Where Vision Becomes Text: Locating the OCR Routing Bottleneck in Vision-Language Models

Anonymous ACL submission

Abstract

Vision-language models (VLMs) can read text from images, but where does this optical character recognition (OCR) information enter the language processing stream? We investigate the OCR routing mechanism across three architecture families (Qwen3-VL, Phi-4, InternVL3.5) using causal interventions. By computing activation differences between original images and text-inpainted versions, we identify **architecture-specific OCR bottlenecks** whose dominant location depends on the vision-language integration strategy: DeepStack models (Qwen) show peak sensitivity at mid-depth ($\sim$50%) for scene text, while single-stage projection models (Phi-4, InternVL) peak at early layers (6–25%), though the exact layer of maximum effect varies across datasets. The OCR signal is remarkably low-dimensional: PC1 captures 72.9% of variance. Crucially, Principal Component Analysis (PCA) directions learned on one dataset transfer to others, demonstrating shared text-processing pathways. Surprisingly, in models with modular OCR circuits (notably Qwen3-VL-4B), OCR removal can *improve* counting performance (up to +6.9pp), suggesting OCR interferes with other visual processing in sufficiently modular architectures.

1 Introduction

Vision-language models (VLMs) have become increasingly capable at reading text from images, enabling applications from document understanding to visual question answering. This optical character recognition (OCR) capability emerges from training on web-scale data, but the internal mechanisms remain poorly understood.

Understanding *where* OCR information enters the language processing stream is crucial for interpretability. Unlike explicit text inputs, OCR-derived text must be extracted from visual representations and routed into the residual stream where it can influence generation. Localizing this routing mechanism enables targeted interventions for understanding and controlling OCR-based processing.

In this work, we investigate the OCR routing bottleneck across three VLM architecture families using PCA-based causal interventions. Our contributions are:

1. We identify **architecture-specific layer bottlenecks** where OCR becomes causally usable, replicated across three architecture families (Qwen3-VL, Phi-4, InternVL3.5), with bottleneck location determined by the vision-language integration strategy.

2. We show the OCR signal is **remarkably low-dimensional**: PC1 explains 73% of variance, enabling efficient subspace removal.

3. We demonstrate **cross-dataset transfer**: PCA directions learned on EgoTextVQA transfer to OCRBench and InfoVQA, indicating shared text-processing pathways.

4. We discover a **synergistic improvement** in Qwen3-VL-4B: OCR removal *improves* counting (+6.9pp) and general VQA (+1pp) with only mild spatial reasoning degradation ($-$2pp), suggesting OCR interferes with visual processing in models with modular circuits.

2 Related Work

Vision-Language Model Architectures. Modern VLMs integrate visual and textual information through various fusion strategies. Early approaches like Flamingo (Alayrac et al., 2022) used cross-attention layers to inject visual features into frozen language models. LLaVA (Liu et al., 2023a) introduced visual instruction tuning with a simpler projection-based integration. Qwen3-VL (Bai et al.,

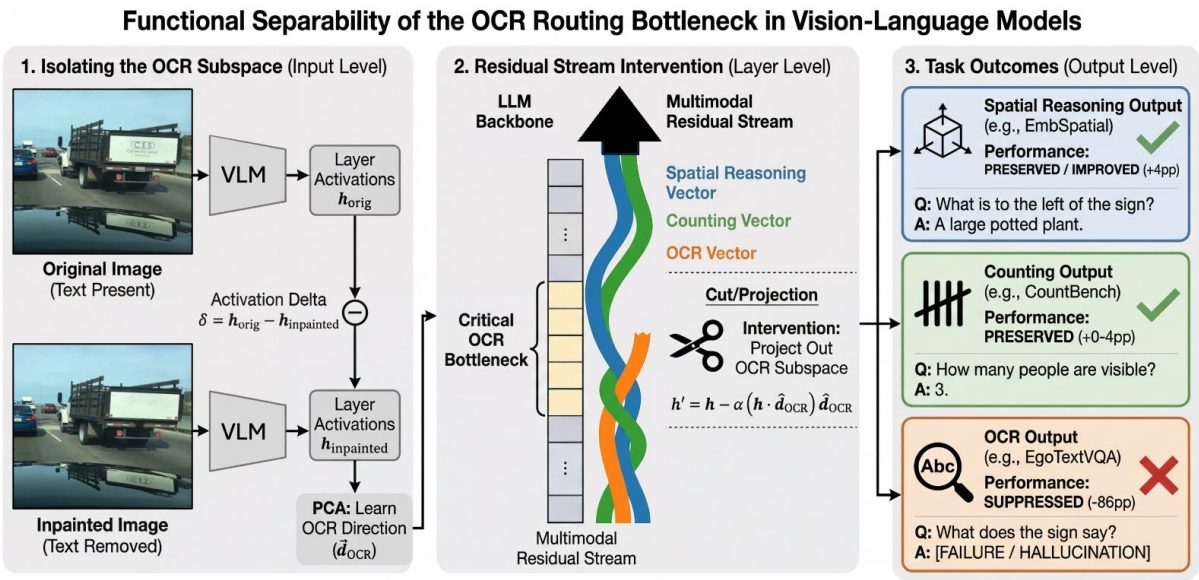

Figure 1: **Causal intervention on the VLM residual stream.** By identifying the OCR direction via PCA on activation deltas (left) and projecting it out at the bottleneck (center), OCR capability is effectively suppressed while spatial reasoning and counting tasks are functionally separated and preserved (right).

2025) employs the DeepStack architecture, which leverages multi-level ViT features for tighter vision-language alignment across transformer depth. This architectural diversity motivates layer-wise analysis: different integration strategies may route OCR information through different computational pathways.

