PROCESS-THEN-RETRIEVE: A MECHANISTIC STUDY OF CROSS-MODAL ALIGNMENT IN VISION-LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Understanding the internal integration of visual and textual data in vision–language models (VLMs) remains a significant challenge. We present a mechanistic study of adapter-based VLMs, using PALIGEMMA-3B and QWEN2-VL as representative models, to test the hypothesis that models follow a two-phase workflow: early layers prioritize textual processing, while later layers execute cross-modal retrieval. Using representational similarity analysis, attention patching, and residual stream attribution, we reveal that early layers preserve visual embeddings with minimal modification while focusing on text. Significant cross-modal alignment and visual attention appear only in the final layers. We find that this structural bias is a primary contributor to textual dominance, where linguistic priors can override conflicting visual evidence. Our results provide a foundation for addressing the ”modality gap” and offer insights into multimodal reasoning in VLM architectures.

1 INTRODUCTION

State-of-the-art vision–language models (VLMs) such as PALIGEMMA integrate large language models with pre-trained vision encoders via projection adapters to achieve strong multimodal performance, but the internal mechanisms that reveal how visual information enters and shapes the linguistic latent space remain a ”black box” (Beyer et al., 2024b;a; Shukor & Cord, 2024). Prior work identifies a “modality gap,” characterized by geometric separation, late-stage processing, and text dominance, causing models to perform worse on vision-centric tasks (Shukor & Cord, 2024; Neo et al., 2024; Venhoff et al., 2025b; Nikankin et al., 2025; Wang et al., 2024). Focusing on PALIGEMMA-3B, we show that its ”processing-then-retrieval” pipeline causes early layers to prioritize textual context while deferring visual information. We attribute this behavior to the structural constraints of adapter-based architectures, in which the linear projection of visual tokens requires a context-building phase before effective cross-modal integration can occur. While recent work by Venhoff et al. (2025b) identifies late-stage integration as a general phenomenon in VLMs, our work moves beyond observation to identify mechanisms driving this behavior in adapter-based architectures. We validate the existence of the ”modality gap” in two ways. First, we mechanistically attribute this delay to a ”process-then-retrieve” workflow, in which early layers prioritize text-based context before visual retrieval occurs. Second, through activation patching, we demonstrate that visual processing is largely redundant across subsequent layers for VQA tasks, contradicting the assumption that ”visual blindness” is solely an issue of how attention is allocated (Kang et al., 2025).

2 EXPERIMENTS

To evaluate cross-modal alignment, we utilize a Visual Question Answering (VQA) retrieval task. We prompt the model with "`<image> How many objects are in the image?`" using 100-200 valid samples from the PixelProse dataset (Singla et al., 2024).

Datasets: We utilize the PixelProse dataset containing over 16 million image-caption pairs. For each entry, we use the image and its paired caption (VLM-generated or human-made).

Model: We use PALIGEMMA's SigLIP encoder (448x448 input, 1024 tokens) and a linear adapter that projects image patches into the language decoder's text space. The decoder consists of 18 layers (PALIGEMMA) or 28 layers (QWEN2-VL), each containing transformer blocks that integrate multi-modal information (Beyer et al., 2024b).

2.1 VISION REPRESENTATION ACROSS LAYERS

We track how visual embeddings evolve through the layers by computing the cosine similarity between vision-token representations for consecutive layers l and $l+1$ and averaging it across all vision tokens for 100 samples. As shown in Figure 1, in layers 1-10, the similarity remains high, indicating that these layers consistently attend to tokens without modifying the visual content. When the raw data in the embedding layer is carried forward into layers 11-18, the similarity drops, indicating a transition towards cross-modal representation. The visual vectors bridge into the textual space, allowing the final layer to determine an answer based on the information from both modalities.

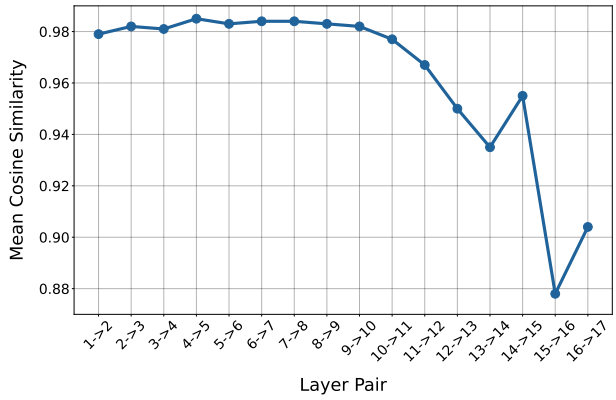


Figure 1: Mean cosine similarity between vision-token representations

With this information, we determined how PALIGEMMA modifies visual embeddings. The residual stream processes these changes through a mixture of attention and MLP contributions. Specifically, attention acts dynamically to absorb context, while MLP integrates individual tokens systematically. Our analysis shows that attention contributions plays a dominant role in processing visual embeddings. Refer to Appendix B for residual stream sub-contribution results. Subsequently, we focused on attention mechanisms to understand the role of relational processing in cross-modal integration.

2.2 ATTENTION-OUTPUT ACTIVATION PATCHING

We conduct attention-output activation patching experiments on PALIGEMMA to determine the influence of intermediate representations in the visual and textual domains, specifically intervening at the Self-Attention submodule. We extract the hidden states from the self-attention mechanisms of the source layers (early: 2-4, mid: 5-7, late: 10-12) and feed them into the self-attention outputs of the final 3 layers of the model. We evaluated with logit and cross-entropy losses on the target token. Refer to Appendix D for results on QWEN2-VL.

