Towards Understanding Multimodal Fine-Tuning: A Case Study into Spatial Features

Anonymous Author(s)
Affiliation
Address
email

Abstract

Vision–Language Models (VLMs) demonstrate strong performance on a wide range of tasks by fine-tuning pretrained language backbones to process projected visual tokens alongside text. Yet despite these empirical gains, it remains unclear how backbone representations adapt during multimodal training and when vision-specific capabilities emerge. In this work, we present the first mechanistic analysis of VLMs adaptation with stage-wise model diffing, a technique that isolates representational changes introduced during multimodal fine-tuning to reveal how a language model learns to "see". Concretely, we fine-tune sparse autoencoders trained on LLaMA-3.1-8B over multimodal activations from LLaVA-More (based on LLaMA-3.1-8B) using 50k VQAv2 pairs. We first isolate vision-preferring features that appear or reorient during multimodal fine-tuning. We then test for spatial selectivity using a controlled shift to spatial prompts and use attribution patching to identify the attention heads that causally activate these units. Our findings show that stage-wise model diffing reveals when and how spatially grounded multimodal features arise. It also provides a clearer view of modality fusion by showing how visual grounding reshapes features that were previously text-only. This methodology enhances the interpretability of multimodal training and provides a foundation for refining training regimes as well as auditing and steering models in safety-critical or domain-specific settings.

1 Introduction

Large vision–language models (VLMs) have achieved strong performance on multimodal tasks, including visual question answering (VQA), image captioning, object detection, and visual grounding [28, 27, 1, 2, 11]. These gains are typically realized by fine-tuning pretrained language models to process visual inputs through projected token sequences, allowing for seamless fusion of image and text representations [49, 19, 51, 13, 12]. Yet we lack a mechanistic account of how language representations adapt during multimodal training and when vision-specific capabilities emerge [23, 45, 46, 42, 5].

In this work, we introduce a methodology for analyzing multimodal adaptation in VLMs through stage-wise model diffing [6]. This mechanistic interpretability technique isolates representational changes introduced during fine-tuning by comparing sparse autoencoder (SAE) dictionaries across training stages, models, or datasets. By tracking how individual features rotate, emerge, or are repurposed, stage-wise diffing has been shown to uncover subtle shifts such as sleeper-agent features [21, 6]. We extend this approach to the multimodal setting, presenting the first application of stage-wise model diffing to study how pretrained language features evolve under visual grounding.

Concretely, we fine-tune LLaMA-Scope SAEs on activations extracted from the LLaVA-More model [20] on 50k samples from the VQAv2 dataset [17]. This warm-start preserves continuity with the original feature basis while allowing the SAE to adapt to new multimodal activations. By examining reconstruction quality and geometric alignment across different masking regimes, we isolate a subset of adapted features that show both a preference for vision-conditioned tokens and a marked rotation in their semantic geometry. These features serve as anchors for understanding how spatially grounded representations emerge within a pretrained language model.

To determine whether these features encode spatial reasoning, we introduce a controlled dataset shift from general VQA to spatially targeted queries. We track feature firing patterns under both distributions and identify a selective subset that is preferentially recruited for spatial prompts. These spatial candidates are further validated through both automatic interpretation (via GPT-4o-mini [38]) and manual inspection, revealing consistent activations on questions involving object placement, relative position, and orientation.

Finally, we use attribution patching to trace the causal pathways by which these spatial features are activated. We adapt gradient-based attribution techniques to measure how attention heads across the LLM backbone contribute to spatially selective feature activations. Our results reveal a sparse set of mid-to-deep layer heads that consistently drive spatial representations, often localizing to semantically meaningful regions and reappearing across related prompts. These findings support the hypothesis that a small number of specialized attention heads coordinate visual grounding within the model.

Our contributions offer a unified methodology for dissecting multimodal adaptation in large models:

- We propose stage wise model diffing as a method for dissecting multimodal adaptation in large language models, and show that it isolates the emergence of vision specific features within the backbone.
- We identify sparse SAE features that encode spatial relationships and are selectively activated by spatial prompts.
- We causally attribute these features to a small subset of attention heads using scalable patching methods.

By focusing on feature-level change, our approach complements high-level alignment analyses and probing-based methods, providing a deeper mechanistic view of how models “learn to see”. More broadly, this work offers a framework for auditing and refining multimodal training regimes, with implications for safety-critical domains and targeted fine-tuning in specialized applications.

2 Related Work

Model Diffing and Representation Dynamics. Model diffing techniques aim to isolate how internal representations change across models or training stages. Early work compared networks through global similarity measures, such as visualizing function space geometry with meta-SNE [37, 14], stitching intermediate representations across models [26, 3], or defining new similarity metrics [25, 4]. Others examined alignment at the level of individual neurons and features, showing evidence for convergent units across independently trained networks [29, 36]. Relatedly, Kissane et al. [24] investigated whether SAEs trained on a base model transfer to fine-tuned variants, finding that they largely do. This suggests that fine-tuning mostly preserves representational structure, with only a small subset of features altered—underscoring the importance of methods that can isolate and interpret precisely those changes. More recently, Anthropic’s Sparse Crosscoders [30] extended autoencoder-based analysis to discover shared features across layers and models, while stage-wise model diffing has proven more precise at uncovering subtle representational shifts, including sleeper-agent features [6] and interpretable distinctions between base and chat-tuned models [33].

Extensions to multimodal LLMs have highlighted how fine-tuning alters representational structure: Khayatan et al. [23] proposed concept-shift vectors for steering, and Venhoff et al. [45] showed that vision-language alignment converges in middle-to-late layers. However, these approaches remain at the level of semantic shifts or adapter alignment. In contrast, our work applies stage-wise model diffing with sparse autoencoders directly to the LLM backbone and the multimodal feature aligned model, providing the first mechanistic analysis of how multimodal fine-tuning rotates features and induces spatially grounded representations within pretrained language models.

Multimodal Mechanistic Interpretability. Compared to the rapidly growing literature on mechanistic interpretability of textual LLMs, relatively few studies have examined the internal mechanisms of multimodal large language models (MLLMs). Recent efforts in multimodal interpretability have explored a variety of approaches. Stan et al. [42] introduced an interpretability tool for vision–language models that leverages attention patterns, relevancy maps, and causal interventions to provide high-level explanations of model behavior. Basu et al. [5] applied causal intervention methods to trace information storage and transfer in MLLMs, while Palit et al. [39] used causal mediation analysis to study how BLIP integrates visual evidence into its predictions.

Other work has focused on probing the representations of vision encoders and multimodal backbones. Tong et al. [44] and Gandelsman et al. [15], along with Chen et al. [8], analyzed the interpretability of CLIP, revealing both its strengths and representational shortcomings. Schwettmann et al. [41] identified multimodal neurons that respond jointly to visual and textual concepts, while Jiang et al. [22] investigated how VLMs differentiate between hallucinated and real objects. More recent probing-based methods have attempted to map visual representations into linguistic space, such as Neo et al. [35], who projected visual embeddings onto language vocabulary, and Venhoff et al. [46], who studied the late emergence of visual representations within LLM backbones. These approaches leave open the mechanistic question of how multimodal fine-tuning restructures the language model's internal features, which is the focus of our work.

3 Preliminaries

3.1 Vision–Language Models

A vision–language model (VLM) consists of three components: a visual encoder f_V, a pretrained language model f_{LM}, and a trainable projector P. The visual encoder (e.g., CLIP [40]) extracts image patch embeddings $V = f_V(x) = [v_1, \dots, v_{N_V}]$, which the projector maps into the token space as $\tilde{V} = P(V)$. These projected image tokens are concatenated with tokenized text embeddings $T = [t_1, \dots, t_{N_T}]$ to form the multimodal sequence $X = [\tilde{v}_1, \dots, \tilde{v}_{N_V}, t_1, \dots, t_{N_T}]$.

The visual encoder functions as a perception tool to "see" the image, while the projector ensures that the extracted features can be seamlessly integrated into the input space of the language model.

The language model processes X through a stack of transformer layers, each consisting of multi-head self-attention (MHA) and a feed-forward network. For each head h, attention is computed as

$$\text{Attn}(Q, K, V) = \text{Softmax}\Big(\frac{QK^\top}{\sqrt{d_h}} + M\Big)V, \tag{1}$$

where M is the causal mask that prevents attending to future tokens. The outputs of all heads are concatenated and projected back into the hidden dimension, and the final hidden states are mapped through the unembedding matrix to yield next-token probabilities.

For our experiments, we adopt LLaVA-More [9], which extends LLaVA framework [32, 31] by integrating recent language models and diverse visual backbones; specifically, we use the variant combining the CLIP ViT-Large-Patch14–336 encoder with a LLaMA-3.1-8B language model backbone [18].

