entropy (APE; "activation probability entropy"). Entropy-based selection aims to capture neurons with uncertain on/off patterns across samples, which has been used as a heuristic signal for identifying neurons that respond to diverse inputs. We compute the Bernoulli entropy of p_j^l:

$$H_j^l \;=\; -p_j^l\log(p_j^l) \;-\; (1-p_j^l)\log(1-p_j^l),$$

and rank neurons by H_j^l in descending order. This baseline follows probability-entropy-style scoring used in multimodal neuron analysis (e.g., Huo et al., 2024).

Model	Method	Time (s/inst.)↓
LLaVA-1.5-7B	Group Patching (Zhang and Nanda, 2024)	82.40 ± 5.60
LLaVA-1.5-7B	QRNCA (Chen et al., 2025)	46.50 ± 3.45
LLaVA-1.5-7B	**HONES**	$\mathbf{8.30 \pm 0.50}$
Qwen2.5-VL-7B	Group Patching (Zhang and Nanda, 2024)	106.20 ± 7.90
Qwen2.5-VL-7B	QRNCA (Chen et al., 2025)	53.80 ± 3.90
Qwen2.5-VL-7B	**HONES**	$\mathbf{10.15 \pm 0.65}$

Table 9: Efficiency assessment of neuron localization on VQA. **Time** reports the average localization runtime per instance (s/inst.) under the same hardware environment. All numbers are reported as mean $\pm$ standard deviation over 10 random trials.

Random Selection. As a lower-bound baseline, we uniformly sample B neurons from the full set of MLP neurons (across all layers), repeat this procedure for multiple seeds, and report the mean performance drop. This controls for the neuron budget and quantifies the expected effect of masking arbitrary neurons.

D.4 Absolute Performance Metrics

Table 7 reports the *absolute* performance scores (Accuracy for VQA, ANLS for OCR, BLEU-4 for Captioning, and NDCG@5 for Retrieval) after masking the top-1% neurons identified by different strategies. These raw metrics correspond to the relative drops reported in the main text. Consistent with the relative analysis, **HONES** yields the lowest absolute scores (i.e., the largest damage) across tasks, confirming its effectiveness in localizing causally critical neurons.

D.5 Additional Scale Validation on a Larger Backbone

To assess whether our neuron localization results remain stable beyond 7B dense backbones, we further evaluate HONES on a larger model, **LLaVA-1.5-13B**, under the same four-task setting as in the

Masked neuron group	VQA	OCR	Caption	Retrieval
Specific (only one task)				
VQA-specific	5.90	1.50	1.50	1.30
OCR-specific	−0.50	6.50	−0.40	−0.60
Caption-specific	0.50	−2.00	6.20	−0.20
Retrieval-specific	0.30	−2.30	−2.70	2.72
Pair (two-task shared)				
VQA+OCR	1.30	**9.80**	–	–
VQA+Caption	0.50	–	5.00	–
VQA+Retrieval	**10.70**	–	–	**6.30**
OCR+Caption	–	−1.20	0.90	–
OCR+Retrieval	–	2.00	–	0.40
Caption+Retrieval	–	–	−2.80	−0.50
Triple (three-task shared)				
VQA+OCR+Caption	1.40	3.10	**14.30**	–
VQA+OCR+Retrieval	1.00	4.50	–	0.60
VQA+Caption+Retrieval	0.40	–	0.90	−1.00
OCR+Caption+Retrieval	–	−1.20	−0.60	0.20
General (four-task shared)				
GENERAL	1.40	0.20	3.90	1.20
Random control (matched to the most critical shared group; mean of 10 trials)				
Random (same size)	0.20	0.80	−1.10	0.10

Table 10: Cross-task neuron ablation results on LLaVA-1.5. Values report relative performance change (%); positive indicates performance drop, while negative indicates performance gain. For each target task, we bold the single most damaging *shared-group* ablation (Pair/Triple/General), excluding task-specific (*Only*) and random controls. Random controls are computed by masking the same number of neurons as the most critical shared group for each target task (mean over 10 trials).

main paper. The results remain consistent with those observed in the 7B models, with HONES continuing to be the strongest localization method across all four tasks. Table 8 shows the results.