Vision and textual pathways yield contrasting results. In the vision pathway, early-to-mid and late-to-mid patching had minimal effect, while early-to-late and mid-to-late patching reduced logits by approximately 0.27-0.28, indicating interference when earlier visual features overwrite later representations. In the textual pathway, however, early-to-mid and early-to-late patching increased logits by 6 and significantly reduced loss, whereas late-to-mid patching decreased performance. This suggests that textual representations are processed heavily in earlier layers, while vision representations stay stable throughout. Refer to Appendix E for residual stream patching results in PALIGEMMA, as well as QWEN2-VL.

Table 1: Attention Output Patching Results

Category	Patch Direction	Avg Clean Score	Avg Patched Score	Avg Δ Logit	Avg Clean Loss	Avg Patched Loss
Vision	Early → Mid	12.8346	12.9177	+0.0831	14.8269	14.8542
Vision	Late → Mid	12.8374	12.8316	-0.0058	14.8260	14.8178
Vision	Early → Late	12.8346	12.5501	-0.2845	14.8269	14.5287
Vision	Mid → Late	12.8346	12.5695	-0.2651	14.8269	14.5158
Text	Early → Mid	12.8346	19.0637	+6.2291	14.8269	11.1738
Text	Late → Mid	12.8346	12.3032	-0.5313	14.8269	15.6569
Text	Early → Late	12.8346	18.7865	+5.9519	14.8269	9.9437
Text	Mid → Late	12.8363	17.3727	+4.5364	14.8272	10.1878

2.3 Attention by Modality per Layer

Building on the activation patching results, we analyze how attention is distributed across modalities and layers to determine whether PaliGemma integrates visual and textual information throughout the network or through late layers. For each layer, we extract the attention weights from the final token query and compute the mean attention towards vision and text tokens separately. From this, we track how the model's focus on each modality shifts through layers.

As shown in Figure 2, the model shows a clear reliance on textual tokens in most layers, with visual attention increasing only in the final layers. This indicates that PaliGemma defers integrating visual features for the final layers, where cross-modal alignment for answer generation occurs. Attention decomposition reveals strong modality segregation in query-key interactions. We find that vision queries preferentially attend to vision keys, and respectively, text queries and text keys do the same. Cross-modal attention, from text queries to vision keys required for answering visual questions, is negligible in early layers and becomes significant only in the final transformer blocks. This confirms that visual representations are processed in isolation until late layers, but that they are "retrieved" only when the textual reasoning phase is near completion.

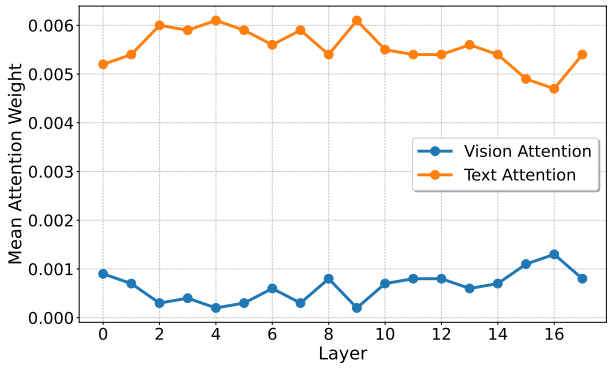
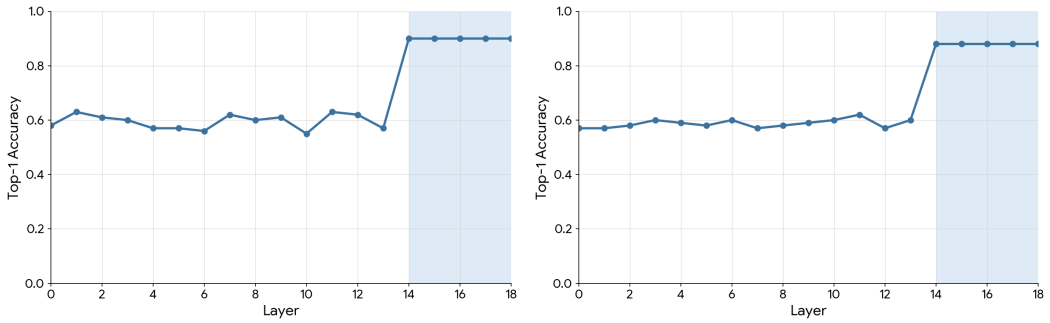


Figure 2: Mean vision vs. text attention weights across layers

2.4 Linear Probing Generalization

We evaluate the stability of visual representations across the network using linear probing. From the PixelProse dataset, we constructed a subset of image–caption pairs containing six common object categories: dog, cat, car, person, tree, and building. Linear classifiers were trained separately on visual and textual representations by extracting image-token embeddings and text-token embeddings from the last four transformer layers, and were evaluated across all layers. We hypothesize that the accuracy of vision probes remains relatively stable across layers, in contrast to text probes, which are trained on representations expected to be processed more heavily in earlier layers. However, we observe that probe performance is highest when classifiers are evaluated on the same layers on which they were trained, and accuracy degraded when applied to earlier layers. This might be because probes assume linearity and may miss non-linear changes.



(a) Vision Representation Accuracy (b) Text Representation Accuracy

Figure 3: Linear probing: Accuracy remains relatively stable across layers (about 55%–63%), followed by a spike in layers 14-18 to around 90% for both vision (left) and text (right) representations.

2.5 Residual Stream Update

To quantify vision processing, we analyze the total vision-token representations updates in the residual stream. We compute the difference between pre- and post-layer representations. Figure 4 visualizes the averaged magnitude changes and discovered that processing is weakest in early-to-mid layers and spikes in the final layers, suggesting transformation occurs primarily at output stages.

