



Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

PCLens: Spectral Explainability and Concept-Level Interventions in CLIP ViT Attention Heads

Master Thesis

V. Strozzi

April 28, 2025

Advisors: Emanuele Palumbo, Alain Ryser, Prof. Dr. Julia Vogt

Department of Computer Science, ETH Zürich

Abstract

Leveraging the shared image-text representation space, recent work [24] proposes to interpret the role of multi-heads self attention (MSA) of the CLIP vision encoder (ViT) by reconstructing the activation output of individual MSA heads with text-embeddings. However, this approach fails to capture the semantic structure of polysemantic heads, where no single dominant concept emerges. We introduce PCLens, a mechanistic and spectral framework that decomposes the activation space of each MSA head into principal components (PCs), each bidirectional interpreted with both text and image semantics. This method reveals highly specialized, semantically meaningful directions, distributed across the ViT residual stream, encoding same semantic meaning in both MSA heads and final CLIP embedding latent space, enabling finer-grained interpretation of each head’s contribution to the final output. We further propose QuerySystem, which automatically pinpoints PCs encoding a given visual-text concept across different heads, highlighting redundancy and specialization of semantic across different components, and PCSelection, which enables targeted manipulation of PCs, with the goal of amplifying or removing concepts without any fine-tuning. Together, our tools uncover intriguing details of the latent structure of both intermediate and final CLIP representations, across four different ViT models, and support automatic targeted interventions such as mitigation of spurious correlations. Finally, we examine the broader applicability of our findings within Large Vision-to-Language Models (LVLMs) using the frozen CLIP ViT encoder, and find no direct transferability due to intriguing contrasts in residual stream dynamics between the [CLS] tokens and the image patch tokens, thereby proposing possible directions for future work.

Acknowledgements

I would like to thank Prof. Dr. Julia Vogt for the opportunity to work on this Master's thesis project. I am also grateful to my advisors, Emanuele Palumbo and Alain Ryser, together with Irene Cannistraci, for the valuable discussions and feedback they provided throughout these six months.

I would love to thank my mother and father: thank you for your continuous support throughout my studies and for our weekends in the mountains and in nature. They gave me so much good energy and the right peace of mind to face the week. To my sisters, Matilde and Irma, thank you for simply being around and for the joy of spending time together. I also want to express my deep gratitude to my friends, who filled this period with lots of laughter, wonderful music, random conversations, and an overall carefree and happy atmosphere. Thank you to Lorenzo, Max, Cristina, and Fil! Thank you also to Jacopo, Indro, Filo, Monday, and Andrea for the great weekends, our matches, and the lightness of our time together!

Finally, I would like to thank the professionals at the CAB Entspannung group for creating a homely atmosphere where I could always enjoy some relaxing time. This experience, together with all the sports activities offered by ASVZ, kept my hyperactive needs more than satisfied.

Contents

Acknowledgements	iii
Contents	v
1 Notation	1
1.1 Dimensions	1
1.2 Model Architectures	1
1.3 Concepts	2
1.4 CLIP Model	2
1.5 Datasets	3
1.6 ViT Residual Stream	3
1.7 LLaVA-Specific Notations	4
1.8 Principal Component Analysis	4
1.9 Explainability Methods	4
2 Introduction	7
2.1 Motivation	8
2.2 Open Questions	10
2.3 Contributions	11
3 Background	15
3.1 Contrastive Vision-Language Models	16
3.1.1 CLIP Architecture.	16
3.1.2 CLIP ViT Image Encoder $E_{\text{image}}(I)$	17
3.1.3 ViT Residual Stream ViT(I)	17
3.1.4 CLIP ViT Image Encoder Residual Stream $E_{\text{image}}(I)$	19
3.2 Textspan	20
3.3 Large Vision-Language Models	21
3.3.1 LLaVA	21
3.3.2 Image Features Residual Stream $F_{\text{image}[i]}(I)$	22