D.6 Efficiency Assessment on VQA

To further quantify the computational cost of neuron localization, we compare **HONES** with two representative baselines under a unified VQA evaluation protocol: the gradient-based attribution method **QRNCA** (Chen et al., 2025) and the explicit intervention-based method **Group Patching** (Zhang and Nanda, 2024). We choose these two methods because they represent the two types of comparison targets most relevant to HONES: one is gradient-based methods, which are used to examine the efficiency advantage of HONES as a gradient-free method; the other is causal intervention-based localization methods, which are used to compare the efficiency of HONES against strong intervention-style attribution methods. Meanwhile, in the preceding comparison of localization effectiveness, these two types of methods are also among the most competitive non-HONES baselines. Specifically, we conduct 10 independent random trials on the VQA analysis split, where each trial samples 200 examples for neuron ranking and localization. For fairness, all methods use the same candidate space (all FFN neurons), identical data preprocessing, and the same masking budget (top-1% neurons) throughout the comparison.

Under the same hardware environment (Tesla V100 GPU and Intel Xeon Gold 6230R CPU), we report the average localization runtime per instance for each method. As shown in Table 9, on LLaVA-1.5-7B and Qwen2.5-VL-7B, the per-instance localization time of HONES is consistently much lower than that of the two baselines, reaching 8.30±0.50s and 10.15±0.65s, respectively; correspondingly, QRNCA reaches 46.50±3.45s and 53.80±3.90s, and Group Patching reaches 82.40±5.60s and 106.20±7.90s. These results indicate that, under the same evaluation protocol, HONES achieves substantially higher neuron-localization efficiency than the compared baselines.

Masked neuron group	VQA	OCR	Caption	Retrieval
Specific (only one task)				
VQA-specific	14.80	1.90	2.10	0.60
OCR-specific	−3.40	9.20	−0.90	−0.25
Caption-specific	2.35	−3.80	7.35	0.10
Retrieval-specific	1.95	−2.70	−2.90	1.80
Pair (two-task shared)				
VQA+OCR	3.90	**12.50**	–	–
VQA+Caption	2.10	–	6.10	–
VQA+Retrieval	**26.01**	–	–	**3.41**
OCR+Caption	–	−2.30	−1.30	–
OCR+Retrieval	–	2.70	–	0.75
Caption+Retrieval	–	–	1.60	−0.20
Triple (three-task shared)				
VQA+OCR+Caption	4.80	3.80	**16.12**	–
VQA+OCR+Retrieval	5.20	5.80	–	0.90
VQA+Caption+Retrieval	3.40	–	3.00	0.35
OCR+Caption+Retrieval	–	−2.60	−2.20	0.15
General (four-task shared)				
GENERAL	4.10	3.90	5.25	0.95
Random control (matched to the most critical shared group; mean of 10 trials)				
Random (same size)	1.25	1.70	2.25	0.38

Table 11: Cross-task neuron ablation results on Qwen2.5-VL. Values report relative performance change (%); positive indicates performance drop, while negative indicates performance gain. For each target task, we bold the single most damaging *shared-group* ablation (Pair/Triple/General), excluding task-specific (*Only*) and random controls. Random controls are computed by masking the same number of neurons as the most critical shared group for each target task (mean over 10 trials).

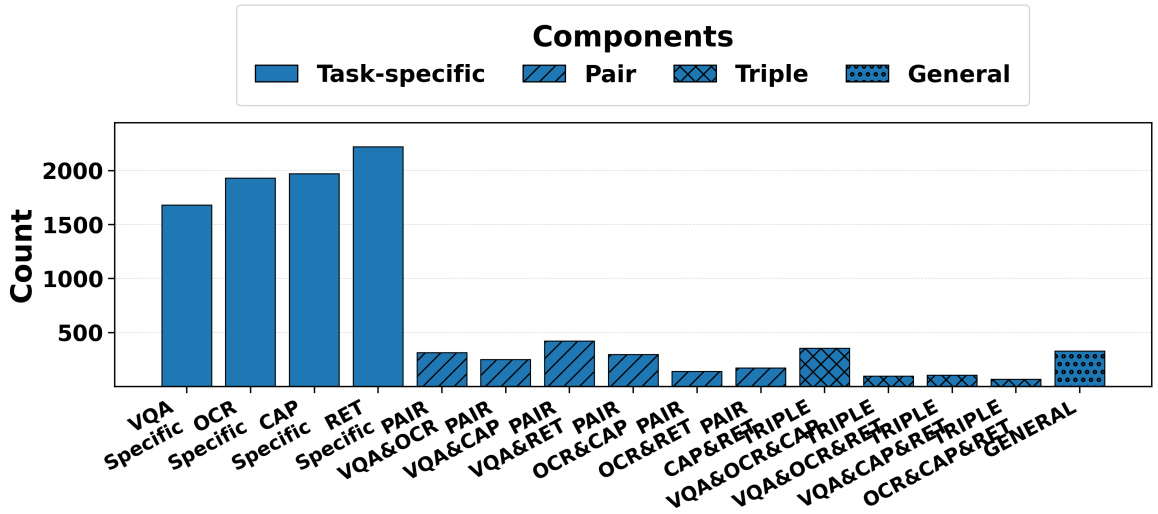
E Supplementary Cross-Task Neuron Ablation Results