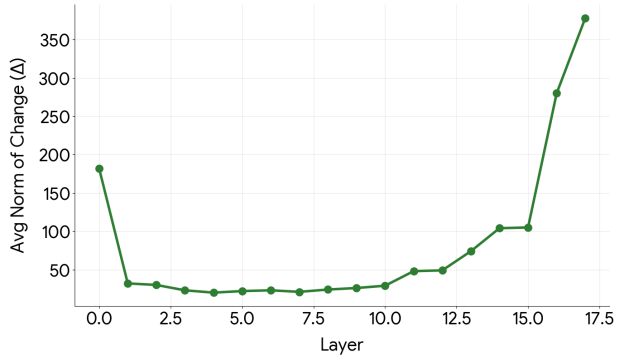


Figure 4: Vision token representation changes: High initial processing at layer 0 before dropping for layers 1-10. Spikes in final layers 15-17.

3 Conclusion

In this work, we analyzed the mechanistic structure of PaliGemma-3B, confirming our hypothesis that adapter-based VLMs typically exhibit early-layer textual data processing behaviors, followed by information retrieval in later layers. Our findings raise fundamental questions about multi-modal representation learning in VLMs. The separation of modalities until late layers suggests that current architectures consist of modality-specific processing streams that merge only near output. The dominance of textual attention across the network, with visual attention peaking only in the final layers, suggests an unbalanced focus on language that ultimately limits visual reasoning capabilities. Our linear probing experiments further confirmed that models prioritize text processing in the initial layers, with visual data being used for information retrieval only in later layers, which may not be sufficient time to extract all relevant information.

REFERENCES

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Lucas Beyer, Andreas Steiner, Jean-Baptiste Alayrac, et al. PaliGemma: A versatile 3b VLM for transfer. *arXiv preprint arXiv:2407.07726*, 2024a.

Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, et al. Paligemma: A versatile 3b vlm for transfer. *arXiv preprint arXiv:2407.07726*, 2024b.

Ido Cohen, Daniela Gottesman, Mor Geva, and Raja Giryes. Performance gap in entity knowledge extraction across modalities in vision language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 29095–29108, 2025.

Sedigheh Eslami and Gerard de Melo. Mitigate the gap: Improving cross-modal alignment in clip. In *The Thirteenth International Conference on Learning Representations*, 2025.

Abrar Fahim, Alex Murphy, and Alona Fyshe. It's not a modality gap: Characterizing and addressing the contrastive gap. *arXiv preprint arXiv:2405.18570*, 2024.

Stephanie Fu, Tyler Bonnen, Devin Guillory, and Trevor Darrell. Hidden in plain sight: Vlms overlook their visual representations. *arXiv preprint arXiv:2506.08008*, 2025.

Tianze Hua, Tian Yun, and Ellie Pavlick. How do vision-language models process conflicting information across modalities? *arXiv preprint arXiv:2507.01790*, 2025.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2312.06968*, 2023a.

Chaoya Jiang, Haiyang Xu, Mengfan Dong, Jiaxing Chen, Wei Ye, Ming Yan, Qinghao Ye, Ji Zhang, Fei Huang, and Shikun Zhang. Hallucination augmented contrastive learning for multimodal large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27036–27046, 2024.

Zixuan Jiang, Jiaqi Gu, Hanqing Zhu, and David Pan. Pre-rmsnorm and pre-crsnorm transformers: Equivalent and efficient pre-ln transformers. *Advances in Neural Information Processing Systems*, 36:45777–45793, 2023b.

Seil Kang, Jinyeong Kim, Junhyeok Kim, and Seong Jae Hwang. See what you are told: Visual attention sink in large multimodal models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=7uDI7w5RQA.

Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in Neural Information Processing Systems*, 35:17612–17625, 2022.

Zhining Liu, Ziyi Chen, Hui Liu, Chen Luo, Xianfeng Tang, Suhang Wang, Joy Zeng, Zhenwei Dai, Zhan Shi, Tianxin Wei, et al. Seeing but not believing: Probing the disconnect between visual attention and answer correctness in VLMs. *arXiv preprint arXiv:2510.17771*, 2025.

Grace Luo, Trevor Darrell, and Amir Bar. Vision-language models create cross-modal task representations. *arXiv preprint arXiv:2410.22330*, 2024.

Jack Merullo, Louis Castricato, Carsten Eickhoff, and Ellie Pavlick. Linearly mapping from image to text space, 2023. URL https://arxiv.org/abs/2209.15162.

Clement Neo, Luke Ong, Philip Torr, Mor Geva, David Krueger, and Fazl Barez. Towards interpreting visual information processing in vision-language models. *arXiv preprint arXiv:2410.07149*, 2024.

Yaniv Nikankin, Dana Arad, Yossi Gandelsman, and Yonatan Belinkov. Same task, different circuits: Disentangling modality-specific mechanisms in vlms. *arXiv preprint arXiv:2506.09047*, 2025.

nostalgebraist. interpreting gpt: the logit lens. https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens, 2020.

Isabel Papadimitriou, Huangyuan Su, Thomas Fel, Sham Kakade, and Stephanie Gil. Interpreting the linear structure of vision-language model embedding spaces. *arXiv preprint arXiv:2504.11695*, 2025.

Achyuta Rajaram, Sarah Schwettmann, Jacob Andreas, and Arthur Conmy. Line of sight: On linear representations in VLLMs. *arXiv preprint arXiv:2506.04706*, 2025.

François Role, Sébastien Meyer, and Victor Amblard. Fill the gap: Quantifying and reducing the modality gap in image-text representation learning. *arXiv preprint arXiv:2505.03703*, 2025.

Simon Schrodi, David T Hoffmann, Max Argus, Volker Fischer, and Thomas Brox. Two effects, one trigger: On the modality gap, object bias, and information imbalance in contrastive vision-language models. *arXiv preprint arXiv:2404.07983*, 2024.

Jain Shantanu. tiktoken, 2025. URL https://github.com/openai/tiktoken/.

Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

PY Shi, M Welle, M Bjørkman, and D Kragic. Understanding the modality gap in clip. *ICLR, Stockholm, Sweden*, 2023.