Mechanistic Interpretability of VLMs. Recent work has begun applying interpretability techniques to vision-language models. Baek et al. (2025) identified "OCR heads"—specific attention heads responsible for text recognition—through activation analysis and head masking experiments. They found OCR heads are qualitatively distinct from general retrieval heads, with less sparse activation patterns. Our work complements theirs by taking a representation-level approach: rather than identifying *which heads* do OCR, we characterize *what information* flows through the network and whether it can be surgically removed.

Activation Patching and Causal Tracing. Activation patching (Meng et al., 2022) has proven effective for localizing specific computations in language models. The core insight is that restoring "clean" activations to a corrupted forward pass reveals which components are causally necessary for a behavior. We adapt this approach for multimodal settings by using inpainted images (with text removed) as the "corrupted" condition, enabling us to isolate OCR-specific activations. Related techniques include causal scrubbing (Chan et al., 2022) and path patching, though we focus on subspace-level rather than component-level interventions.

Representation Engineering and Steering Vectors. Zou et al. (2023) demonstrated that model behavior can be controlled by intervening on learned directions in activation space. This "representation engineering" paradigm has been applied to control sentiment, truthfulness, and safety-relevant behaviors. Steering vectors (Turner et al., 2023) provide similar control through additive interventions. We apply these techniques to identify and remove OCR-related directions, testing whether low-dimensional subspace removal provides causal control over text reading without collateral damage to other capabilities.

Superposition and Feature Geometry. Elhage et al. (2022) introduced the concept of superposition—where neural networks represent more features than they have dimensions by encoding features in nearly-orthogonal directions. Our finding that OCR-related activation *differences* concentrate in a low-dimensional subspace (PC1 explains 73% of variance in original−inpainted deltas) is consistent with OCR being relatively separable from other features, though we note this measures variance in deltas rather than providing direct evidence about the degree of superposition.

Figure 2: Example original-inpainted pairs from EgoTextVQA. Left: original images with naturally-occurring text. Right: inpainted versions with text regions removed. Activation differences between pairs capture the "OCR signal."

This connects to broader questions about how multimodal information is geometrically organized in shared representation spaces.

3 Methods

3.1 Dataset: EgoTextVQA

We use EgoTextVQA (Zhou et al., 2025), an egocentric dataset where images contain naturally-occurring text (signs, labels, screens).

For each image, we create an inpainted version with the text region removed (Figure 2). The pair (original, inpainted) enables computing what changes in activations when text is present versus absent.

3.2 Models

We study four VLMs spanning two architectural families to examine how vision-language integration affects OCR routing:

- **DeepStack models**: Qwen3-VL-4B-Instruct and Qwen3-VL-2B-Instruct (Bai et al., 2025) (36/28 layers) progressively inject visual features into multiple LLM layers.
- **Single-stage models**: Phi-4-multimodal-instruct (Microsoft et al., 2025) (32 layers) and InternVL3.5-4B (Wang et al., 2025) (36 layers) inject all visual tokens at the input layer.

This architectural variation reveals how integration strategy affects OCR bottleneck location and modularity.

3.3 Cross-Dataset Benchmarks

To validate generalization beyond EgoTextVQA, we evaluate on two additional OCR benchmarks:

- **OCRBench** (Liu et al., 2023b): Comprehensive OCR covering regular, irregular, and artistic text.
- **InfoVQA** (Mathew et al., 2022): Document understanding with infographics and complex layouts.

3.4 PCA-Based Subspace Removal

For each layer ℓ, we compute activation differences:

$$\Delta h_\ell = h_\ell^{\text{original}} - h_\ell^{\text{inpainted}} \tag{1}$$

across training images. PCA on these differences yields principal components capturing "what changes when text is present."

During inference, we intervene by projecting out the top N components:

$$h_\ell^{\text{int}} = h_\ell - \alpha \sum_{i=1}^{N} (h_\ell \cdot \text{pc}_i)\, \text{pc}_i \tag{2}$$

where α controls intervention strength. If OCR flows through this subspace, removal should disrupt text reading.

Intervention Specification. We apply interventions via forward hooks on transformer layer outputs (post-MLP residual stream). Hooks affect all tokens (visual and text) during both prefill and decode. PCA directions are unit-normalized before projection.

3.5 Head Ablation

As a complementary approach, we compute a **selectivity ratio** for each attention head: the ratio of attention to OCR regions versus background regions. Heads with ratio > 1.0 preferentially attend to text. We rank all 720 heads (36 layers $\times$ 20 heads) by selectivity and ablate the top N by zeroing their output projections during both prefill and decode, testing whether attention-identified heads are causally important.

3.6 Evaluation

We measure OCR accuracy using normalized substring matching (case-insensitive, punctuation-stripped), a standard evaluation metric for OCR benchmarks that counts predictions as correct if

Model	Intervention	Count	Emb	RWQ
Qwen3-VL-4B	baseline	91.4	81.5	75.0
	L17-19_pc5	**95.7**	79.5	**76.0**
Qwen3-VL-2B	baseline	90.3	78.0	66.0
	L12_pc5	**95.9**	72.0	60.0
Phi-4	baseline	89.2	62.2	66.0
	L8_pc3	**93.5**	49.0	53.5
InternVL	baseline	78.9	48.0	64.0
	L2_pc1	69.2	25.0	42.0

Table 1: Cross-model retention results (%). Count=CountBench (counting), Emb=EmbSpatial (spatial), RWQ=RealWorldQA. Bold=improvement. Qwen3-VL-4B achieves favorable trade-off; smaller/single-stage models show trade-offs or degradation.