E.1 Additional information for Cross-Task Neuron Analysis

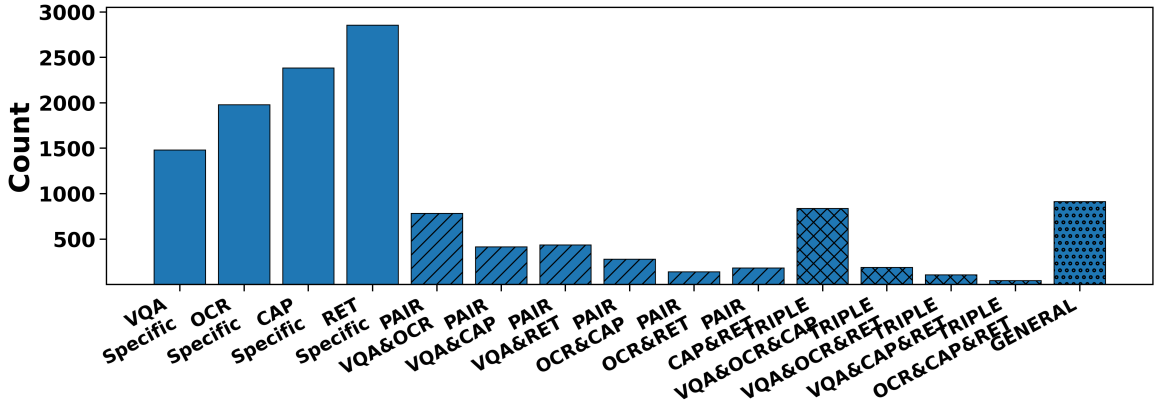
Full cross-task ablations and group definitions. We provide the complete cross-task neuron ablation tables for LLaVA-1.5 and Qwen2.5-VL in Table 10 and Table 11. All neuron groups are constructed from the **top-1%** task-critical neuron sets $\mathcal{N}_t$ identified by **HONES** (ranked by *Causal Write-in Effect*). Concretely, let $\mathcal{N}_t$ denote the top-1% task-critical neuron set for task $t \in \{\text{VQA}, \text{OCR}, \text{Cap}, \text{Ret}\}$. We partition neurons into four categories by set union/intersection: (1) *Task-specific*: neurons belonging to exactly one task set (e.g., $\mathcal{N}_{\text{VQA}} \setminus (\mathcal{N}_{\text{OCR}} \cup \mathcal{N}_{\text{Cap}} \cup \mathcal{N}_{\text{Ret}})$); (2) *Pair*: neurons shared by *exactly* two tasks (e.g., $(\mathcal{N}_{\text{VQA}} \cap \mathcal{N}_{\text{Ret}}) \setminus \mathcal{N}_{\text{OCR}} \setminus \mathcal{N}_{\text{Cap}}$); (3) *Triple*: neurons shared by *exactly* three tasks; (4) *General*: neurons shared by all four tasks ($\mathcal{N}_{\text{VQA}} \cap \mathcal{N}_{\text{OCR}} \cap \mathcal{N}_{\text{Cap}} \cap \mathcal{N}_{\text{Ret}}$). For each target task, we highlight the single most damaging *shared-group* ablation among Pair/Triple/*General* to define the *dominant salient neuron group* (as summarized in Fig. 3).