Mustafa Shukor and Matthieu Cord. Implicit multimodal alignment: On the generalization of frozen LLMs to multimodal inputs. *arXiv preprint arXiv:2405.16700*, 2024.

Mong Yuan Sim, Wei Emma Zhang, Xiang Dai, and Biaoyan Fang. Can VLMs actually see and read? a survey on modality collapse in vision-language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 24452–24470, 2025.

Vasu Singla, Kaiyu Yue, Sukriti Paul, Reza Shirkavand, Mayuka Jayawardhana, Alireza Ganjdanesh, Heng Huang, Abhinav Bhatele, Gowthami Somepalli, and Tom Goldstein. From pixels to prose: A large dataset of dense image captions. *arXiv preprint arXiv:2406.10328*, 2024.

Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal LLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9568–9578, 2024.

Constantin Venhoff, Ashkan Khakzar, Sonia Joseph, Philip Torr, and Neel Nanda. How visual representations map to language feature space in multimodal LLMs. *arXiv preprint arXiv:2506.11976*, 2025a.

Constantin Venhoff, Ashkan Khakzar, Sonia Joseph, Philip Torr, and Neel Nanda. Too late to recall: Explaining the two-hop problem in multimodal knowledge retrieval. *arXiv preprint arXiv:2512.03276*, 2025b. URL https://openreview.net/forum?id=qeL8fi8GS7.

Jiayu Wang, Yifei Ming, Zhenmei Shi, Vibhav Vineet, Xin Wang, Sharon Li, and Neel Joshi. Is a picture worth a thousand words? delving into spatial reasoning for vision language models. *Advances in Neural Information Processing Systems*, 37:75392–75421, 2024.

Zhaofeng Wu, Xinyan Velocity Yu, Dani Yogatama, Jiasen Lu, and Yoon Kim. The semantic hub hypothesis: Language models share semantic representations across languages and modalities. *arXiv preprint arXiv:2411.04986*, 2024.

Our code is available for reproducibility at `https://anonymous.4open.science/r/Process-then-Retrieve/`.

A Other Models

For the purposes of output replication, we repeat these experiments on other models as noted in later sections of the Appendix. We use Qwen2-VL-2B-Instruct, a pretrained model by Alibaba. Experiments with this model dynamically process images with a patch size of 14x14 pixels, while text input is tokenized systematically through byte pair encoding (BPE) based on tiktoken (Shantanu, 2025). The model uses a Qwen2-specific Vision Transformer Architecture (ViT) that is optimized by SwiGLU, which affects activation function (Shazeer, 2020), and RMSNorm, which replaces the traditional layer normalization technique (Jiang et al., 2023b; Bai et al., 2023). This is implemented throughout 28 layers.

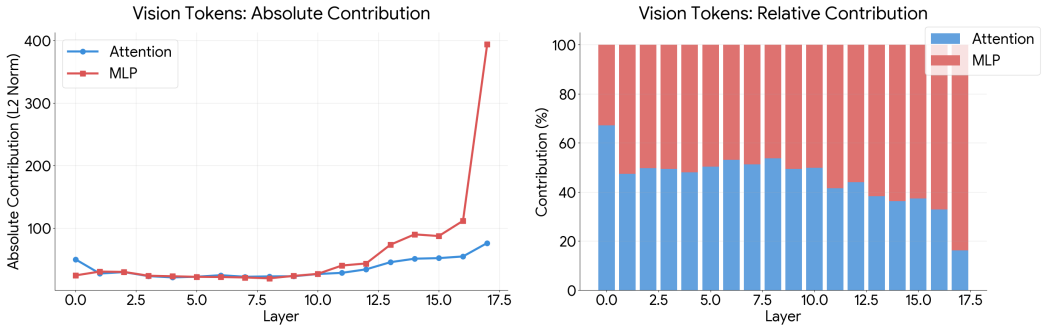
B Attention vs MLP Contributions

B.1 PaliGemma

In order to understand how much each layer of the multi-modal model, PaliGemma, processes vision tokens, we analyze the sum of layer outputs which flows through the transformer, or the residual stream. The attention and MLP sublayers add contributions, which we calculate, to the vision token representations. This is found using the change of vision token representations before (pre-contribution) and after (post-contribution) each layer processes the tokens across all hidden states. The scores are extracted from the post-contribution layer.

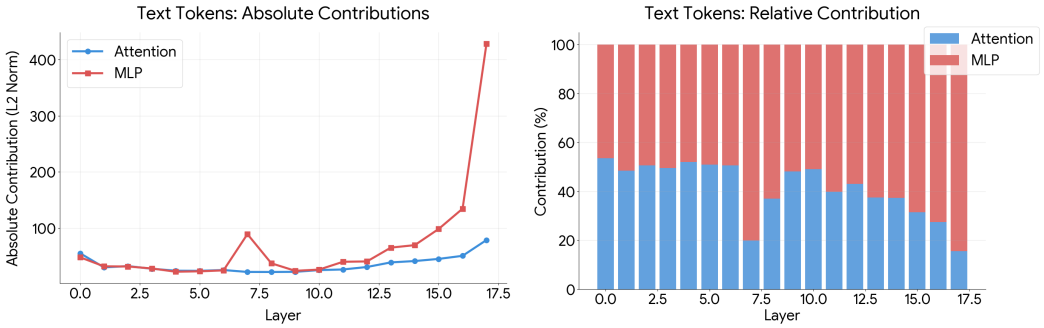
To determine which components drive these changes, we further decompose each layer's residual update by separately measuring the contributions of the attention and MLP sublayers, enabling a direct comparison of their roles in vision-token processing as shown in Figure 5. To determine which of these drives vision processing, we decompose each layer's residual stream contributions by passing vision tokens through the attention and MLP sublayers separately.