3.2 Sparse Autoencoders (SAEs)

Sparse Autoencoders (SAEs) are designed to reverse superposition by extracting features that are sparse, linear, and decomposable [7, 10]. A vanilla SAE consists of a single hidden layer where an input $x \in \mathbb{R}^D$ is linearly mapped to hidden activations

$$f(x) = \text{ReLU}(W_{\text{enc}}x + b_{\text{enc}}), \quad W_{\text{enc}} \in \mathbb{R}^{F\times D},\ b_{\text{enc}} \in \mathbb{R}^F, \tag{2}$$

which are then decoded back into the input space: $\hat{x} = W_{\text{dec}}f(x) + b_{\text{dec}}, \quad W_{\text{dec}} \in \mathbb{R}^{D\times F},\ b_{\text{dec}} \in \mathbb{R}^D$. Sparsity is encouraged via an L_1 penalty on the hidden activations, yielding the objective

$$\mathcal{L} = \|x - \hat{x}\|_2^2 + \lambda \sum_{i=1}^{F} \|f_i(x)\|_1. \tag{3}$$

Here, decoder columns $(W_{\text{dec}})_{:,i}$ define the direction of each feature in input space, while encoder rows $(W_{\text{enc}})_{i,:}$ act as detectors that determine when a feature is present. The ReLU activation enforces non-negativity, while the L_1 penalty drives most hidden units toward zero, ensuring sparse feature use.

Top-K Sparse Autoencoders Top-K Sparse Autoencoders (SAEs) [16] improve upon the vanilla formulation by enforcing sparsity through hard feature selection: for each input, only the K most active hidden units are retained, while all others are set to zero,

$$f_i(x) = \mathrm{TopK}\big(\mathrm{ReLU}(W_{\mathrm{enc},i} x + b_{\mathrm{enc},i})\big), \quad i \in \{1, \ldots, F\}.$$

As in the vanilla SAE, the surviving hidden activations are decoded back into input space. This mechanism achieves a sharper sparsity–fidelity tradeoff, prevents feature co-adaptation, and improves interpretability by ensuring only a small set of features contributes to reconstruction.

We build on the LLAMA-SCOPE suite of SAEs trained on LLaMA-3.1-8B [20], which introduce several enhancements over the baseline Top-K design; including incorporating the 2-norm of the decoder columns norms directly into the Top-K computation [43], post-processing Top-K SAEs to JumpReLU variants to ensure approximately K active features at inference, and using a K-annealing schedule to smoothly reduce activations during early training. Since the Vision–Language Model used in our experiments (LLaVA-More) is also built on the LLaMA-3.1-8B backbone, we initialize from its pretrained SAEs rather than retraining from scratch, enabling us to directly leverage millions of monosemantic features across layers.

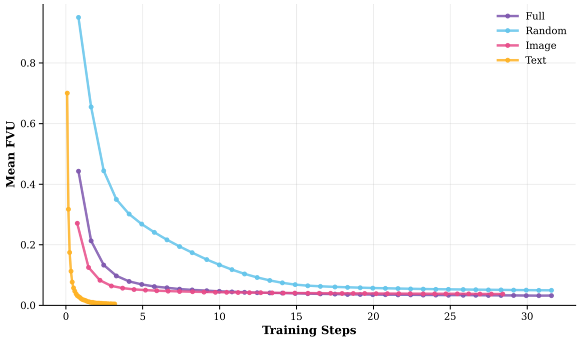
4 Adapting Language Dictionaries to Vision-Language Space

To study how multimodal fine-tuning reshapes internal representations, we adapt sparse autoencoders (SAEs) to the hidden states of LLAVA-MORE (Llama 3.1 8B backbone) [9]. We use 50k image–question pairs from the VQAv2 dataset [17], a widely used benchmark for visual question answering that pairs natural images with open-ended queries. Each SAE is attached to the output of a transformer block and trained on cached activations from these samples. Each image contributes a contiguous span of 575 visual tokens, while the accompanying question provides a variable-length text sequence, together enabling token-type–specific masking.

We initialize SAEs from the pretrained `llama_scope_lxr_8x` release, re-instantiated as a top-k model (k=50), to preserve a meaningful basis while enabling sparse, interpretable codes. This warm-start ensures continuity with the pretrained language feature space, whereas training from scratch serves only as a control. For comparison, we also train SAEs from random initialization. Training uses Adam with a layer-scaled learning rate, and cached activations are processed in padded mini-batches. To disentangle modality-specific contributions, we consider four regimes: (i) full sequence, (ii) image-only, using only the visual-token span, (iii) text-only, using only the non-visual span, and (iv) random initialization. In all cases the SAE receives the full hidden state sequence, but masking controls which token spans contribute to the training signal.

We evaluate reconstruction quality using the fraction of variance unexplained (FVU) and report sparsity to verify that codes remain selective. Evaluation is performed on a held-out split. Figure 1 shows FVU as a function of tokens seen across layers and masking regimes. Text-only SAEs converge rapidly, while image-only and full-token regimes converge slowly and plateau at higher error, reflecting the mismatch between projector embeddings and the LLM basis. Random initialization performs worst, underscoring the importance of starting from a pretrained language dictionary. These findings establish text-only SAEs as a reliable reconstruction baseline, which we later use for stage-wise feature diffing.

Implications for stage-wise model diffing. Stage-wise diffing assumes that fine-tuning induces *localized* (feature-level) changes rather than wholesale rotations. Prior work reports that image-token representations in early layers exhibit higher reconstruction error than text tokens, indicating a distributional gap between projector outputs and the LLM basis [47]. Consistent with this, our decoder–cosine analysis (Appx.Fig.6) shows that *text-only* SAEs remain highly aligned to the base LLM dictionary across layers, whereas *image-only* and *full-sequence* SAEs undergo large rotations in shallow layers and only align in later layers. We also note that text-only SAEs begin with slightly higher error in the very first layers but adapt extremely quickly, converging to near-zero reconstruction, while image and full-sequence SAEs plateau at higher error—underscoring the instability of projector-driven spans (see Appx.Fig.7). We therefore avoid stage-wise diffing on image-only or full-sequence



Metric	Full	Random	Image	Text
Mean	0.032	0.050	0.037	**0.005**
Std	0.028	0.041	0.027	**0.009**
Min	0.013	0.020	0.017	**0.000**
Max	0.123	0.198	0.123	**0.037**
Tokens (M)	31.6	31.6	28.4	3.2

Figure 1: **SAE adaptation on LLaVA-MORE.** Left: Mean fraction of variance unexplained (FVU) across layers on the validation set during adaptation. Right: Summary statistics of FVU values on the validation set, with decimal alignment; the lowest mean is highlighted in **bold**.

SAEs in early layers, and focus on text-only SAEs and on later layers where alignment is stable and feature-level identifiability is more plausible.

5 Identifying Adapted Features

Our goal is to find SAE features that (i) exhibit a *modality preference* for vision input and (ii) reorient geometrically after multimodal adaptation. Such features are the best targets for stage-wise diffing and later causal probes.

5.1 Signals

Modality preference (variance gap). View each SAE feature f as a latent direction whose activation on hidden state x is $h_f(x)$. We quantify f's preference for vision states using the variance gap:

$$\Delta_f \;=\; \mathbb{E}_{\text{vision}}\big[h_f^2\big] \;-\; \mathbb{E}_{\text{text}}\big[h_f^2\big].$$

$\mathbb{E}_{\text{vision}}[\cdot]$ is taken from VQA runs of the VLM (image + question), while $\mathbb{E}_{\text{text}}[\cdot]$ comes from the base LLM on the same prompts, where images are replaced with captions, and the model receives only textual input. A large Δ_f indicates that f shows stronger activation on image-conditioned representations, suggesting visual specialization.

Geometric reorientation (decoder cosine). To test whether f has been *repurposed* by multimodal fine-tuning, we compare its decoder direction before and after adaptation. Let $W^{\text{LLM}}_{\text{dec},f}$ be the decoder row for feature f in the base language SAE and $W^{\text{VLM}}_{\text{dec},f}$ the corresponding row in the VLM-adapted SAE. We compute

$$c_f \;=\; \cos\!\big(W^{\text{LLM}}_{\text{dec},f},\, W^{\text{VLM}}_{\text{dec},f}\big).$$

High c_f means the semantic direction of f stayed aligned with the original language dictionary; low c_f indicates a substantial rotation, consistent with a reallocation of f to encode new multimodal structure. We use decoder rows rather than encoder parameters because decoder directions more directly index the feature's semantics.

5.2 Selection Procedure

We identify adapted features using a *two-stage filter*. All features from every layer are pooled together, and thresholds are computed over this global set. Stage one retains the top $p_{\text{gap}} = 20\%$ of features by variance gap Δ_f, ensuring a preference for vision-conditioned activations. Stage two further narrows this pool to the bottom $p_{\text{cos}} = 20\%$ by cosine similarity c_f, isolating those that underwent the strongest decoder rotations. This procedure produces a single globally defined adapted set comprising under 5% of all features. The joint distribution of variance gap and cosine similarity is shown in Fig. 2, with selected adapted features highlighted. Additional summaries, such as counts of adapted features per layer and their mean cosine similarities, are provided in Appx. Fig. 8a and Appx. Fig. 8b.