Neuron group composition. As shown in Fig. 8a and Fig. 8b, task-critical neurons are first identified *separately* for each of the four tasks by selecting the top-1% neurons. We then jointly consider the four task-wise top-1% sets and assign each neuron to an overlap group based on its cross-task membership pattern, yielding 15 mutually exclusive groups (4 task-specific, 6 pair, 4 triple, and 1 general). Across both backbones, the distribution is highly skewed: task-specific neurons account for the majority of critical units, while higher-order overlaps (pair/triple/general) form a much smaller fraction. Moreover, shared neurons are not uniformly spread across combinations; instead, they concentrate on a few pairs/triples that involve VQA, consistent with sparse and structured sharing rather than a dense universal subnetwork.

Additional evidence for VQA-centric selective sharing. Tables 10 and 11 further support the main-text conclusion that cross-task reuse is *sparse*



(a) Neuron overlap composition across tasks (LLaVA-1.5).



(b) Neuron overlap composition across tasks (Qwen2.5-VL).

Figure 8: Neuron overlap composition across tasks. Counts of task-critical neurons are partitioned into 15 mutually exclusive overlap groups induced by four tasks (VQA/OCR/Caption/Retrieval), including task-specific, pair, triple, and general components.

and structured, rather than uniformly distributed. First, across both backbones, the dominant salient neuron group for every target task always overlaps with VQA (e.g., *VQA+Ret* for Retrieval and VQA, *VQA+OCR* for OCR, and *VQA+OCR+Cap* for Caption), consistent with a VQA-centered hub. Second, the four-task *General* group is *not* the most damaging shared set, indicating that multi-task transfer is dominated by *combination-specific* bridges rather than a single universal subnetwork. Third, several non-VQA shared groups yield marginal or even negative drops, suggesting mixed-use computation and potential cross-task interference; this evidence supports our head-constrained, readout-aligned neuron discovery in **HONES**, which narrows the search to task-relevant routing pathways and thus better isolates verifiable causal units for reliable intervention.

Ruling out group-size artifacts via matched random ablation. Tables 10 and 11 also report matched random ablation: for each target task, we mask a random neuron set with the *same size* as the most critical shared group (mean over 10 trials). The resulting drops are consistently much smaller than those caused by the dominant salient neuron group, ruling out a trivial "mask-more-drop-more" explanation and supporting a structured causal dependency. This further implies that causally important neurons are sparsely and non-uniformly distributed, with their density varying across depth and tasks, rather than being evenly spread across neurons, suggesting that a small subset of units carries a disproportionate share of task-critical computation in multi-task VLMs.

Setting	Masked bridge	Caption drop (%)
LLaVA-1.5-7B		
Original	VQA+OCR+Cap	14.30
Original	VQA+Cap	5.00
Text-masked	VQA+OCR+Cap	4.37
Text-masked	VQA+Cap	11.00
Qwen2.5-VL-7B		
Original	VQA+OCR+Cap	16.12
Original	VQA+Cap	6.10
Text-masked	VQA+OCR+Cap	7.68
Text-masked	VQA+Cap	13.25

Table 12: Caption text-factor control. We compare two candidate shared bridges (Triple: VQA+OCR+Cap; Pair: VQA+Cap) under the original setting and the text-masked setting. After masking explicit visual text cues, the dominant bridge for Caption shifts from the Triple group to the Pair group across both backbones.

Task	Spec.	Pair	Triple	Gen.
GQA	6.90	36.50 (VQA+Retrieval)	1.20	0.62
TextVQA	3.40	21.00 (VQA+OCR)	1.50	1.05
Flickr30k-Cap	11.00	6.30	24.80 (VQA+OCR +Caption)	3.00
Flickr30k-Ret	1.60	5.35 (VQA+Retrieval)	1.10	1.20

Table 13: Out-of-distribution validation of shared-neuron saliency. Values denote relative performance drop (%) after masking different neuron groups, measured by Accuracy for VQA-GQA, ANLS for OCR-TextVQA, BLEU-4 for Caption-Flickr30k, and NDCG@5 for Retrieval-Flickr30k. Spec., Pair, Triple, and Gen. denote task-specific, best pair-shared, best triple-shared, and four-task shared groups, respectively. Larger values indicate stronger causal importance.