3.4	Principal Component Analysis and Eigenfaces	22
3.4.1	Principal Component Analysis	23
3.4.2	Eigenfaces	23
3.5	Explainability of ML Models	24
3.5.1	Neural Networks	24
3.5.2	Explainability of ViT Models	25
3.5.3	Explainability of Multimodal Models	25
3.5.4	Explainability CLIP	26
3.5.5	Explainability of LLaVA.	27
4	Related Work	29
4.1	Text Based Explanation of CLIP Image ViT Encoder	29
4.2	Explanation of ViT Patch Tokens Effects on LLaVA	30
5	Explaining the Latent Space of CLIP ViT Encoder $E_{\text{image}}(I)$ MSA Heads Using Text and Images	31
5.1	Overall setup	31
5.1.1	Models and Datasets	32
5.1.2	Goal	32
5.1.3	Assumptions	32
5.2	PCLens: Explaining CLIP Image Encoder $E_{\text{image}(I)}$ MSA Heads through their PCs Directions	33
5.2.1	Metric of Choice: Cosine Similarity	34
5.2.2	Recovered Zero-shot accuracy (PCLens vs TextSpan) .	36
5.2.3	(Non)-Polysemantic Heads Role (PCLens vs TextSpan)	38
5.2.4	Heads' Finer-grained Latent Space	40
5.3	QuerySystem: Automatic Location of Concepts in CLIP ViT Encoder $E_{\text{image}}(I)$	43
5.3.1	Metric of Choice: Absolute Cosine Similarity	44
5.4	PCSelection: Targeted Interventions on the ViT Encoder . .	45
5.4.1	Qualitative Reconstruction of Concept	49
5.4.2	PCs encode Semantic Meaning for both ViT Heads and CLIP Latent Space	49
5.5	Automatic Removal of Spurious Correlations (PCLens vs TextSpan)	51
6	Explaining LVLM Image Tokens: Transferring Knowledge from CLIP ViT to LLaVA Image Features $F_{\text{image}}(I)$	55
6.1	Overall Setup	56
6.1.1	Models and Datasets	56
6.1.2	Assumptions	56
6.1.3	Key differences between CLIP Residual Stream of CLIP and LLaVA Feature Images	57
6.2	MLP Contributions Dominate the Image Features Residual Stream $F_{\text{image}[i]}(I)$	58

6.2.1	Interpretation of Mean Ablation (LLaVA vs CLIP)	58
6.2.2	Zero-Shot Accuracy Under Mean/Zero Ablation of MLP and MSA layers (LLaVA and CLIP)	59
6.2.3	Evolution of LVLM Text Output under Zero/Mean Ablation (LLaVA)	61
6.3	Do MSA and MLP Layer Contributions Attend to the [CLS] Residual Stream Differently?	64
6.4	Concluding Remarks on LLaVA Analysis	66
7	Discussion and Future Work	69
7.1	Dataset Considerations	69
7.2	ViT Interpretation and Intervention Methodology	70
7.3	LLaVA: Bridging ViT and Language Models	71
8	Conclusion	73
8.1	Conclusions	73
A	Appendix	75
A.1	New General Text Dataset D_I	75
A.2	Cosine Similarity of PCs with CLIP Image-Text Embeddings across different ViT Encoders	76
A.3	Non-Polysemantic Colour Heads across ViT Encoders (PCLens vs TextSpan)	77
A.4	Polysemantic MSA Heads across ViT Encoders (PCLens vs TextSpan)	78
A.5	PCs Directions across ViT Encoders (QuerySystem)	78
A.6	Comparison of Concept Reconstruction (<i>Party</i>) across ViT Encoders (QuerySystem and PCSelection)	79
A.7	PCs encode Semantic Meaning for both ViT MSA Heads and CLIP Latent Space (QuerySystem and PCSelection)	80
A.8	Optimal Number of PCs across the Methods for Removing Spurious Correlation (QuerySystem and PCSelection)	81
A.9	Prompt Enhancing by Visual Explanation (QuerySystem and PCSelection)	82
	Bibliography	91

Chapter 1

Notation

This chapter summarizes the symbols and notations used throughout the work and is intended to serve as a helpful reference during the reading.

1.1 Dimensions

d :	CLIP shared text-image embedding dimension.
d' :	ViT latent dimension (pre-projection).
N :	The number of patch tokens computed by a ViT when processing an image.
B :	The number of samples contained in one batch.
W, H :	The width and height of an image.

1.2 Model Architectures

Architectures

ViT:	A Vision Transformer [16] that processes images as a sequence of patches, each with latent dimension $\mathbb{R}^{d'}$. Different configurations exist (e.g. <i>ViT-B-32</i> , <i>ViT-B-16</i> , <i>ViT-L-14</i> , <i>ViT-H-14</i> , ...) that differ in patch size, number of parameters and layers.
CLIP:	Contrastive Language-Image Pre-training [56] model that aligns similar images and text in a shared embedding space of dimension \mathbb{R}^d .