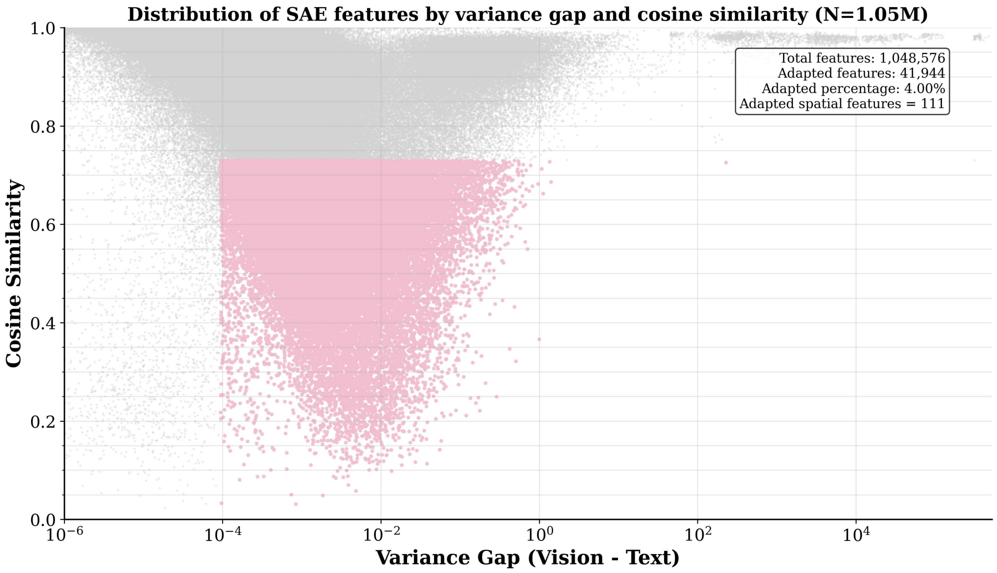


Figure 2: **Joint distribution of variance gap vs. decoder cosine** for all SAE features (gray). Points highlighted in pink are retained by our two-stage filter yielding the globally defined adapted set.

6 Case Study: Identifying Spatial Reasoning Features

We aim to isolate SAE features that encode spatial grounding by comparing firing patterns under a controlled dataset shift from general VQA to spatial queries.

Datasets. We consider two evaluation sets derived from VQAv2. The baseline is the full validation split, denoted $\mathcal{D}_{\text{base}}$. To induce a targeted shift, we construct a spatial subset $\mathcal{D}_{\text{sp}}$ by filtering questions that contain spatial cues (e.g., *left/right/above/behind*). This contrast tests whether some SAE features are selectively recruited under spatial reasoning.

Firing frequencies. Let $h_f(x_t) \geq 0$ denote the activation of feature f on token t of input x. For a dataset $\mathcal{D}$, the firing frequency of f is

$$p_f(\mathcal{D}) \;=\; \frac{1}{n(\mathcal{D})} \sum_{x \in \mathcal{D}} \sum_{t} \mathbf{1}\{h_f(x_t) > 0\},$$

where $n(\mathcal{D})$ is the total number of tokens.

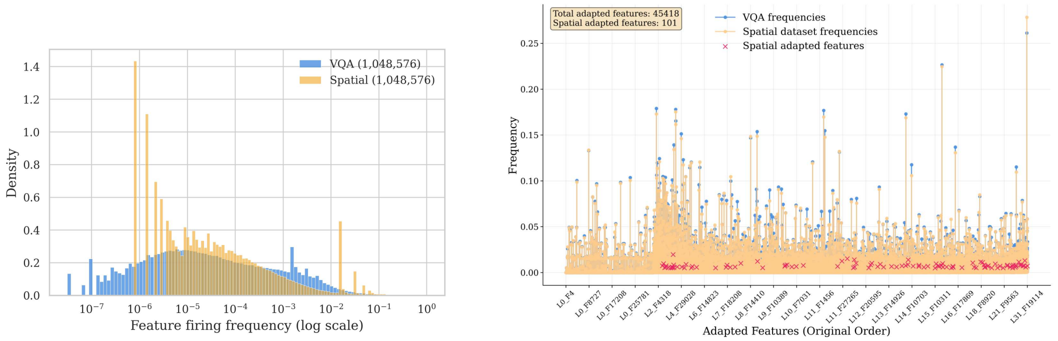
Distribution shift. Figure 3a compares the empirical distributions of feature firing frequencies under $\mathcal{D}_{\text{base}}$ and $\mathcal{D}_{\text{sp}}$. The spatial subset exhibits a heavier right tail, suggesting selective recruitment under spatial queries.

For each feature f, we compute the frequency gap

$$\Delta p_f \;=\; p_f(\mathcal{D}_{\text{sp}}) - p_f(\mathcal{D}_{\text{base}}),$$

and the odds ratio OR_f comparing firing counts across the two splits. Features with large Δp_f and $\mathrm{OR}_f > 1$ are flagged as spatial *candidates*.

Selection and outcome. From the spatial candidates, we retain only those that also lie in the adapted set $\mathcal{A}$ from Sec. 5, ensuring they both reorient under multimodal fine-tuning and respond to spatial distribution shifts (with scatter plot shown in Appx.Fig.9). To remove prompt-lexical artifacts, we further probe with a neutral instructions such as *"Describe the positions of all objects in the image."* and Features that remain active and image-token–dominant are preserved, while prompt-specific units are discarded. Figure 3b visualizes the result: across all adapted features, it compares firing frequencies on $\mathcal{D}_{\text{base}}$ and $\mathcal{D}_{\text{sp}}$, highlighting the subset that survives this full pipeline. The plot shows that adapted features span a wide dynamic range, with the retained spatial set concentrated in the high-frequency tail under $\mathcal{D}_{\text{sp}}$.



(a) Overall distribution shift in feature firing frequencies when moving from generic VQA to spatial queries.

(b) Adapted features under both splits, with spatially selective survivors highlighted.

Figure 3: Identifying spatial reasoning features. Evidence of a distribution shift from $\mathcal{D}$base to $\mathcal{D}$sp, with adapted features highlighted after the full selection pipeline.

7 AutoInterp and Manual Inspection

To characterize the adapted features, we developed an automated interpretation pipeline. For each feature, we gather its top activating VQA question samples and pass them to the `gpt-4o-mini` [38] API in JSON mode. The model is instructed to decide whether the feature is SPATIAL, provide a confidence score, and return a short one-sentence description of the concept it detects, along with common question patterns and cue counts. This yields structured interpretations rather than binary labels alone, allowing us to associate each feature with a candidate spatial meaning. All outputs are stored with the original metrics from Sec. 5. We additionally conduct a light manual pass to verify the results. The final retained set thus combines automatic labeling, descriptive interpretation, and human sanity checks.

8 Attribution Patching to Identify Spatial Heads

Method. Attribution patching [34] is a scalable alternative to activation patching [50], which measures causal effects by replacing activations with counterfactual values. While activation patching requires a separate forward pass per intervention, attribution patching uses a gradient-based linear approximation to estimate the effect of all interventions with only two forward passes and one backward pass. This makes it practical to probe attribution scores across all layers and attention heads in large multimodal models.

We adapt attribution patching to identify which attention heads drive spatially selective SAE features. For a target feature f at layer L, we define a scalar objective by projecting the layer-L activations onto the SAE decoder vector. Gradients of this objective with respect to upstream query/key activations indicate how strongly each attention head contributes to f.

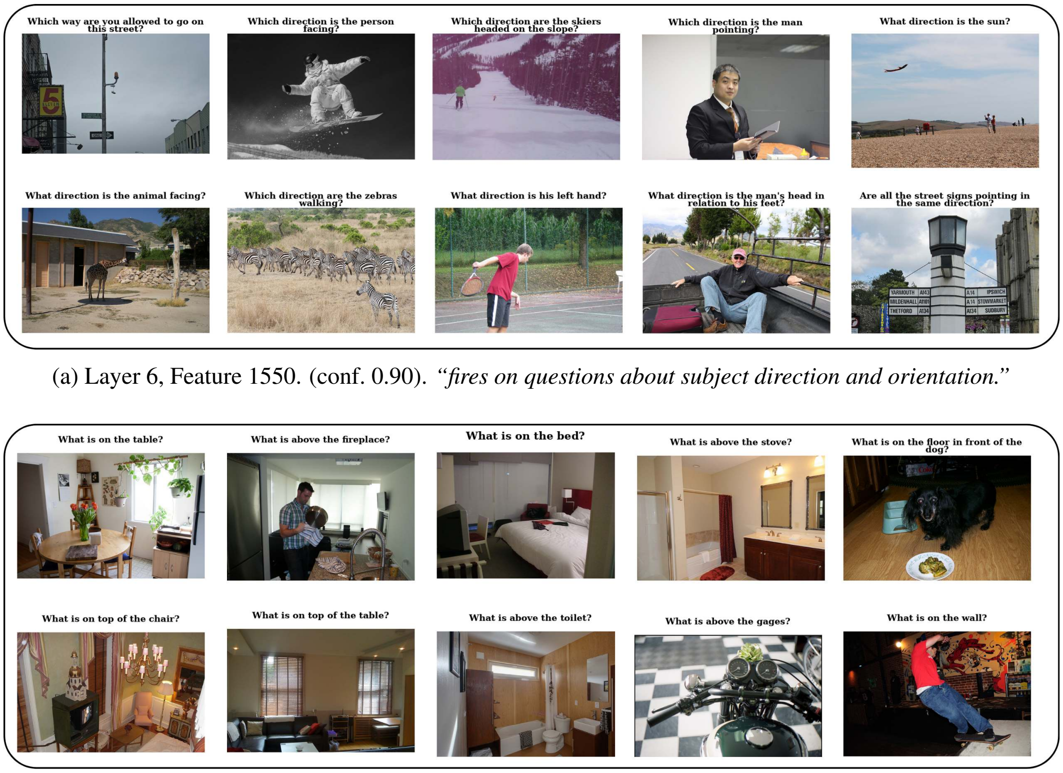
We compare two runs:

- **Clean run:** the original image–text input.
- **Corrupt run:** the same input, but with layer-0 visual token embeddings replaced by a *mean embedding* computed over many VQA samples. This corruption preserves plausible distributional statistics while deliberately suppressing spatial information.