E.2 Exploring Text-Cue Dependence in Captioning via Visual-Text Masking

Motivation and setup. In the main experiments, Captioning can be influenced by explicit visual-text cues (e.g., scene text present in images). To test whether the *Triple* bridge (*VQA+OCR+Cap*) is driven by such cues, we perform a text-factor control study: we *mask out visual text regions* during inference for the Caption task, while keeping the rest of the pipeline unchanged. Concretely, we use the COCO-Text annotations (character/word bounding boxes) to overwrite text pixels with a neutral patch (e.g., mean-color) on the input image before feeding it to the model.

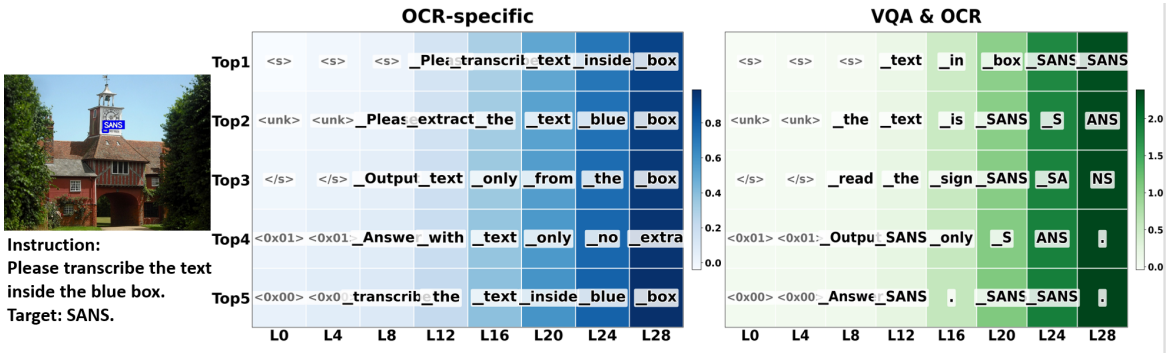
Key observation: the shared bridge contracts from Triple to Pair. As shown in Table 12, after removing explicit text cues, the dominant salient neuron group for Caption consistently shifts from *VQA+OCR+Cap* to *VQA+Cap* on both backbones, indicating that the previously observed Triple bridge is largely triggered by OCR-relevant computation induced by scene text. This provides direct causal support for the interpretation in the main text: Captioning couples *text understanding* (OCR) and *text generation* only when explicit text cues are present; otherwise it behaves as a more standard vision-to-text generation task that primarily shares with VQA-style semantic understanding.

E.3 Out-of-Distribution Validation of Shared-Neuron Saliency

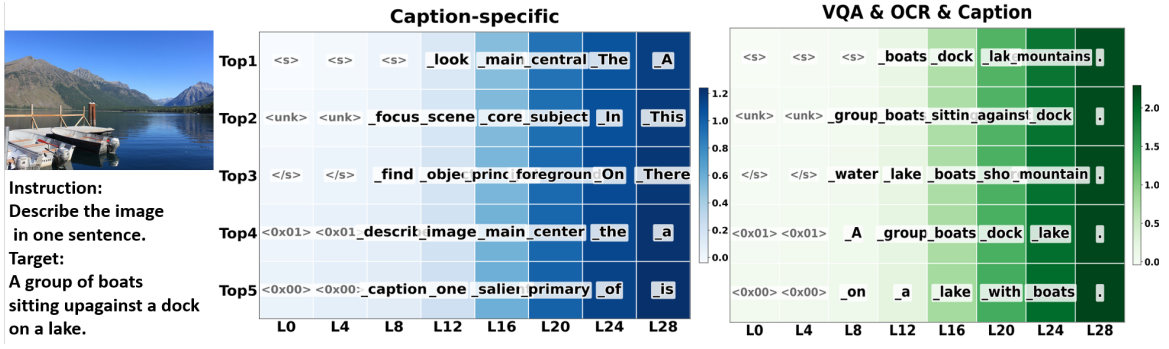
To examine whether the shared-neuron patterns identified on COCO generalize beyond the in-domain benchmark, we repeat the same group-level masking analysis on multiple out-of-distribution datasets, including GQA for VQA, TextVQA for OCR, and Flickr30k for Caption and Retrieval. Table 13 reports the relative performance drop after masking task-specific, best pair-shared, best triple-shared, and general neuron groups. Overall, the results remain qualitatively consistent with the in-domain findings: the most salient shared groups continue to induce larger performance drops, while task-specific and general groups have relatively weaker effects. This suggests that the shared-neuron patterns identified on COCO do not merely depend on a particular data distribution, but remain stable on related benchmarks with distribution shift, further supporting that these routing-and-neuron structures reflect more robust cross-task computation patterns.