Additionally, we analyze the attention and MLP sublayers for text tokens. Figure 6 demonstrates that between these two residual stream contributions, attention consistently increases between layers while MLP spikes dramatically in late layers. However, compared to the absolute total contributions in Figure 5, the contribution is much lower.



(a) PaliGemma Vision Tokens Absolute Contributions (b) PaliGemma Vision Tokens Relative Contributions

Figure 5: Attention and MLP contributions to vision token updates: Relative contributions reveal attention and MLP contribute roughly equally throughout most layers, with attention comprising 40-60% of total changes. MLP spikes at the final layer, comprising over 80% of contributions to the residual stream. Absolute contributions demonstrate that the L2 Norm for MLP jumps from around 100 to 400 in the final transformer layer.



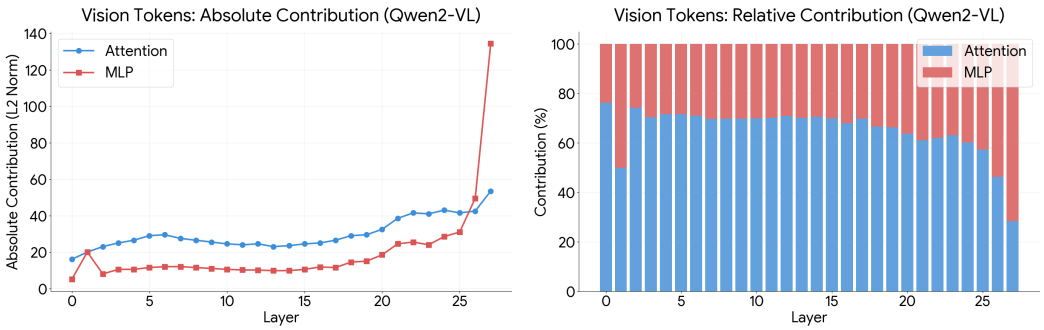
(a) PaliGemma Text Tokens Absolute Contributions (b) PaliGemma Text Tokens Relative Contributions

Figure 6: Attention and MLP contributions to text token updates: Relative contributions reveal attention and MLP contribute roughly equally throughout most layers, with attention comprising 40-60% of total changes. Absolute contributions shows that MLP spikes at Layer 7 to an L2 Norm of 100 before stabilizing, then surges once more at the final layer to 400.

B.2 QWEN2

Next, we replicate these experiments in QWEN2-VL-2B-INSTRUCT. In both Figure 7 and Figure 8, attention is the major contributor to the residual stream. This aligns with PALIGEMMA's distribution whose main contributions are sourced from attention. However, attention in PALIGEMMA only overtakes MLP contribution by a slight amount, while the distribution for QWEN2 shows a much larger degree of favorability towards attention.

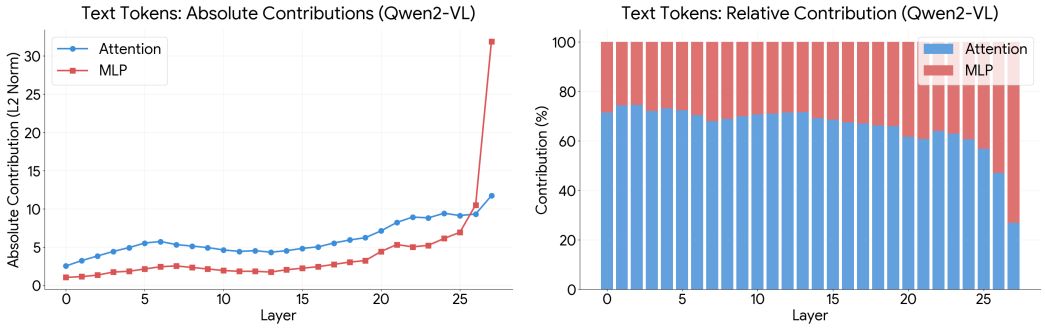
In accordance to vision tokens, MLP contribution in QWEN2 has a small spike in the very early layers and a large spike in late layers. This shows when MLP processes visual features: in very early layers, the raw data is marked. During the late layers, MLP finally transforms the data during cross-modal integration. This differs from Figure 6 in that PALIGEMMA spikes in MLP in early-to-intermediate layers, rather than in the second layer such as QWEN2.



(a) QWEN2-VL Vision Token Absolute Contributions (b) QWEN2-VL Vision Token Relative Contributions

Figure 7: QWEN2's Attention and MLP contributions to vision token updates: Relative contributions reveal attention consistently contributes more than MLP to the residual stream in early to intermediate layers, with a small spike in MLP during the very early layers. In late layers, MLP drastically contributes to the residual stream.

In Figure 8, MLP's contribution to the residual stream spikes in late layers. These results are similar to the ones found in PALIGEMMA's contributions: the textual features are being integrated alongside visual features for the final output. However, in contrast to PALIGEMMA's results, attention contributes to a higher degree overall at nearly 60%-70% until the late layers.



(a) QWEN2-VL Text Token Absolute Contributions (b) QWEN2-VL Text Token Relative Contributions

Figure 8: QWEN2's Attention and MLP contributions to text token updates: Relative contributions reveal attention consistently contributes more than MLP to the residual stream in early to intermediate layers. In late layers, MLP drastically contributes to the residual stream.

C ATTENTION CONTRIBUTIONS VISUALIZED

C.1 PALIGEMMA

For the purposes of this experiment, we compare the raw scores from layer 10, an intermediate layer, to get a better understanding of the processing distribution. Since they are a decomposition of the attention contributions, they are small compared to the total composition. Figure 9 shows that image token queries primarily attend to image token keys rather than incorporating information from text queries. Similarly, text token queries primarily attend to text token keys, with low attention from image queries. This aligns with the "process-then-retrieve" workflow, which defers information integration across modalities in intermediate layers.