We then compute two attribution variants, differing in whether the perturbation direction is taken from the corrupted or the clean representation:

$$\begin{aligned} \text{Method A:} \quad & (\text{corr} - \text{clean}) \cdot \nabla_{\text{clean}}, \\ \text{Method B:} \quad & (\text{clean} - \text{corr}) \cdot \nabla_{\text{corr}}. \end{aligned}$$

Method A measures how strongly the clean gradients indicate that ablating spatial detail affects the feature, whereas Method B measures how strongly the corrupted gradients indicate that retaining


Which way are you allowed to go on this street?
Which direction is the person facing?
Which direction are the skiers headed on the slope?
Which direction is the man pointing?
What direction is the sun?
What direction is the animal facing?
Which direction are the zebras walking?
What direction is his left hand?
What direction is the man's head in relation to his feet?
Are all the street signs pointing in the same direction?
What is on the table?
What is above the fireplace?
What is on the bed?
What is above the stove?
What is on the floor in front of the dog?
What is on top of the chair?
What is on top of the table?
What is above the toilet?
What is above the gages?
What is on the wall?

(a) Layer 6, Feature 1550. (conf. 0.90). *"fires on questions about subject direction and orientation."*

(b) Layer 27, Feature 12845. (conf. 0.90). *"fires on questions about objects and their relative positions"*

Figure 4: **Qualitative Auto-Interp examples.** Top-activating VQA samples for three adapted features automatically labeled as SPATIAL, with short GPT-4o-mini–generated descriptions.

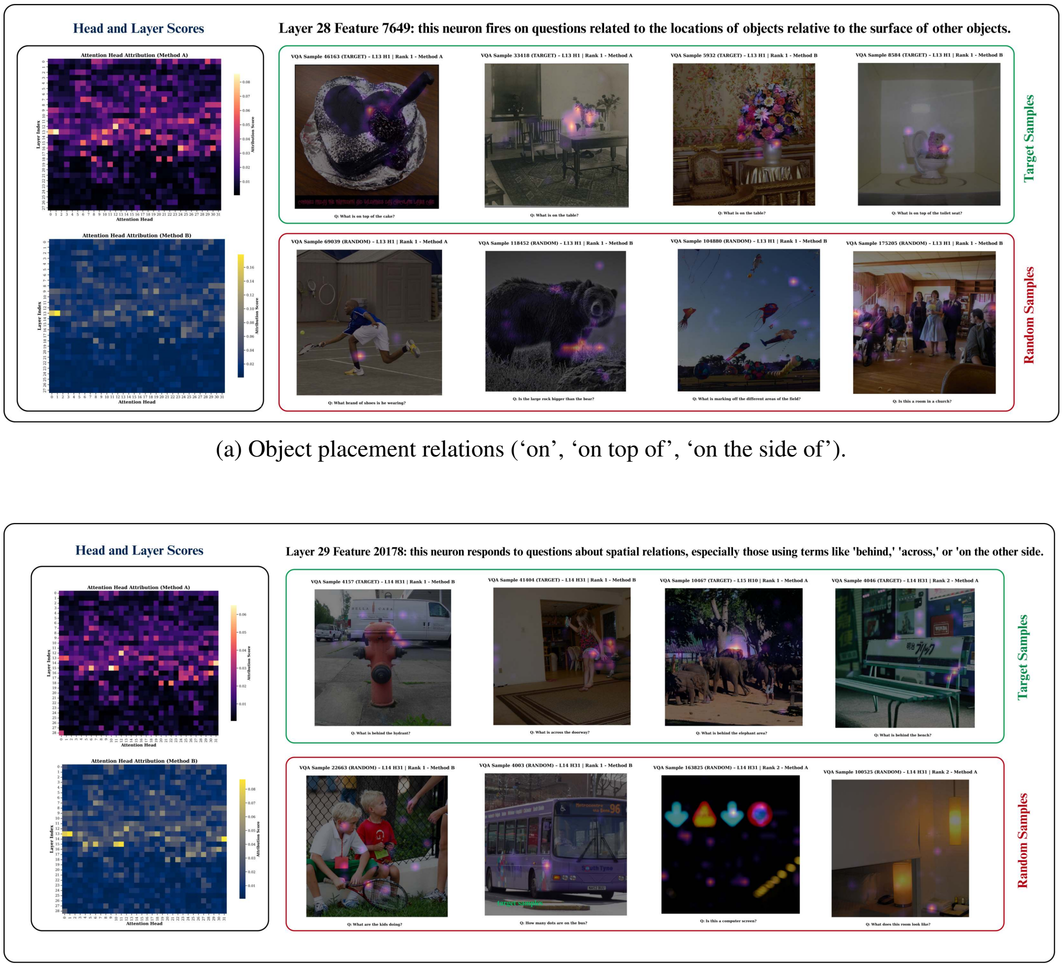
spatial detail matters. In both cases, we obtain per-layer and per-head attribution scores, averaged over the top-k VQA samples that most strongly activate f.

Results. Across the spatially selective features we examined, attribution patching with both methods reveals consistent trends. Layer-wise attribution curves typically peak in mid-to-deep layers, consistent with the emergence of spatial features in Sec. 6. At the head level, both methods generally highlight a small subset of heads with notably high scores, and the top heads identified are often consistent across the two attribution methods. This suggests that spatial information is mediated by a specialized group of heads rather than being spread uniformly across the model.

To illustrate, Figure 5 shows representative examples of individual spatial features. In each case, attribution scores isolate a handful of mid- to deep-layer heads, and qualitative attention visualizations confirm that these heads attend to image regions consistent with the query relation (e.g., "behind," "on top of," or "on the side of"). Interestingly, some of the same heads recur across related spatial relations (Appx. 11). Beyond spatial prompts, we also find that these same heads often attend to semantically meaningful regions such as objects or attributes, even under non-spatial queries (Appx. Fig. 12). As for control, low-scoring heads on the same target images fail to attend to meaningful regions, further highlighting that only the high-attribution heads carry spatial–semantic signal (Appx. Fig. 13). Moreover, random control samples show diffuse or incoherent attention patterns, underscoring that the observed alignment is specific to selective features (bottom rows in Figure 5).

9 Limitations

The main limitation of our work is the need to further evaluate the faithfulness of the identified features: while our analyses suggest spatial selectivity, stronger validation through ablations or steering interventions on spatial benchmarks is required. In addition, our study is restricted to a single



(a) Object placement relations (‘on’, ‘on top of’, ‘on the side of’).

(b) Spatial relation queries (‘behind’, ‘across’, ‘on the other side’).

Figure 5: Neuron interpretability examples of object placement and spatial relations.

multimodal model (LLaVA-More with a LLaMA-3.1-8B backbone). Extending the methodology to diverse language backbones and larger training corpora will be important to assess the generality of our findings.

10 Conclusion

We set out to understand how a pretrained language backbone learns to “see” under multimodal fine-tuning. By extending stage wise model diffing to the vision–language setting, we isolated vision-preferring features that undergo strong rotations during training, showed that a subset reliably encodes spatial relations, and traced their causal drivers to a small number of mid-to-deep attention heads. These results demonstrate that multimodal adaptation is neither diffuse nor opaque: it can be localized, probed, and explained at the feature level. Beyond spatial reasoning, our methodology offers a general framework for uncovering how new capabilities emerge in large models, with practical implications for auditing, safety, and domain-specific fine-tuning. We view this work as an early step toward a mechanistic science of multimodal training, where models can be interpreted not only by what they do, but by how their internal features evolve.

References

[1] Mistral AI. *Pixtral 12B: A New Frontier in Image and Text Understanding*. `https://mistral.ai/news/pixtral-12b/`. Accessed: 2024-12-21. Sept. 2024.

[2] Jinze Bai et al. "Qwen-vl: A frontier large vision-language model with versatile abilities". In: *arXiv preprint arXiv:2308.12966* (2023).

[3] Yamini Bansal, Preetum Nakkiran, and Boaz Barak. "Revisiting model stitching to compare neural representations". In: *Advances in neural information processing systems* 34 (2021), pp. 225–236.

[4] Serguei Barannikov et al. "Representation topology divergence: A method for comparing neural network representations". In: *arXiv preprint arXiv:2201.00058* (2021).

[5] Samyadeep Basu et al. "Understanding information storage and transfer in multi-modal large language models". In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 7400–7426.

[6] Trenton Bricken et al. "Stage-Wise Model Diffing". In: (2024). `https://transformer-circuits.pub/2024/model-diffing/index.html`.