E.4 Logit Lens Evidence: Shared Bridges as Anchors, Task Neurons as Scaffolds

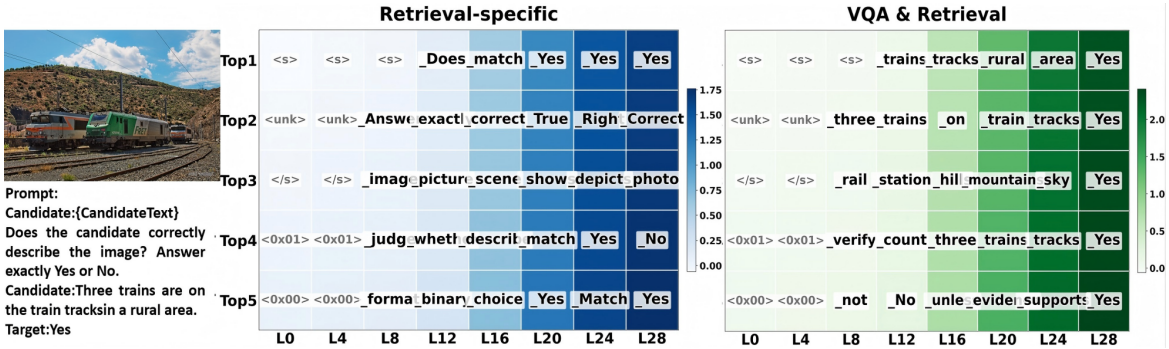
Visualization Setup. We perform an MLP-based Logit Lens visualization on single representative cases. At each transformer layer ℓ, we project the output of the MLP block directly to the vocabulary using the frozen language head to obtain the immediate token distribution. We compare forward passes under three settings: (i) baseline, (ii) masking the dominant salient neuron group, and (iii) masking the Task-specific group. We report ΔLogit (baseline $-$ masked), where a positive value indicates the neuron group's contribution to promoting a specific token. For clarity, we display the Top-5 tokens with the largest ΔLogit increase



(a) OCR case study in LLaVA-1.5.



(b) Caption case study in LLaVA-1.5.



(c) Retrieval case study in LLaVA-1.5.

Figure 9: Logit Lens case studies in LLaVA-1.5. Rows show Top-5 tokens and columns are sampled every 4 layers; color indicates Δlogit (baseline−masked). **(a)** OCR compares the OCR-specific group and the VQA&OCR shared group. **(b)** Caption compares the Caption-specific group and the VQA&OCR&Caption shared group. **(c)** Retrieval compares the Retrieval-specific group and the VQA&Retrieval shared group.

at representative layers.

OCR As shown in Fig. 9a, the OCR-only group acts as a *mode switcher*, boosting meta-tokens related to instruction adherence and output formatting (e.g., *transcribe/text/inside/reads*), thereby setting the operational state for text extraction. The Shared Bridge (VQA&OCR), however, performs the critical **visual-symbol grounding**, directly promoting tokens corresponding to the target string content (e.g., subword pieces of "SANS"), confirming its role as the character recognition engine.

Caption As shown in Fig. 9b, the Caption-only group constructs the *structural scaffolding*. It boosts tokens related to **saliency priors** (e.g., *main/central/core*) to direct attention, and discourse markers (e.g., *The/A/In*) to maintain syntactic fluency. Conversely, the Shared Bridge (VQA&OCR&CAP) functions as the **semantic filler**, strongly boosting content words that map to specific image entities (e.g., *boats/dock/lake*), populating the syntactic template with visual evidence. In contrast to task-specific neurons that mainly shape format and fluency, this bridge contributes more directly to content grounding along the core pathway.

Retrieval As shown in Fig. 9c, the Retrieval-only group enforces *format adherence*, primarily boosting binary decision tokens (e.g., *Yes/True/Match*) and generic visual terms, imposing a task-level **alignment bias**. Crucially, the Shared Bridge (VQA&RET) acts as the **semantic verifier**. Before the final layer, it explicitly activates fine-grained concepts present in the image (e.g., *trains/tracks/rural*) to validate the text candidate, providing the necessary **evidence** to support the final judgment.