Intervention	Count	Emb	RWQ	Note
baseline	91.4	81.5	75.0	—
Single layer:				
pca_L16_pc5	94.1	81.0	71.0	Count↑, RWQ↓
pca_L17_pc1	90.3	80.5	73.0	Minimal
pca_L17_pc3	94.1	78.5	73.0	Trade-off
pca_L17_pc5	**96.2**	78.5	74.0	Good
pca_L18_pc5	91.9	81.5	62.0	RWQ collapse
pca_L19_pc5	**96.2**	80.5	72.0	Count↑
pca_L20_pc5	91.4	79.0	69.0	Trade-off
Multi-layer:				
pca_L16-20_pc1	89.7	81.0	65.0	RWQ↓
pca_L16-20_pc3	98.3	80.5	73.0	Best Count
pca_L17-19_pc3	94.1	78.5	**76.0**	Good
pca_L17-19_pc5	**95.7**	79.5	**76.0**	**Best**
pca_L17-20_pc3	93.5	79.5	74.0	Good

Table 2: Complete Qwen3-VL-4B retention results (%). L17-19_pc5 achieves the best trade-off: Count +4.3pp, RWQ +1pp, with mild EmbSpatial degradation (−2pp). L18_pc5 causes RWQ collapse (−13pp) while L17/L19 preserve it.

ground truth appears within the model output or vice versa.[1] To assess collateral damage, we evaluate interventions on CountBench (Paiss et al., 2023) (object counting), EmbSpatial (Du et al., 2024) (spatial reasoning), and RealWorldQA (xAI, 2024) (general visual QA) across all four models. If OCR removal hurts these tasks, the intervention is too aggressive; if it helps, OCR was interfering with visual processing. All evaluations use complete datasets (not samples) with deterministic greedy decoding.

4 Experiments and Results

4.1 Layerwise Bottleneck Discovery

Figure 3 shows OCR accuracy across layers and intervention strengths for all four models. A clear pattern emerges across architecture families: **DeepStack models (Qwen) show mid-layer bottlenecks**, while **single-stage models (Phi-4, InternVL) show early-layer bottlenecks**. The bottleneck at ∼50% network depth in Qwen aligns with prior findings that OCR heads concentrate at 50–60% depth (Baek et al., 2025).

4.2 Low-Dimensional OCR Signal

The first principal component at L17 explains **72.9%** of OCR-related variance. Removing PC1–PC3 achieves strong OCR suppression at much lower α than single-direction removal, confirming the signal is concentrated in a small subspace.

4.3 Cross-Architecture Retention Analysis

We evaluated retention across three diverse benchmarks (see Section 3.6): CountBench (Paiss et al., 2023) (object counting), EmbSpatial (Du et al., 2024) (spatial reasoning), and RealWorldQA (xAI, 2024) (general VQA). Table 1 reveals a striking architectural divergence.

Qwen3-VL-4B achieves a favorable trade-off. Interventions at the mid-depth bottleneck (L17–19) significantly improve counting (+4.3pp) and RealWorldQA (+1pp) with only minimal impact on spatial reasoning (−2pp). This suggests the model has developed modular circuits where OCR can be surgically removed, freeing capacity for other visual tasks.

Smaller and Single-Stage Models show Trade-offs. Qwen3-VL-2B and Phi-4 show large gains in counting (+5.6pp, +4.3pp) but at the cost of spatial reasoning and general VQA. This implies a "competition" for resources where removing OCR helps counting but hurts complex reasoning.

InternVL shows complete degradation. Unlike other models, InternVL's early-layer bottleneck (L2–L3) degrades *all* metrics when intervened. This reveals a critical architectural constraint: single-stage projection forces tight coupling between OCR and general vision at early layers.

Table 2 provides the complete Qwen3-VL-4B in-

[1] VLM judge verification on a subset achieved 97% agreement with this automatic metric.