1. NOTATION

LLaVA: Large Language and Vision Assistant [43], connecting a frozen vision encoder (e.g., CLIP ViT) to a large language model (LLM) (e.g Vicuna [54]).

Components

MLP: Multilayer Perceptron (MLP), a feedforward neural network consisting of fully connected multiple layers and a nonlinear activation function.

MSA: Multi-Head Self-Attention (MSA) [76], an attention mechanism employing multiple parallel attention heads to capture various representation subspaces simultaneously.

1.3 Concepts

$I \in \mathbb{R}^{3 \times W \times D}$: An arbitrary image.

t : An arbitrary text description.

c : An arbitrary concept, being either an image I or a text description t .

1.4 CLIP Model

CLIP Encoders

$E_{\text{image}}(I) \in \mathbb{R}^d$: The CLIP image encoder that maps an input image $I \in \mathbb{R}^{3 \times H \times W}$ into the shared embedding space \mathbb{R}^d .

$E_{\text{text}}(t) \in \mathbb{R}^d$: The CLIP text encoder that maps a text description t into the shared embedding space \mathbb{R}^d .

$e_c \in \mathbb{R}^d$: A CLIP concept embedding (derived from either text or image) \mathbb{R}^d .

CLIP Loss and Metrics

$\mathcal{L}_{\text{contrastive}} \in \mathbb{R}$: The contrastive loss function aligning similar image and text representations over a batch of size B .

$\text{cos_sim}(e_I, e_t) \in \mathbb{R}$: The cosine similarity metric between an image embedding $e_I \in \mathbb{R}^d$ and a text embedding $e_t \in \mathbb{R}^d$.

CLIP ViT Projections and Loss

$P \in \mathbb{R}^{d \times d'}$: A projection matrix that maps the ViT latent output from $\mathbb{R}^{d'}$ to the shared embedding space \mathbb{R}^d .

$\text{LN}(\cdot) \in \mathbb{R}^{d'}$: A layer normalization function expressed in affine form as $\text{LN}(x) = A x + b$ for $x \in \mathbb{R}^{d'}$, with parameters $A \in \mathbb{R}^{d' \times d'}$ and $b \in \mathbb{R}^{d'}$.

$P' = P A \in \mathbb{R}^{d \times d'}, b' = P b \in \mathbb{R}^d$: The combined effect of projection and layer normalization on the latent ViT output.

1.5 Datasets

D_I : A general image dataset subset of ImageNet’s validation set, uniformly sampled across all its 1000 classes. It contains N_I images

D_T : A general text dataset that represents a wide range of concepts. It includes five descriptive sentences per ImageNet class label plus five sentences for the 2,500 most common English words, generated in the style of CLIP prompts (“An image of ...”). It contains N_T text descriptions

$E_{D_I} \in \mathbb{R}^{N_I \times d}$: The set of image embeddings from D_I produced by CLIP image encoder $E_{\text{image}}(I)$.

$E_{D_T} \in \mathbb{R}^{N_T \times d}$: The set of text embeddings from D_T produced by CLIP text encoder $E_{\text{text}}(t)$.

1.6 ViT Residual Stream

$\text{ViT}(I) \in \mathbb{R}^{d' \times (N+1)}$: The hidden output of the ViT, expressed as a residual sum. Each column represents the embedding of a patch token and an extra [CLS] token at position 0.

$I_{[i]} \in \mathbb{R}^{d'}$: The initial embedding of the i -th image patch (after positional encoding).

$H_{[i]}^l \in \mathbb{R}^{d'}$: The output of the MSA block at layer l for patch i .

1. NOTATION

$M_{[i]}^l \in \mathbb{R}^{d'}:$	The output of the MLP block at layer l for patch i .
$H_{[i]}^{l,h} \in \mathbb{R}^{d'}:$	The output of the h -th attention head of the MSA block at layer l for patch i .
$\bar{M}_{[\text{CLS}]}^l \in \mathbb{R}^{d'}:$	The mean-ablated [48] MLP contribution across D_I for the [CLS] token at layer l . The initial [CLS] embedding is denoted as $\bar{M}_{[\text{CLS}]}^0$.
$A^{l,h} = \{P' H_1^{l,h}, \dots, P' H_{N_l}^{l,h}\} \in \mathbb{R}^{N_l \times d}.$	The projected activation matrix of the h -th MSA head at layer l for token [CLS] for all images in D_I .