The case visualizations across four tasks exhibit a relatively consistent pattern: task-specific neuron groups tend to boost task format, expression scaffolds, or a coarse candidate space; shared-bridge neuron groups are more likely to concentrate their boosts on fine-grained semantic tokens consistent with image evidence or the target answer. This pattern aligns with the cross-task ablation results, providing token-level supporting evidence for the conclusion that "task-only units mainly serve auxiliary adaptation, while shared bridges are closer to the core pathway."

F Supplementary Neuron Steering Results

F.1 Learning-based Neuron Scaling and Baselines

Steering target (bank) and evaluation split. We perform neuron steering on the same unified COCO-based split used for neuron analysis to keep the setting consistent across discovery, ablation, and editing. We learn neuron-wise scaling factors on the **2K dev** split and report performance on the **3K test** split with all backbone weights frozen. Instead of scaling the entire top-1% set, we steer a fixed-budget subset drawn from each task's *dominant salient neuron group* (defined in §5.2 and Appendix E.1), which improves stability and reduces unintended cross-task interference.

Learnable scaling wrapper. For each edited layer, we keep the original MLP weights (`gate_proj/up_proj/down_proj`) frozen and insert a lightweight wrapper on the intermediate FFN activation. We introduce a neuron-wise learnable vector `ns_scale` together with a binary mask `ns_mask` indicating which hidden dimensions belong to the chosen bank; only masked dimensions are learnable, while all others are fixed (scaling factor equals 1). This design yields a parameter-efficient editing interface while preserving the base model architecture.

Training objective (CE + KL regularization). We optimize `ns_scale` using a supervised task loss on dev data (cross-entropy with respect to ground-truth targets) while adding a KL regularizer that constrains the edited model's output distribution to stay close to the unedited backbone (teacher). The KL term effectively prevents overly aggressive edits and improves the stability of gains across tasks and settings.

Baselines (matched budget). We compare against train-free and learnable baselines under the same neuron budget: (i) **fixed amplification ($\times$2)** on the same bank; (ii) **dev-tuned amplification (grid search)** that selects an amplification factor on dev; (iii) **learnable scaling (random-neuron bank)** that learns the same number of scales but on random neurons; (iv) **w/o KL (CE only)** that drops the KL regularizer. (LoRA-style low-rank tuning was also explored under a low-budget setting but did not yield competitive improvements, so we omit it from the main comparisons.)

Task	Baseline	Mean Δ (pp)	SD (pp)	p-value
VQA	Base	2.15	1.00	***
	Fixed-Amp	1.25	0.96	***
	RandNeuron	2.05	1.04	***
OCR	Base	2.59	1.10	***
	Fixed-Amp	1.99	1.20	***
	RandNeuron	2.84	1.16	***
Caption	Base	1.22	0.90	***
	Fixed-Amp	1.17	0.92	***
	RandNeuron	1.47	0.94	***
Retrieval	Base	2.98	0.96	***
	Fixed-Amp	2.56	1.04	***
	RandNeuron	2.81	1.00	***

Table 14: Performance difference Δ (pp) between our bridge-neuron rescaling method and the baselines (Base, Fixed-Amp, and RandNeuron). Statistical significance is assessed via paired bootstrap resampling (B=1000, K=500). Asterisks indicate significance levels; *** denotes $p < 0.001$ (extremely significant).

F.2 OOD Benchmarks for Generalizability

GQA (OOD for VQA). GQA is a visual reasoning benchmark built to reduce superficial biases and emphasize compositional reasoning, with questions derived from scene-graph structure and controlled answer distributions (Hudson and Manning, 2019). We use it to test whether the learned scaling remains effective beyond the COCO-based evaluation distribution for VQA-style judgments.

TextVQA (OOD for OCR/text understanding). TextVQA focuses on questions that require reading and reasoning over scene text, pairing VQA-style prompts with OCR-relevant visual evidence (Singh et al., 2019). We use it as an out-of-distribution testbed for our OCR-related steering behavior.

Flickr30k (OOD for Caption and Retrieval). Flickr30k contains images with multiple human-written captions and is widely used for captioning and retrieval (Young et al., 2014). We use it to test the OOD generalization of our Caption/Retrieval steering under distribution shift.