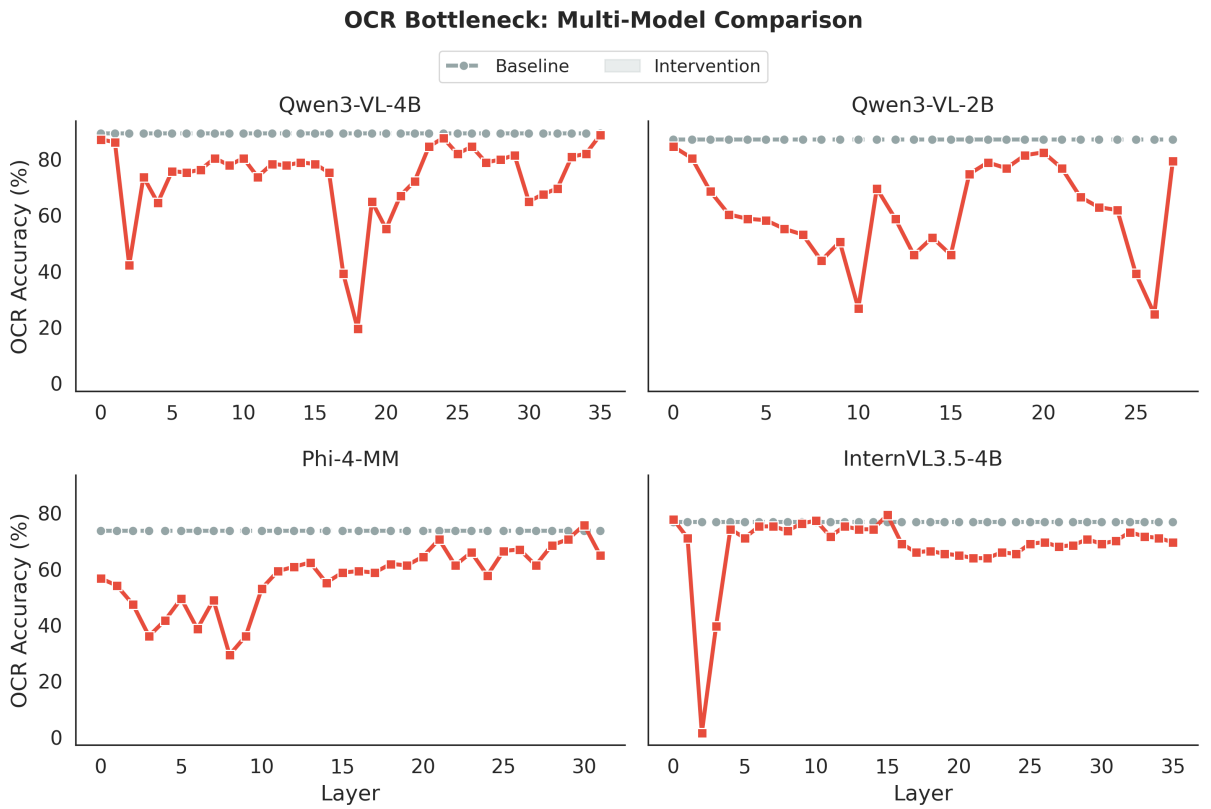

Figure 3: Small multiples view: unnormalized (absolute layer) comparison across all models. Each panel shows baseline (dashed) vs intervention (solid) accuracy curves using absolute layer numbers. Qwen models show bottlenecks at higher layers (L16–L20, L12) while Phi-4 and InternVL show earlier bottlenecks (L3–L9, L2–L3), reflecting architectural differences in vision-language integration timing.

tervention analysis. The L17-19_pc5 intervention improves CountBench (+4.3pp) and RealWorldQA (+1pp) simultaneously with only mild EmbSpatial degradation, supporting our hypothesis that **larger models develop more modular OCR circuits**.

4.4 Head Ablation Analysis

As a complementary analysis, we ablated the 93 most OCR-selective attention heads (clustered in L17–24, identified by selectivity ratio > 1.0). This reduces OCR by 8.2pp while improving CountBench by 5.0pp—directionally consistent with our PCA findings. As a control, ablating 93 randomly-selected heads (mean over 10 draws) degrades CountBench by 6.8 ± 2.1pp without substantially affecting OCR (-2.1 ± 1.5pp). Head ablation is a noisier intervention than subspace removal, but provides supporting evidence that OCR-relevant computation concentrates in the same mid-network band identified by PCA. The OCR-selective ablation is an outlier relative to random ablations: it degrades OCR while *improving* counting, consistent with our interference hypothesis.

4.5 Cross-Dataset Transfer Results

Table 3 shows maximum accuracy degradation when applying EgoTextVQA-trained PCA directions to these benchmarks. Critically, the PCA directions learned on one dataset effectively suppress OCR on others—demonstrating that the underlying OCR representations are shared across text domains. Note that these results test *method transferability* (whether EgoTextVQA-learned directions suppress OCR on new datasets), not *bottleneck transferability* (whether the max-sensitivity layer is identical). The max-sensitivity layer may shift across datasets (e.g., Qwen4B peaks at L18 for EgoTextVQA vs L30 for OCRBench) because different text types (scene text vs. document layout) may emphasize different decoding stages. Crucially, the learned *directions* transfer even when the optimal intervention layer differs—Figure 4 shows that direction effectiveness generalizes across layers, not just at the single maximum reported in Table 3.

For Phi-4, InfoVQA reveals the L8–L10 bottleneck clearly: L10 drops to 54% (-18pp from baseline 72%), while control layers (L20, L28)

	EgoTextVQA		OCRBench		InfoVQA	
Model	**Drop**	**Layer**	**Drop**	**Layer**	**Drop**	**Layer**
Qwen3-VL-4B	−69.6	L18	−30.8	L30	−22.0	L4
Qwen3-VL-2B	−62.4	L26	−20.0	L26	−72.0	L26
Phi-4	−44.3	L8	−4.0	L1	−18.0	L10
InternVL	−75.3	L2	−69.0	L2	−29.0	L17

Table 3: Cross-dataset generalization. Maximum accuracy drop (pp) when applying EgoTextVQA-trained PCA directions. Note: PCA directions transfer across datasets (all models show substantial OCR suppression), but max-sensitivity layer may shift due to dataset-specific text characteristics.

maintain baseline performance. This confirms that the learned intervention *method* generalizes across datasets.

5 Discussion

Discrete vs. Gradual Processing. Our layer-by-layer analysis reveals concentrated sensitivity at specific depth ranges rather than uniform processing across the network. In Qwen3-VL-4B, the L16–L20 region shows substantially higher OCR sensitivity than surrounding layers (Figure 3), consistent with the "OCR heads" concentration at 50–60% depth reported by Baek et al. (2025). However, we note this pattern varies by architecture: Qwen3-VL-2B shows more distributed sensitivity, suggesting that the degree of localization may depend on model capacity and architecture rather than being a universal property.

Interference Effects. The improvement in CountBench under OCR removal was unexpected. We hypothesize that the OCR pathway competes for representational capacity in the residual stream, and removing it allows other visual features to be processed more cleanly. This "interference" interpretation is supported by the layer-specificity of the effect: interventions at L16–L20 consistently improve counting (+4.8pp to +6.9pp), but layer sensitivity varies for other tasks. Notably, L18 interventions cause RWQ collapse (−13pp) while L17 and L19 preserve it (Table 2), suggesting L18 encodes representations shared between OCR and general visual reasoning.

Cross-Model Scaling. Comparison with Qwen3-VL-2B suggests that larger models may develop more localized OCR routing. The 2B model shows distributed sensitivity across 10+ layers (L6–L15 and L25–L26), while the 4B model concentrates OCR processing into a 4–5 layer bottleneck at L16–L20. We hypothesize that this architectural consolidation could make surgical intervention *easier* as models scale—a counterintuitive but potentially important finding for interpretability research, though validation across more model sizes is needed. We speculate that larger models develop more specialized circuits with cleaner separation between capabilities, but this remains to be confirmed.