[7] Trenton Bricken et al. "Towards Monosemanticity: Decomposing Language Models With Dictionary Learning". In: *Transformer Circuits Thread* (2023). https://transformer-circuits.pub/2023/monosemantic-features/index.html.

[8] Haozhe Chen et al. "Interpreting and controlling vision foundation models via text explanations". In: *arXiv preprint arXiv:2310.10591* (2023).

[9] Federico Cocchi et al. "LLaVA-MORE: A Comparative Study of LLMs and Visual Backbones for Enhanced Visual Instruction Tuning". In: *arXiv preprint arXiv:2503.15621* (2025).

[10] Hoagy Cunningham et al. "Sparse autoencoders find highly interpretable features in language models". In: *arXiv preprint arXiv:2309.08600* (2023).

[11] Matt Deitke et al. "Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models". In: *arXiv preprint arXiv:2409.17146* (2024).

[12] Guanting Dong et al. "Progressive Multimodal Reasoning via Active Retrieval". In: 2024. URL: `https://api.semanticscholar.org/CorpusID:274859457`.

[13] Yuhao Dong et al. "Insight-V: Exploring Long-Chain Visual Reasoning with Multimodal Large Language Models". In: *arXiv preprint arXiv:2411.14432* (2024).

[14] Dumitru Erhan et al. "Why Does Unsupervised Pre-training Help Deep Learning?" In: 11 (Mar. 2010), pp. 625–660. ISSN: 1532-4435.

[15] Yossi Gandelsman, Alexei A Efros, and Jacob Steinhardt. "Interpreting clip's image representation via text-based decomposition". In: *arXiv preprint arXiv:2310.05916* (2023).

[16] Leo Gao et al. "Scaling and evaluating sparse autoencoders". In: *arXiv preprint arXiv:2406.04093* (2024).

[17] Yash Goyal et al. "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*. 2017, pp. 6904–6913.

[18] Aaron Grattafiori et al. "The llama 3 herd of models". In: *arXiv preprint arXiv:2407.21783* (2024).

[19] Jiayi He et al. "Self-correction is more than refinement: A learning framework for visual and language reasoning tasks". In: *arXiv preprint arXiv:2410.04055* (2024).

[20] Zhengfu He et al. "Llama scope: Extracting millions of features from llama-3.1-8b with sparse autoencoders". In: *arXiv preprint arXiv:2410.20526* (2024).

[21] Evan Hubinger et al. "Sleeper agents: Training deceptive llms that persist through safety training". In: *arXiv preprint arXiv:2401.05566* (2024).

[22] Nick Jiang et al. "Interpreting and editing vision-language representations to mitigate hallucinations". In: *arXiv preprint arXiv:2410.02762* (2024).

[23] Pegah Khayatan et al. "Analyzing Fine-tuning Representation Shift for Multimodal LLMs Steering alignment". In: *arXiv preprint arXiv:2501.03012* (2025).

[24] Connor Kissane et al. "SAEs (usually) Transfer Between Base and Chat Models". Interim report on AI Alignment Forum. AI Alignment Forum post. July 2024.

[25] Simon Kornblith et al. "Similarity of neural network representations revisited". In: *International conference on machine learning*. PMlR. 2019, pp. 3519–3529.

[26] Karel Lenc and Andrea Vedaldi. “Understanding image representations by measuring their equivariance and equivalence”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 991–999.

[27] Bo Li et al. “Llava-onevision: Easy visual task transfer”. In: *arXiv preprint arXiv:2408.03326* (2024).

[28] Feng Li et al. “Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models”. In: *arXiv preprint arXiv:2407.07895* (2024).

[29] Yixuan Li et al. “Convergent learning: Do different neural networks learn the same representations?” In: *arXiv preprint arXiv:1511.07543* (2015).

[30] Jack Lindsey et al. *Sparse Crosscoders for Cross-Layer Features and Model Diffing*. Published on Transformer Circuits Thread; `https://transformer-circuits.pub/2024/crosscoders/index.html`. Oct. 2024.

[31] Haotian Liu et al. “Improved baselines with visual instruction tuning”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2024, pp. 26296–26306.

[32] Haotian Liu et al. “Visual instruction tuning”. In: *Advances in neural information processing systems* 36 (2023), pp. 34892–34916.

[33] Julian Minder et al. “Robustly identifying concepts introduced during chat fine-tuning using crosscoders”. In: *arXiv preprint arXiv:2504.02922* (2025).

[34] Neel Nanda. *Attribution Patching: Activation Patching at Industrial Scale*. `https://www.neelnanda.io/mechanistic-interpretability`. Accessed: 2025-08-23. 2023.

[35] Clement Neo et al. “Towards interpreting visual information processing in vision-language models”. In: *arXiv preprint arXiv:2410.07149* (2024).

[36] Chris Olah et al. “Zoom In: An Introduction to Circuits”. In: *Distill* (2020). https://distill.pub/2020/circuits/zoom-in. DOI: `10.23915/distill.00024.001`.

[37] Christopher Olah. *Visualizing Representations: Deep Learning and Human Beings*. `https://colah.github.io/posts/2015-01-Visualizing-Representations/`. Accessed: 2025-08-23. 2015.

[38] OpenAI. *GPT-4O-Mini: Advancing Cost-Efficient Intelligence*. `https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/`. Accessed: 2024-12-21. 2024.

[39] Vedant Palit et al. “Towards vision-language mechanistic interpretability: A causal tracing tool for blip”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 2856–2861.

[40] Alec Radford et al. “Learning transferable visual models from natural language supervision”. In: *International conference on machine learning*. PmLR. 2021, pp. 8748–8763.

[41] Sarah Schwettmann et al. “Multimodal neurons in pretrained text-only transformers”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 2862–2867.

[42] Gabriela Ben Melech Stan et al. “LVLM-Interpret: an interpretability tool for large vision-language models”. In: *arXiv preprint arXiv:2404.03118* (2024).

[43] Adly Templeton et al. “Scaling Monosemanticity: Extracting Interpretable Features from Claude 3 Sonnet”. In: *Transformer Circuits Thread* (2024). URL: `https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html`.

[44] Shengbang Tong et al. “Eyes wide shut? exploring the visual shortcomings of multimodal llms”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024, pp. 9568–9578.

[45] Constantin Venhoff et al. “How Visual Representations Map to Language Feature Space in Multimodal LLMs”. In: *arXiv preprint arXiv:2506.11976* (2025).

[46] Constantin Venhoff et al. “Too Late to Recall: The Two-Hop Problem in Multimodal Knowledge Retrieval”. In: *Mechanistic Interpretability for Vision (Non-proceedings Track), CVPR 2025*. 2025. URL: `https://openreview.net/forum?id=VUhRdZp8ke`.

[47] Constantin Venhoff et al. “Too Late to Recall: The Two-Hop Problem in Multimodal Knowledge Retrieval”. In: *CVPR 2025 Workshop on Mechanistic Interpretability of Vision (MIV)*. Non-proceedings Track Poster. 2025.

[48] Zhiyu Wu et al. “DeepSeek-VL2: Mixture-of-Experts Vision-Language Models for Advanced Multimodal Understanding”. In: *arXiv preprint arXiv:2412.10302* (2024).

[49] Guowei Xu et al. “LLaVA-o1: Let Vision Language Models Reason Step-by-Step”. In: *arXiv preprint arXiv:2411.10440* (2024).

[50] Fred Zhang and Neel Nanda. “Towards Best Practices of Activation Patching in Language Models: Metrics and Methods”. In: *International Conference on Learning Representations (ICLR)*. arXiv:2309.16042. 2024. URL: `https://doi.org/10.48550/arXiv.2309.16042`.

[51] Ruohong Zhang et al. “Improve vision language model chain-of-thought reasoning”. In: *arXiv preprint arXiv:2410.16198* (2024).

A Appendix

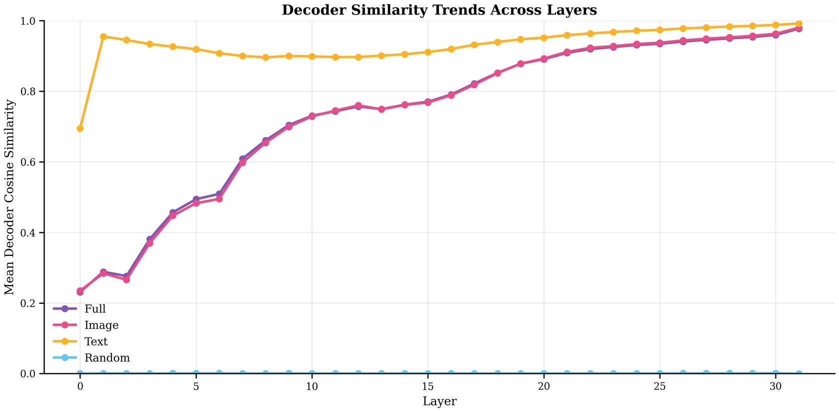


Figure 6: **Decoder cosine similarity vs. layer (LLM SAE vs. VLM SAE).** Text-only stays highly aligned across layers; image-only and full-sequence rotate in shallow layers and align later; random remains near zero. Higher cosine indicates closer alignment of SAE decoder directions.