1.7 LLaVA-Specific Notations

$F_{\text{image}[i]}(I) \in \mathbb{R}^{d_{\text{LLM}}}:$	The image feature for patch i extracted from the penultimate ViT layer and projected via a two-layer $\text{MLP}(\cdot)$ to align with the LLM’s embedding space.
$\text{MLP}(\cdot) : \mathbb{R}^{d'} \rightarrow \mathbb{R}^{d_{\text{LLM}}}:$	The two-layer multi-layer perceptron applied patch-wise in LLaVA to project visual tokens into the LLM’s embedding space.

1.8 Principal Component Analysis

$x_i \in \mathbb{R}^d:$	A data vector.
$X \in \mathbb{R}^{N_l \times d}:$	A data matrix whose rows are the vectors x_i .
$U \in \mathbb{R}^{N_l \times d}, V \in \mathbb{R}^{d \times d}, \Sigma \in \mathbb{R}^{d \times d}:$	The matrices from the Singular Value Decomposition (SVD) of the centered data matrix X_{centered} ; the columns of V (denoted p_k) are the principal components.

1.9 Explainability Methods

TextSpan

TextSpan:	A text-based explainability method that applies a Synchronous Orthogonal Matching Pursuit (SOMP) [73] algorithm to the activation matrix $A^{l,h}$ to retrieve M text
-----------	---

embeddings from $E_{D_T} \in \mathbb{R}^{N_T \times d}$ that best explain the variance in the head’s activation space, yielding human-interpretable characterizations.

PCLens

PCLens: A text-image explainability method that extracts principal components p_k from the activation matrices $A^{l,h}$ of each MSA head and characterizes both its directions using text E_{D_T} and image embeddings E_{D_I} .

$p_k \in \mathbb{R}^d$: A principal component derived from the PCA decomposition of an activation matrix $A^{l,h}$ that captures a semantic direction in the head’s latent space.

$f(p_k, e_c) \in \mathbb{R}$: A function that measures the similarity between a principal component p_k and a concept embedding e_c , quantifying their semantic alignment.

$K \in \mathbb{N}$: The number of principal components retained from the PCA decomposition explaining 99% of the variance of an activation matrix $A^{l,h}$.

$M \in \mathbb{N}$: The number of text (or image) embeddings retrieved per principal component direction p_k from each activation matrix $A^{l,h}$ by PCLens.

QuerySystem and PCSelection

QuerySystem: A procedure that, given a concept embedding $e_c \in \mathbb{R}^d$ and the set of all PCs R , computes the similarity for each p_k with e_c and returns a ranked subset of PCs across all MSA heads most relevant to the concept.

$R = \{c_1, \dots, c_{N_{PCs}}\} \subset \mathbb{R}^d$: The set of all principal components computed from the activation matrices of the final four MSA layers across all heads; N_{PCs} denotes the total number.

$S \subset R$: A selected subset of principal components identified by the QuerySystem as being most aligned with a given concept embedding e_c .

1. NOTATION

$s_k \in \mathbb{R}$: The score assigned to the k -th principal component, computed as the absolute cosine similarity between p_k and a concept embedding e_c .

PCSelection

PCSelection: A method that leverages a subset S of principal components related to a concept c to modify an image's residual stream by removing, selecting, or amplifying the PCs in S , thereby isolating or manipulating the residual stream representation.

$\tilde{E}_{\text{image}}(I) \in \mathbb{R}^d$: The modified CLIP image encoder output for an image I (i.e. the altered residual stream) after applying PCSelection with a subset S of PCs.

$\bar{\alpha}_k^{l,h} \in \mathbb{R}$: The k -th projection coefficient, averaged over the image dataset D_I , by expressing the activation value at MSA $P' H^{l,h}$ head h in layer l using the principal components basis of the activation matrix $A^{l,h}$ of that same head.