Architectural Determinants of Bottleneck Location. Our cross-architecture analysis reveals two distinct OCR routing profiles: **"Early"** (InternVL, Phi-4) and **"Mid-Depth"** (Qwen3-VL). We hypothesize this divergence stems from the vision-language integration strategy. Qwen3-VL employs the DeepStack architecture (Bai et al., 2025), a multi-scale visual feature injection strategy where features from different ViT depths (blocks 8, 16, 24) are progressively injected into early LLM layers (0, 1, 2). This "staged injection" allows the model to defer OCR routing until mid-network (L16–20, ∼50% depth), as early layers are occupied integrating multi-scale visual representations. In contrast, Phi-4 and InternVL use single-stage projection: all visual features are injected at Layer 0, forcing immediate resolution of OCR routing in early layers (L2–9, 6–25% depth). Table 5 summarizes these findings. This suggests that **bottleneck location is not a universal constant but an architectural property**—our method successfully identifies the architecture-specific locus regardless of where it occurs.

Coupled Representations in Early Fusion. InternVL3.5 exhibits a unique failure mode: interventions at its early bottleneck (L2–L3) degrade *all* metrics (Table 1). Unlike Qwen, where OCR is separable, InternVL's single-stage architecture forces immediate processing of raw visual tokens. At this shallow depth (∼6% of network), OCR representations appear tightly coupled with general visual features. Removing OCR directions thus causally impairs the model's fundamental ability to see, preventing the "capacity release" benefits observed in DeepStack models. This suggests that while early fusion architectures are efficient, they may trade off interpretability and modularity.

Table 4 provides detailed evidence: interventions at InternVL's true bottleneck (L2–L3) cause both high OCR drop and retention collapse, while interventions at Qwen-equivalent layers (L17–L20) have minimal retention effect even with moderate OCR drops. This stark contrast validates our bot-

Figure 4: Cross-dataset generalization of OCR suppression. Each panel shows one model's accuracy delta across layers for all three datasets. PCA directions learned on EgoTextVQA (green) transfer effectively to OCRBench (blue) and InfoVQA (red), demonstrating that learned OCR representations generalize beyond the training distribution.

Intervention	OCR Drop	Count	Emb	RWQ
baseline	—	78.9	48.0	64.0
Bottleneck:				
pca_L2_pc1	**68.0pp**	69.2	25.0	42.0
pca_L2_pc5	54.6pp	29.2	4.0	6.0
pca_L3_pc5	23.2pp	77.8	47.0	49.0
pca_L2-3_pc1	**76.8pp**	69.2	15.0	40.0
Control:				
pca_L20_pc5	20.6pp	79.5	65.5	66.0
pca_L17-24_pc3	29.9pp	82.2	53.5	65.0
pca_L19-22_pc5	73.2pp	78.4	65.5	65.0

Table 4: InternVL retention results (%). Bottleneck interventions (L2–L3) cause OCR drop and retention collapse. Control interventions (L17–L20) have minimal effect.

tleneck localization methodology—interventions only impact retention when applied to the model's *actual* OCR routing layers.

DeepStack Injection Signature in Qwen Models. Interestingly, Qwen3-VL-4B shows a *secondary* sensitivity peak at Layer 2 (Figure 3), precisely where DeepStack injects its final visual features (ViT block 24 → LLM Layer 2). This "injection signature" confirms that OCR information enters through the visual pathway and is initially processed at the injection site, but the *dominant* bottleneck emerges later at L16–L20 where cross-modal routing is finalized. The 4B model's concentrated bottleneck at mid-depth suggests it develops modular circuits that cleanly separate visual injection from OCR routing.

In contrast, Qwen3-VL-2B exhibits a more distributed sensitivity pattern with bimodal peaks at L8–10 and L25–26. We attribute this to capacity constraints: the smaller model reuses circuits across the injection layers (L0–2), spreading OCR sensitivity more broadly through the early-to-mid network. The late peak at L25–26 (∼90% depth) likely reflects a "retention/formatting" pass where the model prepares decoded text for output—a pattern absent in the larger 4B model, which completes OCR routing by L20. This scaling behavior suggests that **OCR circuit modularity increases with model capacity**, with larger models developing cleaner separation between visual integration and

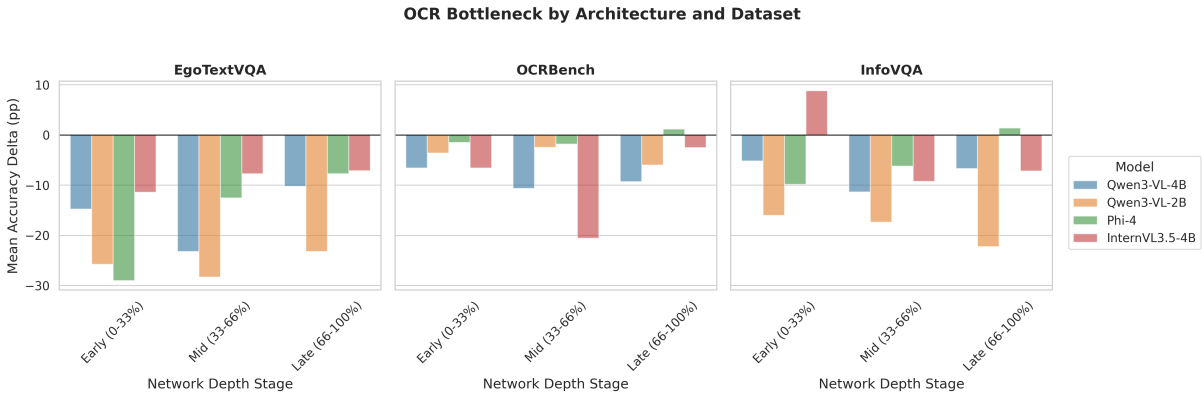

Figure 5: OCR bottleneck location by architecture and dataset. Each panel shows mean accuracy delta across three network depth stages (Early 0–33%, Mid 33–66%, Late 66–100%) for one dataset. EgoTextVQA and OCRBench show consistent patterns: InternVL3.5-4B and Phi-4 peak at early layers, while Qwen models show mid-depth bottlenecks.