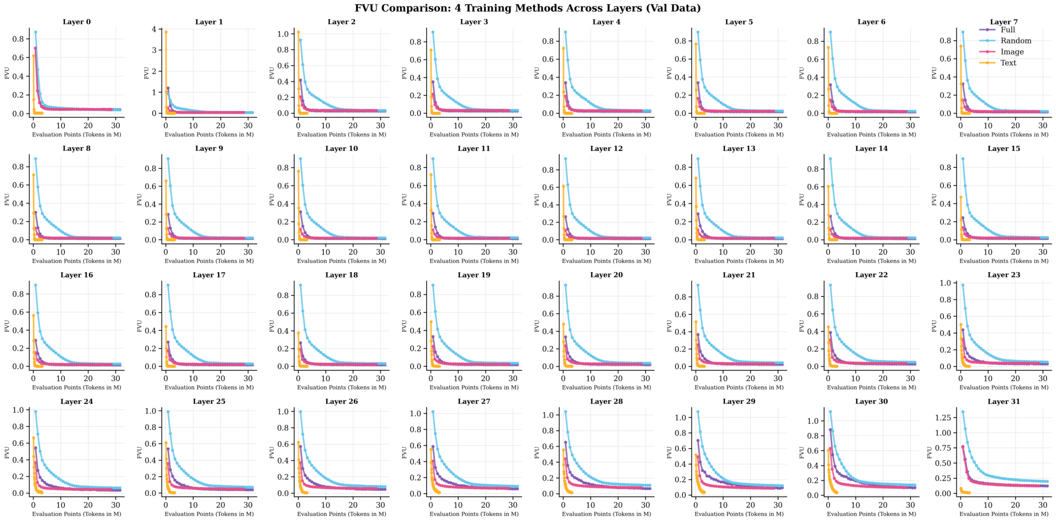
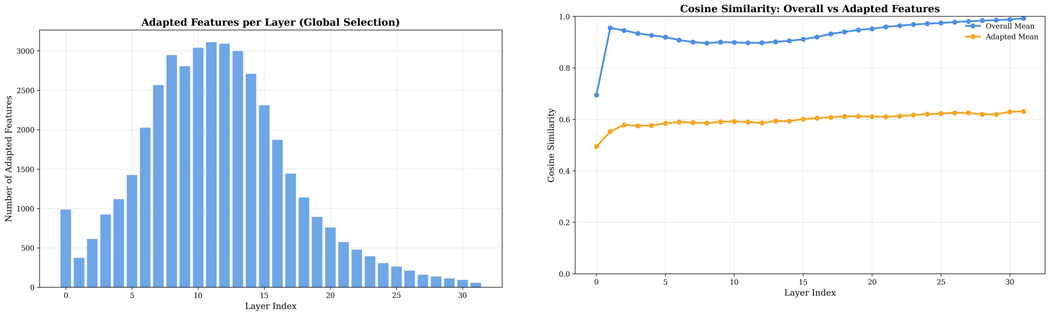


Figure 7: **Per-layer FVU across regimes.** Each panel shows the convergence of SAEs trained with different masking regimes for a specific layer. Text-only SAEs begin with slightly higher error in the shallowest layers but adapt almost immediately to near-zero reconstruction. Image-only and full-sequence SAEs converge more slowly and plateau at higher error, while random initialization performs worst throughout. This confirms that projector-driven spans remain off-distribution in early layers and only align with the LLM basis in later layers.



(a) **Adapted features per layer.** Most concentrate in mid layers, tapering in deeper blocks.

(b) **Decoder cosine by layer.** Adapted features remain less aligned to the base dictionary than the overall pool.

Figure 8: **Per-layer statistics of adapted features.** (a) Distribution of adapted feature counts across depth. (b) Mean decoder cosine similarity for adapted features vs. the overall pool.

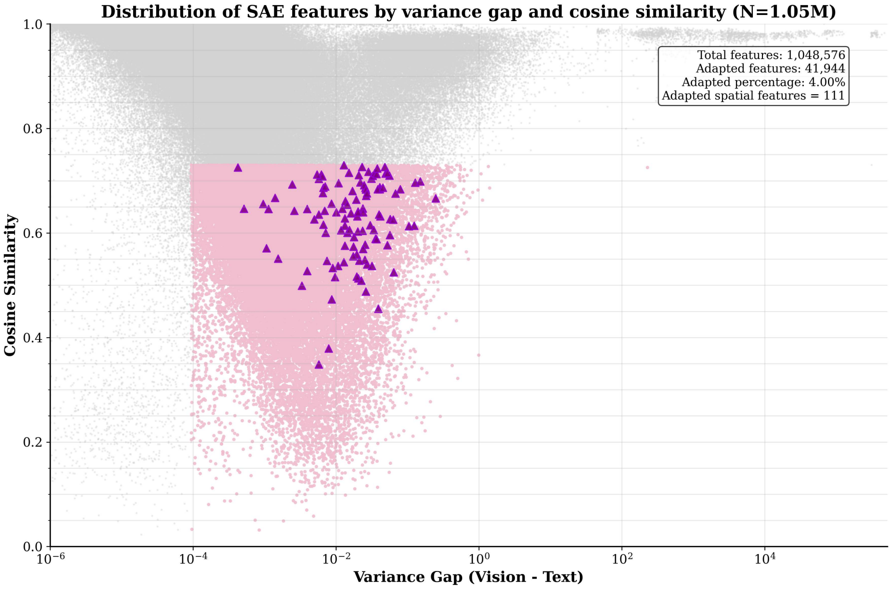
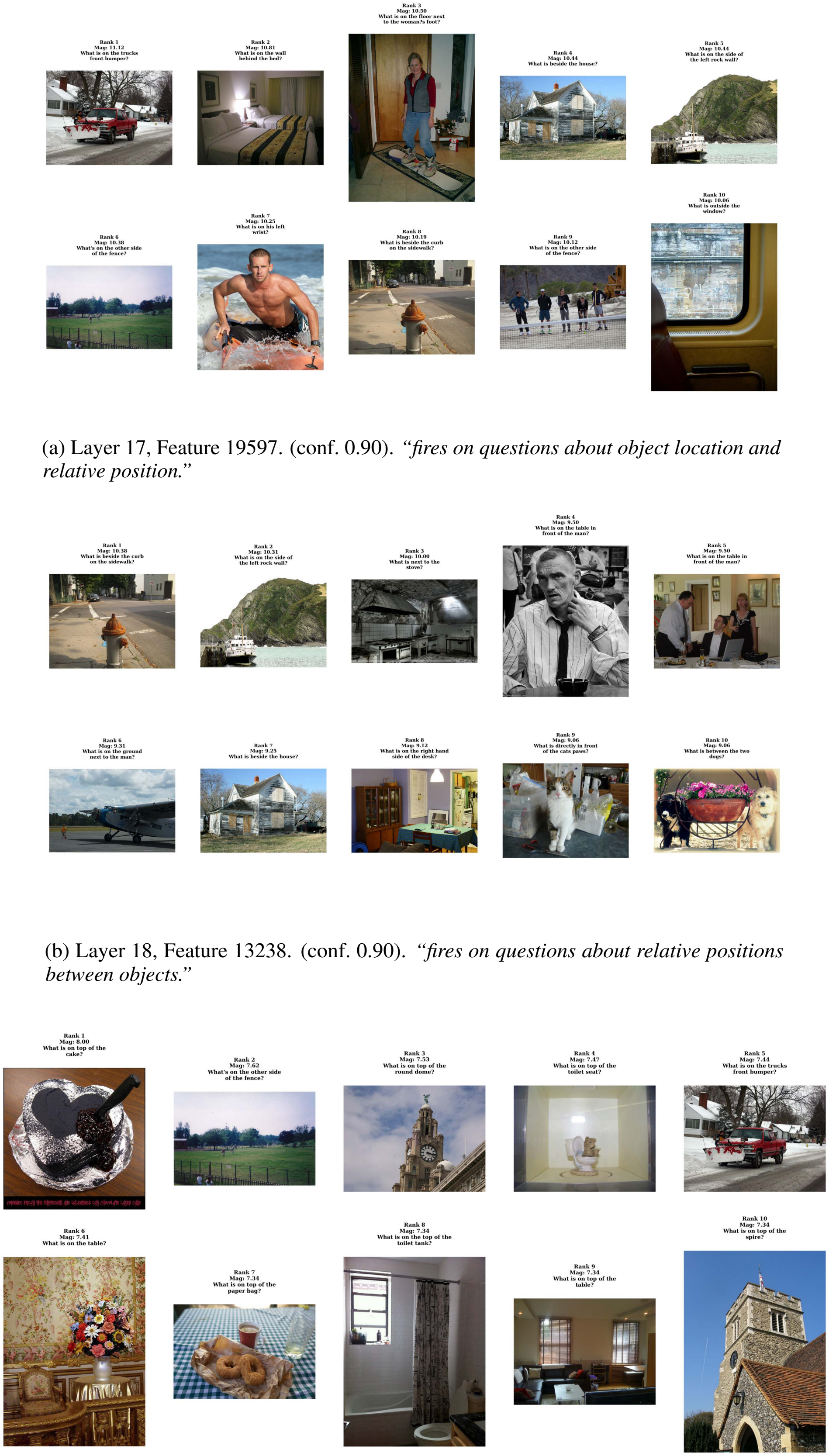


Figure 9: **Joint distribution of SAE features by variance gap and cosine similarity.** Adapted features (pink) are highlighted, with the retained spatial subset (purple) concentrated in the high-frequency tail.