Chapter 2

Introduction

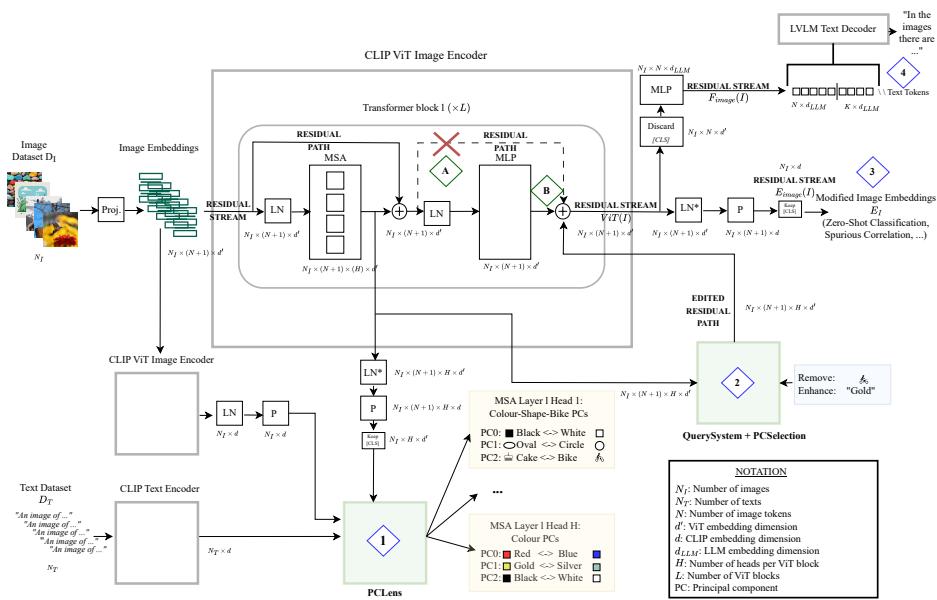


Figure 2.1: This figure provides a detailed overview of our work, highlighting our methods, serving as a reference throughout the entire paper, and illustrating the targeted edits we apply to both LLAVA and the CLIP image embeddings. We mark the critical points of our work with different numbers: (1) **PCLens**, (2) **QuerySystem + PCSelection**, (3) the modified residual stream $E_{image}(I)$ of the CLIP image encoder, and (4) the modified residual stream $F_{image}(I)$ of the LLAVA image features, where targeted edits are applied. Using letters, we indicate the following critical points in the $ViT(I)$ residual stream: (A) the standard MSA residual path followed during normal ViT computation, and (B) the modified MSA residual path after applying our edits using **QuerySystem + PCSelection**.

2.1 Motivation

Explanation and Interpretation of ML Models Understanding the role of specific and collective neural network components in constructing data representations is crucial for identifying model limitations, detecting underlying biases, and designing targeted interventions. This understanding directly relates to the concepts of explainability and interpretability in machine learning (ML), both of which aim to make ML model decisions comprehensible to humans, although in different ways [9]. Specifically, *interpretability* refers to internal model structures that are inherently understandable [14, 17]. This concept is more prominent in inherently simpler models, such as linear regression or decision trees, whose internal logic is clear by construction. Conversely, *explainability* involves additional post-hoc techniques designed to provide insights into the behavior of internal components behind a model’s decisions [11]. This is particularly suitable for more complex, opaque models like deep neural networks, whose internal information flow is not transparent by design.

Necessity of Explanation With the transition from simpler interpretable methods to powerful but more opaque multimodal deep learning architectures for most applications, ranging from natural language processing [78], computer vision [5], medicine [79], and many more, explainability has become increasingly critical. Despite their superior predictive capabilities, the internal mechanisms of these complex models remain poorly understood; they are frequently viewed as “black boxes” and rarely analyzed in depth. Explainability tools can significantly enhance model transparency and credibility, which is particularly important across numerous application areas. In these contexts, opacity poses substantial challenges, as relying solely on a model without comprehending its inner workings is insufficient. This is especially true in sensitive domains such as medical applications, where machine learning has been widely adopted yet remains insufficiently trusted as a standalone diagnostic tool, making human intervention essential [64, 80]. In such scenarios, trustworthiness, bias mitigation, and regulatory compliance are crucial considerations, therefore the explainability of the models [71]. Explaining model components naturally offers additional advantages, as a deeper understanding of their internal mechanisms can guide thoughtful improvements in model architectures by explicitly highlighting bottlenecks and facilitating the design of future models [7, 62]. In addition, specific neural network architectures can encode different inductive biases to varying degrees, often hidden behind architectural choices, and working toward their explainability can help uncover them. Due to these previous reasons, several approaches have been proposed to address the opacity of neural networks. These include general and well-known, architecture-independent methods such as saliency maps [69] and gradient-based visualizations [50], as well as

more architecture-specific approaches such as text-based interpretations [24], the main starting point of this work, for contrastive vision-and-language models such as CLIP [57]. Each method offers a distinct, often complementary, perspective into the inner workings of the model, aiming to improve human understanding and trust.