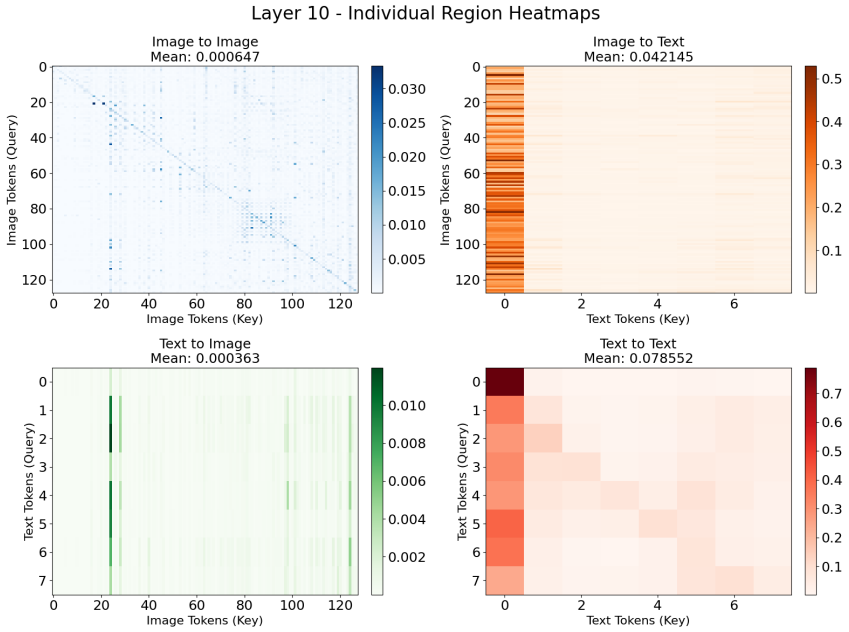


Figure 9: Attention score decomposition for PALIGEMMA-3B: Each quadrant shows attention from query tokens (rows) to key tokens (columns). Top-left: Vision queries attending to vision keys (V to V, mean=0.000647). Top-right: Vision queries attending to text keys (V to T, mean=0.000363). Bottom-left: Text queries attending to vision keys (T to V, mean=0.042145). Bottom-right: Text queries attending to text keys (T to T, mean=0.078552).

C.2 QWEN2

This experiment was repeated with the QWEN2-VL-2B-INSTRUCT model.

Similar to the experiment performed in PALIGEMMA, Figure 10 demonstrates that each query token primarily attends to a key of the same modality. As further evidence, the image token queries do not attend to the text token keys at all, indicating the lack of cross-modal integration towards text token keys. This signals a structural bias, in which both vision and text queries attend to image keys while only text queries attend to text keys. Broadly, this suggests that QWEN2 has a textual bias, allowing cross-modal integration to occur in only one direction. On a lesser note, the image queries attend to the image keys more than the text queries do. However, the mean scores still lie within a similar range of each other. The relatively equal distribution, in which both modalities are emphasized when attending to image token keys, further signals model reliance on text information for visual output.

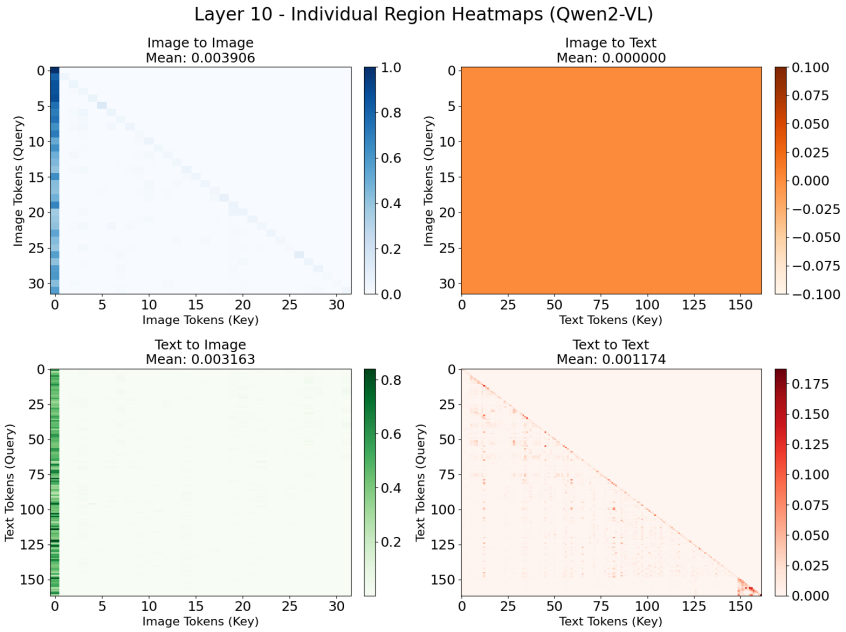


Figure 10: Attention score decomposition for QWEN2-VL-2B: Each quadrant shows attention from query tokens (rows) to key tokens (columns). Top-left: Vision queries attending to vision keys (V to V, mean=0.003906). Top-right: Vision queries attending to text keys (V to T, mean=0). Bottom-left: Text queries attending to vision keys (T to V, mean=0.003163). Bottom-right: Text queries attending to text keys (T to T, mean=0.001174).

D ATTENTION-OUTPUT PATCHING EXPERIMENTS ON QWEN2-VL

We perform attention-only activation patching in QWEN2-VL to determine the role of the attention mechanism in the model. We isolate the self-attention submodule in the Vision and Text layers. With the same patching configuration as illustrated in PALIGEMMA, we extract the hidden states output from the attention mechanism of the ”source layers” and inject them into the attention output of ”target layers.” The MLPs of the target layers were left untouched, and we continued to process the residual stream naturally. Patching all activations to the late layers from early layers compared to only the attention activation outputs, shows the model’s reliance on the MLPs – if the later layers only have the activation outputs from the earlier layers, the model can still derive a mostly reliable answer.