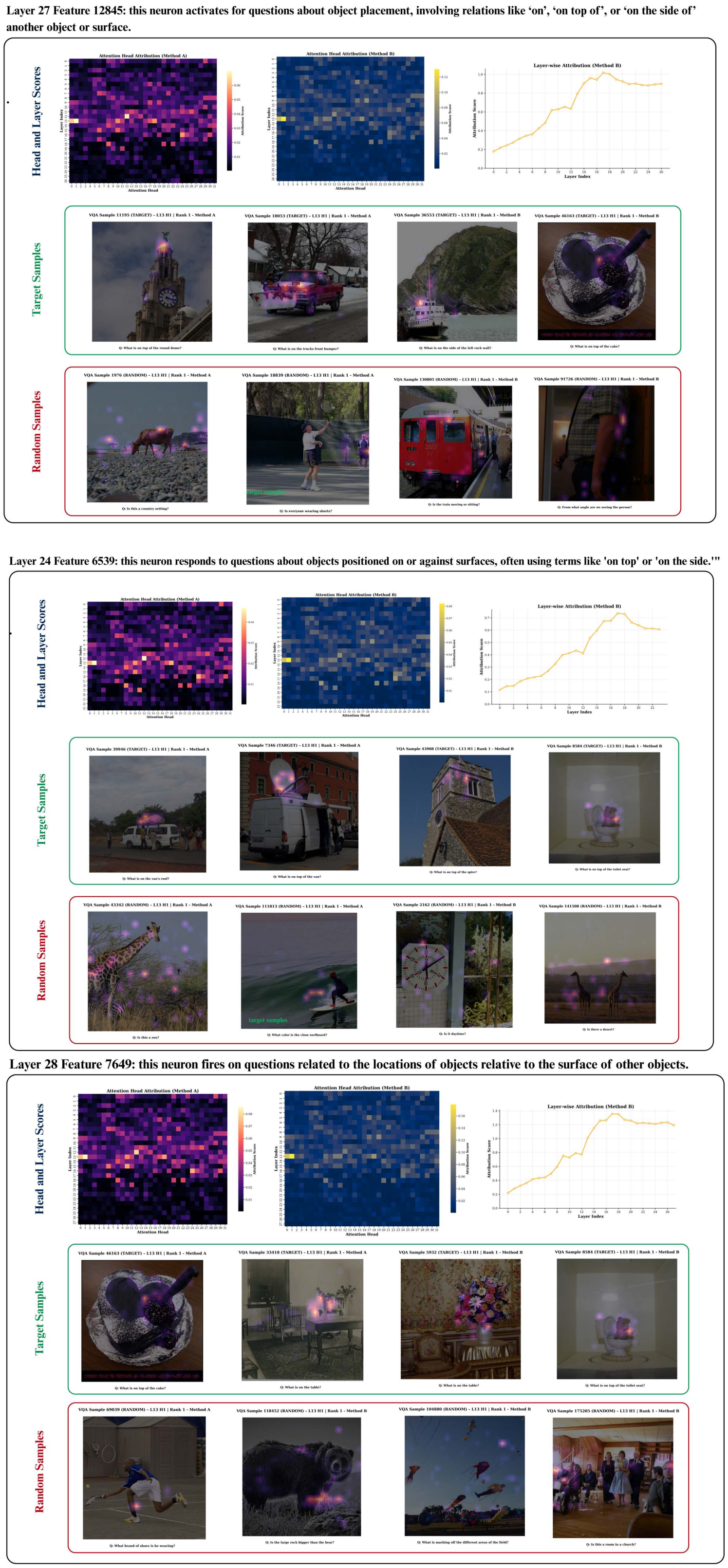

Rank 1
Mag: 11.12
What is on the trucks front bumper?
Rank 2
Mag: 10.81
What is on the wall behind the bed?
Rank 3
Mag: 10.50
What is on the floor next to the woman?s foot?
Rank 4
Mag: 10.44
What is beside the house?
Rank 5
Mag: 10.44
What is on the side of the left rock wall?
Rank 6
Mag: 10.38
What's on the other side of the fence?
Rank 7
Mag: 10.25
What is on his left wrist?
Rank 8
Mag: 10.19
What is beside the curb on the sidewalk?
Rank 9
Mag: 10.12
What is on the other side of the fence?
Rank 10
Mag: 10.06
What is outside the window?
Rank 1
Mag: 10.38
What is beside the curb on the sidewalk?
Rank 2
Mag: 10.31
What is on the side of the left rock wall?
Rank 3
Mag: 10.00
What is next to the stove?
Rank 4
Mag: 9.50
What is on the table in front of the man?
Rank 5
Mag: 9.50
What is on the table in front of the man?
Rank 6
Mag: 9.31
What is on the ground next to the man?
Rank 7
Mag: 9.25
What is beside the house?
Rank 8
Mag: 9.12
What is on the right hand side of the desk?
Rank 9
Mag: 9.06
What is directly in front of the cats paws?
Rank 10
Mag: 9.06
What is between the two dogs?
Rank 1
Mag: 8.00
What is on top of the cake?
COMBINED FRIDAY THE THIRTEENTH AND VALENTINES DAY CHOCOLATE LAYER CAKE
Rank 2
Mag: 7.62
What's on the other side of the fence?
Rank 3
Mag: 7.53
What is on top of the round dome?
Rank 4
Mag: 7.47
What is on top of the toilet seat?
Rank 5
Mag: 7.44
What is on the trucks front bumper?
Rank 6
Mag: 7.41
What is on the table?
Rank 7
Mag: 7.34
What is on top of the paper bag?
Rank 8
Mag: 7.34
What is on the top of the toilet tank?
Rank 9
Mag: 7.34
What is on top of the table?
Rank 10
Mag: 7.34
What is on top of the spire?

(a) Layer 17, Feature 19597. (conf. 0.90). *"fires on questions about object location and relative position."*

(b) Layer 18, Feature 13238. (conf. 0.90). *"fires on questions about relative positions between objects."*

(c) Layer 23, Feature 1657. (conf. 0.80). *"fires on questions about what object is located on top of or placed on another object or surface."*

Figure 10: **Qualitative Auto-Interp examples.** Top-activating VQA samples for three adapted features automatically labeled as SPATIAL, with short GPT-4o-mini–generated descriptions.



Layer 27 Feature 12845: this neuron activates for questions about object placement, involving relations like ‘on’, ‘on top of’, or ‘on the side of’ another object or surface.

Layer 24 Feature 6539: this neuron responds to questions about objects positioned on or against surfaces, often using terms like 'on top' or 'on the side.'"

Layer 28 Feature 7649: this neuron fires on questions related to the locations of objects relative to the surface of other objects.

Head and Layer Scores
Target Samples
Random Samples
Attention Head Attribution (Method A)
Attention Head Attribution (Method B)
Layer-wise Attribution (Method B)
Layer Index
Attention Head
Attribution Score
VQA Sample 11195 (TARGET) - L13 H1 | Rank 1 - Method A
Q: What is on top of the round dome?
VQA Sample 18053 (TARGET) - L13 H1 | Rank 1 - Method A
Q: What is on the trucks front bumper?
VQA Sample 36553 (TARGET) - L13 H1 | Rank 1 - Method B
Q: What is on the side of the left rock wall?
VQA Sample 46163 (TARGET) - L13 H1 | Rank 1 - Method B
Q: What is on top of the cake?
VQA Sample 1976 (RANDOM) - L13 H1 | Rank 1 - Method A
Q: Is this a country setting?
VQA Sample 18839 (RANDOM) - L13 H1 | Rank 1 - Method A
target samples
Q: Is everyone wearing shorts?
VQA Sample 130805 (RANDOM) - L13 H1 | Rank 1 - Method B
Q: Is the train moving or sitting?
VQA Sample 91726 (RANDOM) - L13 H1 | Rank 1 - Method B
Q: From what angle are we seeing the person?
VQA Sample 39946 (TARGET) - L13 H1 | Rank 1 - Method A
Q: What is on the van's roof?
VQA Sample 7346 (TARGET) - L13 H1 | Rank 1 - Method A
Q: What is on top of the van?
VQA Sample 43908 (TARGET) - L13 H1 | Rank 1 - Method B
Q: What is on top of the spire?
VQA Sample 8584 (TARGET) - L13 H1 | Rank 1 - Method B
Q: What is on top of the toilet seat?
VQA Sample 43342 (RANDOM) - L13 H1 | Rank 1 - Method A
Q: Is this a zoo?
VQA Sample 111813 (RANDOM) - L13 H1 | Rank 1 - Method A
target samples
Q: What color is the close surfboard?
VQA Sample 2162 (RANDOM) - L13 H1 | Rank 1 - Method B
Q: Is it daytime?
VQA Sample 141508 (RANDOM) - L13 H1 | Rank 1 - Method B
Q: Is there a desert?
VQA Sample 46163 (TARGET) - L13 H1 | Rank 1 - Method A
Q: What is on top of the cake?
VQA Sample 33418 (TARGET) - L13 H1 | Rank 1 - Method A
Q: What is on the table?
VQA Sample 5932 (TARGET) - L13 H1 | Rank 1 - Method B
Q: What is on the table?
VQA Sample 8584 (TARGET) - L13 H1 | Rank 1 - Method B
Q: What is on top of the toilet seat?
VQA Sample 69039 (RANDOM) - L13 H1 | Rank 1 - Method A
Q: What brand of shoes is he wearing?
VQA Sample 118452 (RANDOM) - L13 H1 | Rank 1 - Method B
Q: Is the large rock bigger than the bear?
VQA Sample 104880 (RANDOM) - L13 H1 | Rank 1 - Method B
Q: What is marking off the different areas of the field?
VQA Sample 175205 (RANDOM) - L13 H1 | Rank 1 - Method B
Q: Is this a room in a church?