Multimodal Models: CLIP and LLaVA When focusing on novel multimodal models, there is still work to be done regarding their explainability. In recent years, contrastive models such as CLIP [57] have seen widespread adoption. This model projects multimodal inputs, text and images, into a shared embedding space through two separate encoders, commonly utilizing Vision Transformers (ViT) [16] as the vision encoder. The entire model is trained using a contrastive loss, explicitly aligning representations of related text-image pairs and naturally separating unrelated pairs within the embedding space via cosine similarity. CLIP’s popularity largely stems from its promising zero-shot transfer capabilities across various vision-and-language downstream tasks, such as zero-shot classification [57], segmentation [87], image generation [60], and integration as a fixed backbone in multimodal models [43]. One of their greatest relevance in recent years lies in this last point: downstream models for image-related tasks built on top of the frozen CLIP ViT encoder in combination with a learned adapter. A well-known example of such an application is LLaVA [43], a Large Vision-to-Language Model (LVLLM). This architecture employs the same frozen ViT encoder from the CLIP model to process an image and produce patch tokens, which are then fed to a patch-wise adapter. This has the effect of aligning image patch tokens with text tokens, which can then be passed to an explicitly fine-tuned LLM to enable the processing of multimodal inputs.

Explainability of CLIP ViT Encoder However, despite CLIP’s notable performance and widespread use, its embedding space and internal representations have not yet been fully explored or understood. In particular, even when focusing solely on its ViT encoder, relevant studies remain scarce. For example, questions related to how the ViT attends to and constructs each patch token [10], or which parts of the model attend to specific concepts [24], have only recently been partially addressed and are giving rise to new branches of explainability. Since its detached ViT is often used in various downstream tasks and is frequently the main component leveraged from the CLIP model, there is a clear need for continued investigation.

Text-Based Explainability of CLIP ViT Encoder: TextSpan Among the works toward the interpretation of the ViT encoder of CLIP, the seminal study by Y. Gandelsman et al. [24] proposes a different explainability method by leveraging the inherent linear residual structure [29] of the CLIP ViT

encoder, decomposing the final image-patch-embedding output (residual stream) as a linear sum of intermediate components, such as Multi-Head Self-Attention (MSA) [76] heads and Multilayer Perceptrons (MLP). This allows the final output to be treated as non-monolithic, splitting individual contributions as a linear sum of simpler MSA and MLP components. Notably, these contributions live in the same dimensionality as CLIP embeddings and can therefore be directly compared together using same dimensionality metrics. Their method, TextSpan, acts on the whole activation matrix of a head and provides a text-based characterization of the role of MSA heads in the last ViT layers in forming the final image representation. As a result, the text-characterized heads can be manually edited out to enhance performance in zero-shot classification tasks, such as spurious background removal.

2.2 Open Questions

Although the work of TextSpan has highlighted the significant role played by later-stage MSA heads in performance on CLIP downstream tasks, many attention heads in these later stages remain ambiguously characterized. This method primarily characterizes the activation output of each ViT attention head as a *whole*, failing to capture finer-grained details of ambiguous heads. Moreover, it is unclear whether the text explanations applied to the CLS token can be transferred to other patch tokens, whether image explanations are even meaningful, or why we can compare the components’ output with the final CLIP embedding, besides their shared dimensionality. In particular, several key questions remain open and warrant further investigation.

Semantic Ambiguity and Polysemaniticity Many of the characterized MSA heads still lack clear semantic meaning when examining their corresponding text embeddings. For example, ambiguity arises when trying to figure out a general concept from the text embeddings of such heads, as in Figure 2.2. This ambiguity partly stems from the global characterization approach employed by TextSpan, which does not explicitly account for heads potentially encoding multiple semantic concepts simultaneously, an issue thought to be caused by the mechanism of *superposition* [19], the strategy by which a neural network encodes more features than its available dimensions by overlapping them in the same activation space. The resulting phenomenon is called *polysemaniticity* [66], where individual neurons, due to this overlap, respond to multiple, seemingly unrelated features. This limitation is acknowledged by the original authors and left for future research.