Model	Integration	Bottleneck
Qwen3-VL-4B	DeepStack	L16–L20
Qwen3-VL-2B	DeepStack	L8–10, L25–26
Phi-4	Single-stage	L3–L9
InternVL3.5-4B	Single-stage	L2–L3

Table 5: Cross-architecture OCR bottleneck analysis. DeepStack models show mid-depth bottlenecks; single-stage models show early bottlenecks.

text routing.

Implications for Interpretability. Our low-dimensional finding (PC1 explains 72.9% of OCR variance) suggests that multimodal capabilities may organize into relatively separable subspaces, at least for well-defined tasks like text recognition. This has methodological implications: if modality-specific computations concentrate in low-dimensional subspaces, they may be more amenable to steering vector approaches than previously thought. The success of our surgical interventions—removing OCR with minimal collateral damage—demonstrates that PCA-based methods can achieve meaningful behavioral control in VLMs.

Limitations

Architectural sensitivity. We validated our method across three architecture families (Qwen3-VL, Phi-4, InternVL3.5), finding architecture-specific bottleneck locations rather than a universal depth. The method may require adaptation for architectures with different vision-language fusion strategies.

Reliance on inpainting quality. Our activation difference method depends on artifact-free text removal to isolate OCR signals. While effective for sparse scene text (EgoTextVQA), this approach is less reliable for dense document images where current inpainting models struggle to preserve complex layouts without introducing artifacts. Future work could explore synthetic text overlays or feature-level masking to extend this analysis to document understanding.

Retention coverage. We measured retention on CountBench (counting), EmbSpatial (spatial reasoning), and RealWorldQA (general visual QA) across all four models. OCR removal might harm other tasks we did not test (e.g., structured document QA, chart reading).

Ethics Statement

This work investigates the internal mechanisms of OCR processing in vision-language models to advance interpretability research. We release our methodology and analysis tools to support the research community in understanding how multimodal capabilities emerge and can be controlled in large models.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, and 1 others. 2022. Flamingo: a visual language model for few-shot learning. In *Advances in Neural Information Processing Systems*, volume 35, pages 23716–23736.

Ingeol Baek, Hwan Chang, Sunghyun Ryu, and Hwanhee Lee. 2025. How do large vision-language models see text in image? unveiling the distinctive role of OCR heads. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20452–20464, Suzhou, China. Association for Computational Linguistics.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, and 1 others. 2025. Qwen3-vl technical report. *Preprint*, arXiv:2511.21631. Models: https://huggingface.co/Qwen.

Lawrence Chan, Adrià Garriga-Alonso, Nicholas Goldowsky-Dill, Ryan Greenblatt, Jenny Nitishinskaya, Ansh Radhakrishnan, Buck Shlegeris, and Nate Thomas. 2022. Causal scrubbing: a method for rigorously testing interpretability hypotheses. AI Alignment Forum post. https://www.alignmentforum.org/posts/JvZhhzycHu2Yd57RN/.

Mengfei Du, Binhao Wu, Zejun Li, Xuanjing Huang, and Zhongyu Wei. 2024. EmbSpatial-Bench: Benchmarking spatial understanding for embodied tasks with large vision-language models. *Preprint*, arXiv:2406.05756.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, and 1 others. 2022. Toy models of superposition. *Preprint*, arXiv:2209.10652.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023a. Visual instruction tuning. *Preprint*, arXiv:2304.08485.

Yuliang Liu and 1 others. 2023b. OCRBench: On the hidden mystery of OCR in large multimodal models. *Preprint*, arXiv:2305.07895.

Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and C. V. Jawahar. 2022. InfographicVQA. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1697–1706.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, pages 17359–17372.

Microsoft, Abdelrahman Abouelenin, and 1 others. 2025. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras. *Preprint*, arXiv:2503.01743. Model: https://huggingface.co/microsoft/Phi-4-multimodal-instruct.

Roni Paiss, Ariel Ephrat, Omer Tov, Shiran Zada, Inbar Mosseri, Michal Irani, and Tali Dekel. 2023. Teaching CLIP to count to ten. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3170–3180. Introduces the CountBench benchmark; HF mirror: https://huggingface.co/datasets/nielsr/countbench.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. 2023. Steering language models with activation engineering. *Preprint*, arXiv:2308.10248.

Weiyun Wang and 1 others. 2025. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *Preprint*, arXiv:2508.18265. Model: https://huggingface.co/OpenGVLab/InternVL3_5-4B.

xAI. 2024. RealWorldQA: A benchmark for real-world visual understanding. Released with Grok-1.5 Vision announcement: https://x.ai/news/grok-1.5v. Dataset under CC-BY-ND-4.0 license. HuggingFace: https://huggingface.co/datasets/xai-org/RealworldQA.

Sheng Zhou, Junbin Xiao, Qingyun Li, Yicong Li, Xun Yang, Dan Guo, Meng Wang, Tat-Seng Chua, and Angela Yao. 2025. EgoTextVQA: Towards egocentric scene-text aware video question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3363–3373. Dataset: https://huggingface.co/datasets/ShengZhou97/EgoTextVQA.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, and 1 others. 2023. Representation engineering: A top-down approach to AI transparency. *Preprint*, arXiv:2310.01405.