Figure 11: Consistency of attribution results across related spatial features. In all three cases, the same attention head (**L13H1**) is identified as the top contributor under both methods.

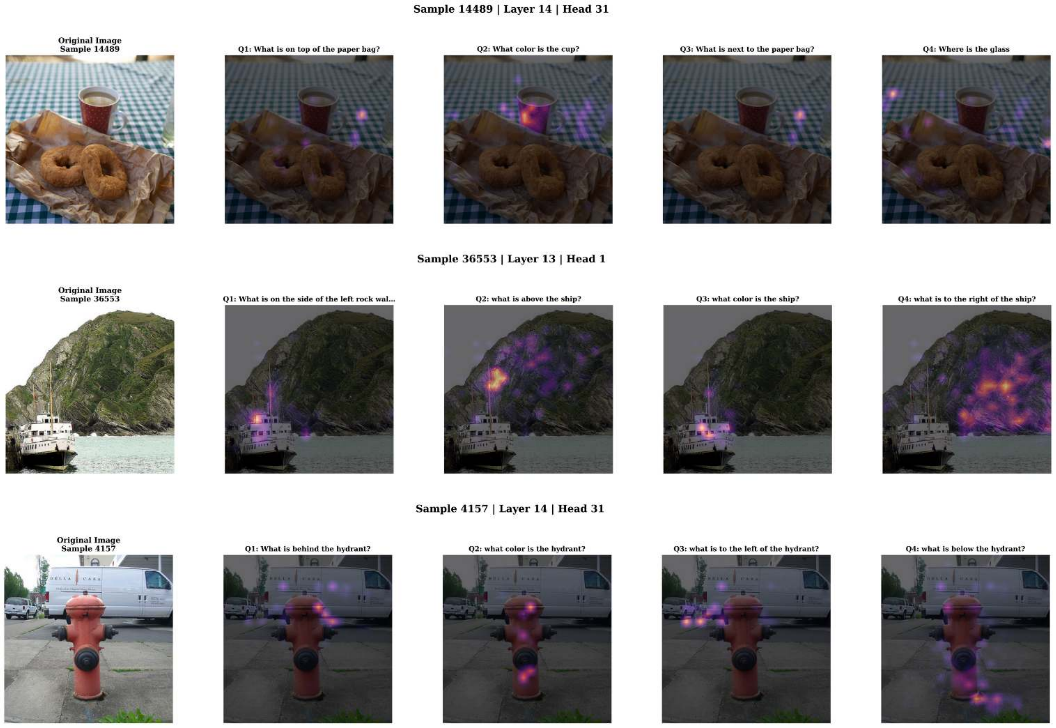

Sample 14489 | Layer 14 | Head 31
Original Image
Sample 14489
Q1: What is on top of the paper bag?
Q2: What color is the cup?
Q3: What is next to the paper bag?
Q4: Where is the glass
Sample 36553 | Layer 13 | Head 1
Original Image
Sample 36553
Q1: What is on the side of the left rock wal...
Q2: what is above the ship?
Q3: what color is the ship?
Q4: what is to the right of the ship?
Sample 4157 | Layer 14 | Head 31
Original Image
Sample 4157
Q1: What is behind the hydrant?
Q2: what color is the hydrant?
Q3: what is to the left of the hydrant?
Q4: what is below the hydrant?

Figure 12: **Attention head visualizations across queries.** Each row shows one image with attention overlays from a single high-attribution head across multiple spatial and non-spatial custom queries. The same heads consistently focus on semantically relevant regions.

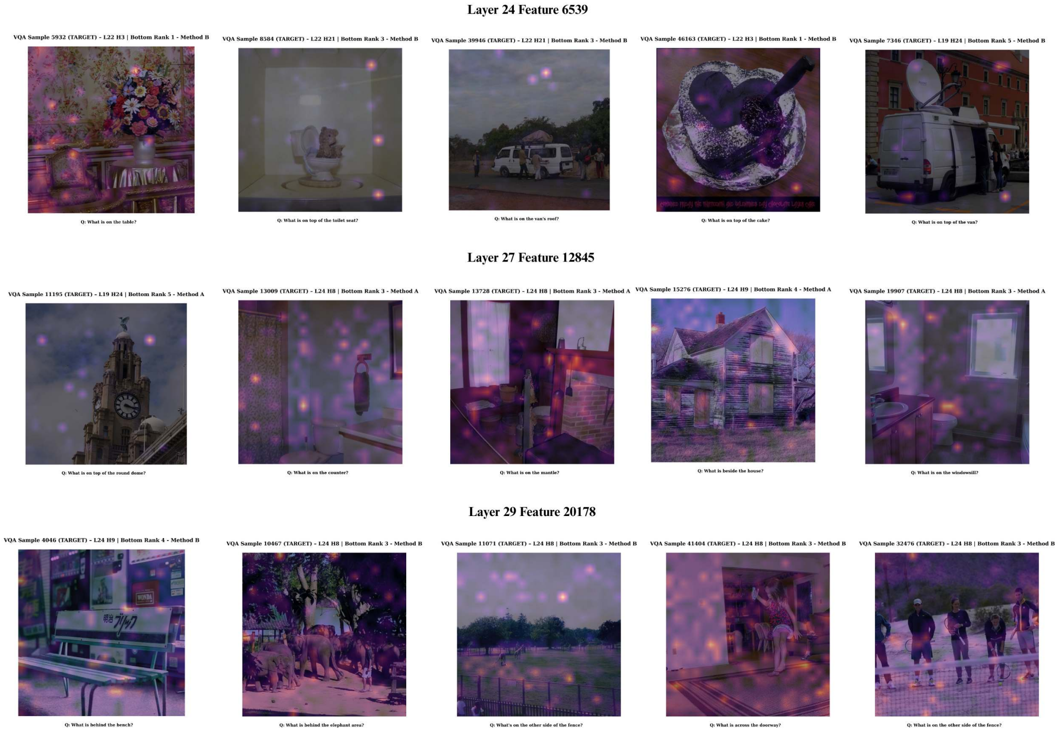

Layer 24 Feature 6539
VQA Sample 5932 (TARGET) - L22 H3 | Bottom Rank 1 - Method B
Q: What is on the table?
VQA Sample 8584 (TARGET) - L22 H21 | Bottom Rank 3 - Method B
Q: What is on top of the toilet seat?
VQA Sample 39946 (TARGET) - L22 H21 | Bottom Rank 3 - Method B
Q: What is on the van's roof?
VQA Sample 46163 (TARGET) - L22 H3 | Bottom Rank 1 - Method B
Q: What is on top of the cake?
VQA Sample 7346 (TARGET) - L19 H24 | Bottom Rank 5 - Method B
Q: What is on top of the van?
Layer 27 Feature 12845
VQA Sample 11195 (TARGET) - L19 H24 | Bottom Rank 5 - Method A
Q: What is on top of the round dome?
VQA Sample 13009 (TARGET) - L24 H8 | Bottom Rank 3 - Method A
Q: What is on the counter?
VQA Sample 13728 (TARGET) - L24 H8 | Bottom Rank 3 - Method A
Q: What is on the mantle?
VQA Sample 15276 (TARGET) - L24 H9 | Bottom Rank 4 - Method A
Q: What is beside the house?
VQA Sample 19907 (TARGET) - L24 H8 | Bottom Rank 3 - Method A
Q: What is on the windowsill?
Layer 29 Feature 20178
VQA Sample 4046 (TARGET) - L24 H9 | Bottom Rank 4 - Method B
Q: What is behind the bench?
VQA Sample 10467 (TARGET) - L24 H8 | Bottom Rank 3 - Method B
Q: What is behind the elephant area?
VQA Sample 11071 (TARGET) - L24 H8 | Bottom Rank 3 - Method B
Q: What's on the other side of the fence?
VQA Sample 41404 (TARGET) - L24 H8 | Bottom Rank 3 - Method B
Q: What is across the doorway?
VQA Sample 32476 (TARGET) - L24 H8 | Bottom Rank 3 - Method B
Q: What is on the other side of the fence?

Figure 13: **Low-attribution heads.** Bottom-ranked heads yield diffuse or irrelevant attention, showing little relation to the spatial queries.