Analysis of MSA Latent Spaces and Image Explanation The explanations and analyses regarding the latent space encoded by these attention heads are insufficiently detailed. Critical questions remain unanswered, such as

determining the exact semantic space spanned by their representations, their semantic specializations, and their alignment with the final latent space. Moreover, the question of whether using images together with text can enhance the characterization of the semantic space remains open.

Automation of Concept Identification Current methodologies utilized by Y. Gandelsman et al. [24], define heads’ role using manual labeling, by reading the text descriptions associated to it. Therefore, there’s need and space for methods that naturally automate the identification of specific semantic concepts encoded within and across heads, subsequently allowing targeted interventions.

Broader Implications for LVLM Models Lastly, it remains unclear whether a deeper characterization of these MSA heads can facilitate explanations of LVLMs, such as LLaVA, which make use of the same frozen CLIP ViT encoder. Particularly pertinent is the open question regarding the extent and nature of MSA contributions to tokens beyond the [CLS] token, which is used in CLIP zero-shot classification and where all the explanation analyses reside, as this aspect has not yet been thoroughly analyzed and are these other tokens the ones processed by LLaVA.

2.3 Contributions

Goal Our work is mostly related to mechanistic-style explainability of ViTs [77, 24], aiming to understand the role of their internal components in producing the final representation by explicitly tackling the polysemy of MSA heads. The goal is to answer the Open Questions 2.2, expanding upon the work of Y. Gandelsman et al. [24], by introducing a finer-grained method, PCLens (Figure 2.1 (1)), that improves both qualitative and quantitative benchmarks for explaining and analyzing the internal components of the CLIP ViT encoder using both textual and image embeddings. This method focuses on elucidating the role of MSA heads in reconstructing the final image embedding, explaining their latent space by characterizing principal components with both text and image embeddings based on cosine similarity, revealing highly specialized, semantically meaningful directions, distributed across the ViT residual stream, encoding same semantic meaning in both MSA heads and final embedding latent space. In addition, another method, QuerySystem, is developed to position and extract arbitrary text-visual concepts inside the ViT encoder, allowing automatic positional detection of the most important principal components. Together with QuerySystem, we develop a method, PCSelection (Figure 2.1 (2)), to directly manipulate specific sets of principal components, either removing or amplifying them. This strategy shows promising results on tasks such as the removal of spurious

2. INTRODUCTION

ViT-L-14 – Layer 23 – Head 4	
Two girls sitting together. A man holding a child. A young girl. A man sitting in a comfortable chair. A group of people with a guide.	TextSpan A woman holding a baby. A man standing next to a woman in a wedding dress. A dog with a friendly expression. A student holding a paper with an A+. A group of children laughing together.
Sisters. Twin animals. A pair walking side by side. Two trains coupling together. Two cars traveling one behind the other on a road. A group of five people. A group of people with a guide. A group of friends with backpacks. A group of young friends laughing. A small group of people talking on the side.	PCLens A young girl. A girl holding hands with another girl. A girl. A young female child. A younger and older girl. A mother holding her newborn in a hospital room. A mother's breast feeding an infant. A woman holding a baby. A mother animal nurturing a baby. An adult feeding a baby.
PC 0 (pair)	PC 6 (nurturing)
        	         

Figure 2.2: A comparison of the characterization of a polysemantic MSA head by *TextSpan* and our method *PCLens*, for ViT-L-14 Layer 23 Head 4. By examining the full text output of *TextSpan*, it is not trivial to identify a general concept explaining the role of the head. In contrast, by analyzing the role of each *individual* principal component (PC) characterized by *PCLens*, their role emerges. Each color highlights sentences corresponding to a different principal component, i.e., PC 0 (pair), PC 1 (girls), PC 5 (group of people), and PC 6 (nurturing). We also show additional image descriptions for PC 0 and PC 6 as naturally provided by *PCLens* to highlight the enhanced interpretability.

correlations (Figure 2.1 (3)). We then extend our investigation to VLM models by leveraging the previously analyzed frozen ViT encoder, with the goal to verify if the explainability knowledge can be transferred there (Figure 2.1 (4)). We find that, surprisingly, MLP contributions dominate the interpretation of image tokens in LVLMs, in contrast to the MSA heads, which dominate the [CLS] token. We provide evidence suggesting that the ViT used in LVLMs acts primarily as a tokenizer.