OOD tuning protocol. For each OOD benchmark, we keep the neuron bank fixed (i.e., we do not re-localize heads or re-rank neurons) and only re-learn scaling factors on a small **20%** subset of the OOD data, then evaluate on the remaining **80%**. We additionally compare against (i) train-free amplification and (ii) directly transferring scales learned on the COCO-based dev split, to isolate the benefit of lightweight in-domain scale learning in OOD settings.

F.3 Statistical Significance Tests

We conduct statistical significance tests for the *bridge-neuron steering* results reported in Table 3. For each task, we compare Ours against three baselines: Base, Fixed-Amp (×2; a train-free uniform amplification baseline), and RandNeuron (learned scaling on a random neuron set matched in size to our bridge-neuron set). Let $\mathcal{D} = \{x_i\}_{i=1}^{N}$ denote the test instances of a task (image-level instances for VQA/OCR/Caption; query-level instances for Retrieval). We perform paired bootstrap resampling with B=1000 resamples. For each resample b, to reduce computation and ensure a consistent bootstrap scale across tasks, we sample K=500 indices with replacement from $\{1, \ldots, N\}$ to form a multiset $\mathcal{D}^{(b)}$, and evaluate OURS and each baseline on the same $\mathcal{D}^{(b)}$ to preserve pairing. For each baseline j, we define the resample-level improvement as the difference between the task metric of OURS and that of j on $\mathcal{D}^{(b)}$, where the metric is Acc. for VQA, ANLS for OCR, BLEU-4 for Caption, and NDCG@5 for Retrieval; for Caption, we compute corpus-level BLEU-4 on the resampled subset. We report the mean and sample standard deviation over the B resamples as the "Mean Δ" and "SD" columns, and compute a two-sided bootstrap p-value by doubling the smaller of the two empirical tail probabilities.

As shown in Table 14, our bridge-neuron rescaling yields statistically significant improvements over all three baselines across all four tasks on LLaVA-1.5 7B. Paired bootstrap tests further confirm the robustness of these gains, with all p-values being extremely small ($p < 0.001$), indicating strong statistical significance.

G Computational Resources.

All experiments were conducted on a single machine equipped with two NVIDIA Tesla V100 GPUs (32GB memory each), an Intel(R) Xeon(R) Gold 6230R CPU running at 2.10GHz (8 cores), and 64GB of RAM. Our interpretability pipeline—including head-level targeted causal ablations for routing localization, neuron attribution, cross-task/group-level neuron masking, and lightweight learnable neuron scaling for steering—was implemented in PyTorch with Hugging Face Transformers.

Task	Model	Prompt / Input Format
VQA	LLaVA-1.5	**USER:** <image> Question: {Question} Answer the question briefly using one or a few words. **ASSISTANT:**
	Qwen2.5-VL	**Structured chat message (via** processor.apply_chat_template**):** [{role: user, content: [{type: image, image: {Image}}, {type: text, text: ...}] }] **Text:** Question: {Question} Answer the question with a short phrase.
OCR	LLaVA-1.5	**USER:** <image> Please transcribe the text inside the blue box. Output text only. **ASSISTANT:**
	Qwen2.5-VL	**Structured chat message (via** processor.apply_chat_template**):** [{role: user, content: [{type: image, image: {Image}}, {type: text, text: ...}] }] **Text:** Transcribe the text inside the blue box. Output text only.
Caption	LLaVA-1.5	**USER:** <image> Describe the image in one sentence. **ASSISTANT:**
	Qwen2.5-VL	**Structured chat message (via** processor.apply_chat_template**):** [{role: user, content: [{type: image, image: {Image}}, {type: text, text: ...}] }] **Text:** Describe the image in one sentence.
Retrieval (Pairwise Matching)	LLaVA-1.5	**USER:** <image> Candidate: {CandidateText} Does the candidate correctly describe the image? Answer exactly Yes or No. **ASSISTANT:**
	Qwen2.5-VL	**Structured chat message (via** processor.apply_chat_template**):** [{role: user, content: [{type: image, image: {Image}}, {type: text, text: ...}] }] **Text:** Candidate: {CandidateText} Does the candidate correctly describe the image? Answer exactly Yes or No.

Table 15: Prompt and input formats used for evaluation. LLaVA-1.5 is shown in its common textual chat style, while Qwen2.5-VL uses structured multimodal chat messages serialized by the processor chat template. For retrieval, candidates are ranked by the model probability of the Yes response (pairwise matching).