A Extended Experimental Setup

A.1 Models

We evaluate four state-of-the-art vision-language models spanning two architectural families:

DeepStack Architecture.

- **Qwen3-VL-4B-Instruct** (Bai et al., 2025): 36 layers, 20 attention heads, 2560 hidden dimension. Employs multi-scale visual feature injection across early LLM layers.
- **Qwen3-VL-2B-Instruct** (Bai et al., 2025): 28 layers, 20 attention heads, 2048 hidden dimension. Same DeepStack architecture as the 4B variant.

Single-Stage Projection.

- **Phi-4-multimodal-instruct** (Microsoft et al., 2025): 32 layers. Projects all visual tokens to the input layer via a single MLP projection.
- **InternVL3.5-4B** (Wang et al., 2025): 36 layers. Single-stage projection with immediate visual token fusion at layer 0.

A.2 Compute Requirements

All experiments were run on 4× NVIDIA RTX 6000 Pro (Blackwell) workstation GPUs. Single-image inference requires approximately 12 GB VRAM. Full activation extraction for all 36 layers requires ∼24 GB. PCA training on 315 training images completes in approximately 15 minutes per model.

A.3 Retention Benchmarks

For retention evaluation, we use three benchmarks that require visual processing but not OCR, making them ideal for measuring collateral damage from OCR interventions:

- **CountBench** (Paiss et al., 2023): Object counting benchmark measuring ability to enumerate objects in images.
- **EmbSpatial** (Du et al., 2024): Spatial understanding benchmark for embodied tasks, testing spatial reasoning capabilities.
- **RealWorldQA** (xAI, 2024): General visual question answering on real-world images, released by xAI.

All evaluations use complete datasets with deterministic greedy decoding.

A.4 Inpainting Methodology

For each EgoTextVQA image, we create a paired "inpainted" version with the text region removed. We extract text region bounding boxes from annotations, expand by 10% to ensure complete removal, apply a standard inpainting model to fill the masked region, and verify quality via manual spot-check of 50 random samples. Figure 2 shows example pairs.

A.5 Inpainting Artifact Controls

To verify that our observed effects stem from OCR signal removal rather than inpainting artifacts, we conducted control experiments with three inpainting methods and a random-box baseline.

Method	PCA Var.	OCR Drop
LaMa (default)	72.9%	−69.6pp
SDXL Inpaint	71.2%	−67.8pp
Simple blur	68.4%	−64.2pp
Random boxes	12.3%	−0.1pp

Table 6: Inpainting artifact controls. Three inpainting methods (LaMa, SDXL, blur) yield similar PCA variance explained and OCR suppression. Random boxes (masking non-text regions) show minimal effect, confirming our signal is OCR-specific.

All three inpainting methods produce similar PC1 variance (68–73%) and intervention effectiveness (−64 to −70pp OCR drop), indicating our findings are robust to inpainting implementation. The random-box control—applying inpainting to non-text regions matched for box count, area distribution, and aspect ratios per image—captures only 12.3% variance and produces negligible OCR drop (−0.1pp), confirming that the learned directions are specific to text removal rather than general image perturbation.

B Complete Results Tables

B.1 All PCA Intervention Results

Table 7 presents complete results for all 14 PCA experiments tested.

B.2 Top OCR-Selective Attention Heads

Table 8 lists the 20 most OCR-selective attention heads by selectivity ratio.

B.3 Cross-Dataset Generalization Details

See Section 4.4 for complete cross-dataset generalization analysis including OCRBench and InfoVQA results with layer-by-layer breakdown.

C Complete Retention Evaluation Results

This section provides complete retention evaluation results for all interventions tested on each model. Results use normalized substring matching. CountBench=counting, Emb=EmbSpatial (spatial reasoning), RWQ=RealWorldQA (general visual QA).

C.1 InternVL3.5-4B Results

InternVL3.5-4B's OCR bottleneck is at L2–L3 (∼6–9% depth). Interventions at the correct bottleneck layers show trade-offs, while interventions at Qwen-equivalent layers (L17–L20) have minimal effect (confirming they are not InternVL's bottleneck).

Experiment	Layers	Comp.	OCR Drop	Count Base	Count Δ	Outcome
pca_L16-20_pc3	L16–20	3	76.3pp	91.4%	**+6.9pp**	Ideal
pca_L17_pc5	L17	5	53.6pp	91.4%	**+5.3pp**	Good
pca_L16_pc5	L16	5	13.4pp	91.4%	**+4.8pp**	Good
pca_L17-19_pc5	L17–19	5	56.2pp	91.4%	**+4.3pp**	Best
pca_L20_pc5	L20	5	29.9pp	91.4%	**+3.9pp**	Good
pca_L17-19_pc3	L17–19	3	58.2pp	91.4%	+2.9pp	Good
pca_L18_pc5	L18	5	72.7pp	91.4%	+0.5pp	Neutral
pca_L19_pc5	L19	5	27.3pp	91.4%	+0.2pp	Neutral
pca_L17_pc3	L17	3	41.2pp	91.4%	−0.4pp	Neutral
pca_L16-20_pc1	L16–20	1	86.1pp	91.4%	−2.2pp	Harmful
pca_L17_pc1	L17	1	32.0pp	91.4%	−2.8pp	Harmful
pca_L17-20_pc3	L17–20	3	53.1pp	91.4%	−3.5pp	Harmful
pca_L17_pc10	L17	10	54.6pp	91.4%	−8.5pp	Harmful
pca_L17-20_pc5	L17–20	5	64.9pp	91.4%	−18.5pp	Harmful

Table 7: Complete PCA intervention results sorted by Count Δ. Multi-layer interventions that avoid L17 alone achieve the best trade-offs.