D.1 TEXTUAL RESULTS

Front-layer activation patching in QWEN2, as seen in Appendix E.2.2, demonstrates that skipping text layers results in a massive logit drop, compared to the much smaller drop observed in most

configurations when patching only attention. This confirms that the MLPs help adapter-based VLMs process and reason the information provided by textual captions. The -1.41 drop from early to late layers shows that attention is not redundant. Instead, the model determines what tokens to attend to, from vision to answer tokens, as it progresses throughout the layers. The MLPs actually perform the computation as prompted, which in this case was counting. Attention patching remained mainly unaffected.

D.2 Vision Results

The vision encoder typically displayed high redundancy in its attention patterns. There is a near-zero impact of patching early attention into mid layers, and the minor drop when patching early attention into late layers suggests that the models' attention maps are established in the very early layers. The subsequent attention layers mainly reinforce the notion of where the model "should look" without adding much additional information. Once the model identifies the relevant object regions in the early layers, the attention module's job is complete. The deeper layers most likely use their MLPs to extract features from those attended regions.

Table 2: Layer-wise Attention Patching Results for Qwen2-VL

Modality	Patch Direction	Clean	Patched	Δ Logit	Clean Loss	Patched Loss
Vision	Early $\rightarrow$ Mid	15.79	15.82	+0.04	19.57	19.92
Vision	Late $\rightarrow$ Mid	15.79	15.52	-0.27	19.57	20.68
Vision	Early $\rightarrow$ Late	15.77	15.10	-0.67	19.58	19.87
Vision	Mid $\rightarrow$ Late	15.77	15.22	-0.55	19.58	19.78
Text	Early $\rightarrow$ Mid	15.77	14.73	-1.04	19.58	20.14
Text	Late $\rightarrow$ Mid	15.79	16.02	+0.23	19.57	20.12
Text	Early $\rightarrow$ Late	15.79	14.37	-1.41	19.57	17.18
Text	Mid $\rightarrow$ Late	15.79	14.89	-0.90	19.58	14.95

E Activation Front Patching Experiments

E.1 PaliGemma

E.1.1 Experimental Setup

We perform front-activation patching experiments to compare textual and visual representations across various layer groups in PaliGemma.

For each image, we run two forward passes: a clean and a patched run. During the clean run, the model processes the image-text input normally, while hidden activations from selected layers are recorded. During the patched run, or the second forward pass, the hidden activations in the target layers are replaced with the activations saved from the earlier source layers in the clean run. For textual front patching, we intervene on the transformer layers of PaliGemma. Activations from early or mid textual layers, the source layers, are saved during the clean pass and injected into the last three textual layers, target layers, before the output layer, during the patched pass. For vision front patching, we apply the same methodology to the encoder layers of the vision tower, injecting activations from earlier vision layers into later ones. Across both modalities, we isolate the textual and vision representations to properly analyze their results. Performance is measured using the logit assigned to the ground-truth answer token at the final position and the corresponding cross-entropy loss. The various source layer groupings we tested front patching on were early layers of 2, 3, 4, mid layers of 7, 8, 9, and late layers of 10, 11, 12,

E.1.2 Textual Representation Patching Results

In textual front patching, the performance typically decreases across various groups of source patching. Early-to-late patching causes target logits to decrease by more than 10 points on average, while cross-entropy loss sharply increases to over 30.

In general, any sort of front patching, early, mid, and late to the final layers all result in a sharp increase in loss. This indicates that textual representations are not interchangeable across layers, and that later textual layers encode essential information required for reasoning and answer generation.

E.1.3 VISION REPRESENTATION PATCHING RESULTS

On the contrary, front patching within the visual representations produces weaker results. Early vision patching causes a reduction in the target logit, but a small increase in the loss, which shows that PALIGEMMA can function regardless.

When patching mid and late vision layers, the target logit and cross-entropy loss continue to decrease compared to the clean run results. Later vision-layer activations aren't critical to the reasoning and answer generation phase of the model. Intervening with these layers only leads to lower-entropy predictions. We suggest that visual representations are partially redundant and become more abstract as the depth increases.

E.1.4 MODALITY ASYMMETRY

Overall, textual representations are depth-sensitive and essential in PALIGEMMA, while visual representations are depth-dependent and have a weaker influence. Visual information isn't an independent factor in the model's final prediction, but rather as a signal for the biases and logic in the textual stream. Later vision-layer activations cannot serve alone, and require multi modal alignment.

Table 3: Layer-wise Representation Patching Results for PaliGemma-3B

Modality	Layers	Clean	Patched	Δ Logit	Clean Loss	Patched Loss
Vision	2–4	12.79	11.79	-0.99	14.86	14.70
Vision	5–7	12.79	12.21	-0.58	14.86	13.98
Vision	7–9	12.79	12.50	-0.28	14.86	13.36
Vision	10–12	12.79	12.40	-0.39	14.86	13.25
Text	2–4	12.76	-0.75	-13.51	14.87	23.64
Text	5–7	12.79	2.62	-10.17	14.86	32.25
Text	7–9	12.76	0.28	-12.48	14.87	33.49
Text	10–12	12.76	0.92	-11.84	14.87	25.98

E.2 QWEN2-VL

E.2.1 EXPERIMENTAL SETUP

Within the QWEN2-VL-2B-INSTRUCT architecture, we perform Front Patching, similar to PALIGEMMA, by extracting internal activation states from earlier layers and injecting them into the final three layers of the model.