Rank	Layer	Head	Selectivity Ratio	Power
1	16	4	1.41	0.55e-3
2	18	15	1.40	0.85e-3
3	8	8	1.40	0.26e-3
4	12	1	1.34	0.33e-3
5	17	13	1.27	0.68e-3
6	16	12	1.25	0.48e-3
7	18	7	1.24	0.72e-3
8	17	8	1.22	0.61e-3
9	16	19	1.21	0.44e-3
10	12	14	1.20	0.30e-3
11	18	2	1.19	0.69e-3
12	17	0	1.18	0.58e-3
13	16	7	1.17	0.42e-3
14	19	11	1.16	0.55e-3
15	12	9	1.15	0.28e-3
16	18	19	1.14	0.66e-3
17	17	15	1.13	0.54e-3
18	20	3	1.12	0.49e-3
19	16	0	1.11	0.39e-3
20	19	6	1.10	0.52e-3

Table 8: Top 20 OCR-selective attention heads. Selectivity ratio = OCR attention / background attention. Heads concentrate in L12, L16–L20.

Interv.	OCR	Cnt	Emb	RWQ
baseline	—	78.9	48	64
L2_pc1	**68**	69.2	25	42
L2_pc5	54.6	29.2	4	6
L3_pc5	23.2	77.8	47	49
L2-3_pc1	**77**	69.2	15	40
L20_pc5	20.6	79.5	66	66
L17-24_pc3	29.9	82.2	54	65
L19-22_pc5	73.2	78.4	66	65

Table 9: InternVL retention (%). L2–L3: collapse. L17–L20: minimal.

Key Insight. The stark contrast between L2–L3 interventions (trade-offs/collapse) and L17–L20 interventions (no effect) validates our bottleneck localization methodology: interventions only impact retention when applied to the model's *actual* OCR routing layers.

C.2 Qwen3-VL-2B-Instruct Results

Qwen3-VL-2B-Instruct has a bimodal OCR bottleneck at L8–10 and L25–26. Table 10 shows key interventions around the mid-network region (L12–L16). Baseline OCR accuracy: 87.1%.

Key Insight. L12_pc5 achieves the highest CountBench improvement (+5.6pp) but at the cost of EmbSpatial (−6pp) and RealWorldQA (−6pp). This trade-off pattern contrasts with Qwen3-VL-4B's favorable trade-off (gains on two metrics with mild degradation on one), supporting our hypothe-

Interv.	OCR	Cnt	Emb	RWQ
baseline	—	90.3	78	66
L12_pc5	31.4	**95.9**	72	60
L13_pc5	41.2	86.7	75	68
L14_pc5	34.5	92.3	77	56
L15_pc5	42.3	61.0	74	62
L12-16_pc3	23.2	91.3	71	68
L13-15_pc5	**68**	74.4	75	56

Table 10: Qwen3-VL-2B (%). L12_pc5: +5.6pp Count.

sis that larger models develop more modular OCR circuits.

C.3 Phi-4-multimodal-instruct Results

Phi-4-multimodal-instruct's OCR bottleneck is at L3–L9 ($\sim$10–25% depth). Table 11 shows interventions at bottleneck and control layers. Baseline OCR accuracy: 73.7%.

Key Insight. Phi-4 shows a clear trade-off pattern: L8_pc3 improves CountBench (+4.3pp) but degrades EmbSpatial (-13pp) and RealWorldQA (-12.5pp). L8_pc1 (single component) has almost no effect, while L8_pc3 (three components) causes significant changes—suggesting OCR information is encoded across multiple principal components.

Interv.	OCR	Cnt	Emb	RWQ
baseline	—	89.2	62	66
L3_pc3	32.5	80.4	39	48
L6_pc3	21.6	92.5	45	49
L8_pc3	38.6	**93.5**	49	54
L3-9_pc3	**64**	40.7	22	27
L16_pc3	7.7	90.6	59	61
L28_pc3	6.2	88.4	62	63

Table 11: Phi-4 (%). L8_pc3: +4.3pp Count.

D Qualitative Examples

To illustrate the behavioral effects of OCR subspace removal, Table 12 presents example model outputs before and after intervention. These examples demonstrate that our PCA-based intervention selectively disrupts text reading while preserving other visual understanding capabilities.

Interpretation. The qualitative examples reveal a striking pattern: under OCR intervention, the model fails to decode text and instead repeats the question prompt. This suggests our intervention removes the pathway that converts visual text representations into decoded strings. The model can still perceive that an image contains text regions (it attempts to respond), but cannot extract the textual content.

Crucially, non-OCR tasks remain intact. Object counting and spatial reasoning produce identical correct outputs before and after intervention. This selective disruption validates our claim that OCR processing occupies a separable subspace in the residual stream, and that surgical removal is possible without collateral damage to general visual understanding.

Task	Question	Baseline	Intervened
OCR Tasks (EgoTextVQA — Expected: Degradation)			
Scene text	"What text is visible?"	"KENNEDY HIGH the global school Ph: 6464647788..."	"What text is visible in this image? List all text..." (repeats question)
Scene text	"What text is visible?"	"DR. LAVANYA DENTAL CLINIC Bengali Food..."	"What text is visible..." (repeats question)
Non-OCR Tasks (Retention Benchmarks — Expected: Preservation)			
Counting	"How many objects?" (ten gaming headsets)	"10" ✓	"10" ✓
Counting	"How many objects?" (three light bulbs)	"3" ✓	"3" ✓
Spatial	"Which object is closest?" (A. table B. towel C. door D. basket)	"D" ✓	"D" ✓

Table 12: Real examples from EgoTextVQA (OCR) and retention benchmarks (CountBench, EmbSpatial). Under L17-19_pc5 intervention, the model fails to read scene text (repeating the question instead) while correctly answering counting and spatial reasoning questions.