E.2.2 TEXTUAL REPRESENTATION PATCHING RESULTS

By patching different configurations of layers (2-4, 5-7, 7-9, and 10-12) into the last three layers, the language model of QWEN2-VL severely collapsed. Across all configurations, the logit score for the correct answer dropped from a highly confident score of 15.79 to around 1.0-2.3. The LLM's residual stream experiences don't benefit from patching and require linear transformation. The extreme logit drop reveals that the "concept" of the target logit "2" is not retrieved immediately. The model likely spends the intermediate layers integrating the visual tokens with the text prompt, which would've increased the probability of the counting token. Patching across these configurations underscores the importance of the reasoning phase, where both modalities align.

E.2.3 VISION REPRESENTATION PATCHING RESULTS

Patching earlier layers of the vision tower into the final layer improved the logit score by approximately 0.65, which is a slight performance increase. However, for other source layer configurations, there were negligible performance drops, maintaining the model's confidence in the correct answer.

This suggests that the fundamental features required for "counting" (what the model was specifically prompted with) are established in the very early layers of the encoder.

Table 4: Layer-wise Representation Patching Results for Qwen2-VL

Modality	Layers	Clean	Patched	Δ Logit	Clean Loss	Patched Loss
Vision	2–4	15.78	16.42	+0.65	19.58	20.18
Vision	5–7	15.78	15.49	-0.29	19.58	20.33
Vision	7–9	15.79	15.40	-0.39	19.57	20.20
Vision	10–12	15.77	15.49	-0.28	19.58	20.60
Text	2–4	15.79	2.03	-13.76	19.57	12.09
Text	5–7	15.79	1.82	-13.97	19.57	10.09
Text	7–9	15.79	1.01	-14.77	19.57	11.24
Text	10–12	15.79	2.29	-13.50	19.57	12.00

F Formal Metric Definitions

We formally define the metrics used to evaluate the models' cross-modal behavior. Let L be the number of transformer layers; $H^{(l)} \in \mathbb{R}^{T \times d}$ denote the hidden state activations at layer l, where T is the sequence length and d is the hidden dimension; and I_{vis} and I_{text} denote the sets of indices corresponding to vision and text tokens, respectively.

F.1 Cosine Similarity of Representations

To measure the stability of visual representations across consecutive layers, we compute the cosine similarity between the hidden state of a token at layer l and layer $l+1$. For a specific vision token at position $t \in I_{\text{vis}}$, the similarity is defined as:

$$\text{CosSim}(t, l) = \frac{H_t^{(l)} \cdot H_t^{(l+1)}}{\|H_t^{(l)}\|_2 \|H_t^{(l+1)}\|_2} \tag{1}$$

The reported metric is the average similarity over all vision tokens across N samples in the Pixel-Prose dataset:

$$\text{AvgSim}^{(l)} = \frac{1}{N} \sum_{k=1}^{N} \left(\frac{1}{|I_{\text{vis}}|} \sum_{t \in I_{\text{vis}}} \text{Sim}_k(t, l) \right) \tag{2}$$

F.2 Residual Stream Update & Decomposition

The transformer residual stream at layer l is updated by two distinct sub-modules: Multi-Head Attention (Attn) and the Multi-Layer Perceptron (MLP). Within the transformer stack of an adapter-based VLM (assuming a standard pre-LN configuration), the hidden state update is defined as:

$$H^{(l)} = H^{(l-1)} + \Delta_{\text{Attn}}^{(l)} + \Delta_{\text{MLP}}^{(l)} \tag{3}$$

where the individual contributions are:

$$\Delta_{\text{Attn}}^{(l)} = \text{Attn}(\text{LN}(H^{(l-1)})) \tag{4}$$

$$\Delta_{\text{MLP}}^{(l)} = \text{MLP}(\text{LN}(H^{(l-1)} + \Delta_{\text{Attn}}^{(l)})) \tag{5}$$

The **Residual Stream Update** metric quantifies the magnitude of change introduced by layer l. We calculate the L_2 norm of the total difference between the pre-layer and post-layer representations for a specific modality set $S \in \{I_{\text{vis}}, I_{\text{text}}\}$:

$$\text{Update}_S^{(l)} = \frac{1}{|S|} \sum_{t \in S} \|H_t^{(l)} - H_t^{(l-1)}\|_2 \tag{6}$$

To determine the drivers of this processing, we calculate the **Relative Contribution** of each submodule:

$$\text{Contrib}_{\text{Attn}}^{(l)} = \frac{\|\Delta_{\text{Attn}}^{(l)}\|_2}{\|\Delta_{\text{Attn}}^{(l)}\|_2 + \|\Delta_{\text{MLP}}^{(l)}\|_2} \tag{7}$$

F.3 Attention by Modality

We analyze the attention distribution of the final token (the query q_{last}), which is responsible for generating the next token. Let $A_{last,j}^{(l)}$ represent the attention weight from the last token to key token j at layer l, averaged across all attention heads h:

$$A_{last,j}^{(l)} = \frac{1}{N_{heads}} \sum_{h=1}^{N_{heads}} \text{Softmax}\left(\frac{Q_{last}^{(l,h)}(K_j^{(l,h)})^T}{\sqrt{d_k}}\right) \tag{8}$$

We define the **Modality Attention Score** as the total probability mass allocated to a specific modality (Vision or Text):

$$\text{Attn}_{\text{vision}}^{(l)} = \sum_{j \in I_{\text{vis}}} A_{last,j}^{(l)}, \qquad \text{Attn}_{\text{text}}^{(l)} = \sum_{j \in I_{\text{text}}} A_{last,j}^{(l)} \tag{9}$$

F.4 Averaging Procedures

All reported results are averaged over the PixelProse dataset. For any metric M computed for a single sample k, the final reported value is:

$$\bar{M} = \mathbb{E}_{x \sim \mathcal{D}}[M(x)] \approx \frac{1}{N} \sum_{k=1}^{N} M_k \tag{10}$$

Where $N = 100$ for our representational analysis experiments.