In summary, our contributions are the following:

- **PCLens, PC-based text-image-explainability:** Finer-grained semantic explainability of MSA heads using both text and image explanations associated with their principal components in both directions, explicitly targeting polysemy. We use a simple PCA-based method, mathematically supported, which achieves better quantitative zero-shot

reconstruction accuracy than TextSpan, with fewer text descriptions per head and arguably clearer qualitative semantic explanations of heads.

- **Polysemanticity, Semantics Alignment of Text-Image PCs and Latent Directions Analysis:** Using the text-image characterization from PCLens, we clearly demonstrate the presence of multiple semantic concepts simultaneously encoded within individual polysemantic heads. We also analyze the directions encoded by the PCs in both the head and the final CLIP latent space, showing semantic coherence between the two. We show that PCs encode both more (e.g., *black-and-white*) and less intuitive directions (e.g., *happiness vs. hand-craft*) and can be highly specialized on concept (e.g., *pair, nurturing, ...*). These findings reflect the complex and not always naturally explainable structure of CLIP’s final and internal latent space, which varies depending on the ViT size.
- **QuerySystem, Automated PC Localization:** We develop an automated and efficient method, based on the PCs decomposition of the residual stream for each MSA head, that localizes distributed PCs across multiple heads, corresponding to specific visual-text concepts. This has the effect of highlighting redundant and specialized encoding of concepts in different MSA heads of the ViT encoder. Through both quantitative and qualitative experiments, we demonstrate that in most cases only a small number of PCs suffice for effective concept representation.
- **PCSelection, Concept-level Targeted Interventions:** We develop a method that, given a set of PCs, is able to effectively act on their spectral space inside the ViT by enhancing or removing their contributions. Together with QuerySystem, we are able to automatically perform targeted interventions, without any manual labeling or finetuning, on the removal and enhancement of a specific concept c localized in different MSA heads of the ViT. Hence, we are able to improve zero-shot classification performance by removing spurious cues from datasets or by achieving better qualitative results using only an active subset of the model’s contributions.
- **Targeted interventions on LLaVA:** Lastly, we extend our intuitions to LVLM models, such as LLaVA, using the frozen ViT of the CLIP model. Our goal is to gain further insights into whether our previous analysis on the same ViT can be transferred to the LVLM case. We find that this is not the case, as the image-tokens used by LLaVA are more susceptible to slight changes in the MLP contributions compared to the MSA, opposite of the [CLS] token used in the zero-shot classification task performed by the vanilla CLIP.

The implementation code for the methods developed in this thesis, including PCLens, QuerySystem, and PCSelection, is available from the author upon reasonable request.

Chapter 3

Background

Understanding the inner workings of contrastively-trained ViT and multi-modal models is crucial for explaining their predictions and guiding interventions. In this chapter, we provide the necessary background to facilitate a smooth and engaging reading experience of this work. We begin with an overview of the CLIP [61] model and the mathematical foundations required to understand our method, focusing especially on the way its image encoder modifies the inherent residual stream output of the ViT [16]. Next, we focus our attention on the explainability work on the CLIP ViT Encoder of TextSpan [24], explaining the original method and consecutive extensions. Additionally, we turn our attention to the architecture of the LVLMs model LLaVA [43], which leverages the same frozen ViT encoder of CLIP. Here, we highlight an important architectural deviation: the projection and layer normalization layers present in the original CLIP model, through which the residual stream passes, are removed in LLaVA and replaced by a patch-wise adapter composed of two MLP layers. Additionally, the [CLS] token, commonly used in CLIP for classification, is not utilized in LLaVA, which instead makes use of all the other image tokens. This difference requires a different formulation of the residual stream for LLaVA’s image features and motivates the problem of a natural transition of explainability findings towards the LVLM. Lastly, we present a review of principal component analysis (PCA) with one of its most well-known applications, Eigenfaces [74], from which our method takes inspiration and which we complement with various explanation techniques used in machine learning. Rather than delving into implementation details, this section is intended to offer interpretative insights into the explanations produced by these models, helping the reader grasp the intuitions that guided the development of our work. Throughout, we progressively narrow our focus, from general concepts to specific architectures, culminating in ViT models trained with a contrastive loss, namely CLIP, and finally on LLaVA. We think that general explainability knowledge on ML

3. BACKGROUND

models is necessary to fully understand the work.

3.1 Contrastive Vision-Language Models

In this section we explain the foundation of contrastive vision-language models, with a focus on CLIP.

3.1.1 CLIP Architecture.

Contrastive Language-Image Pretraining (CLIP) is a vision-and-language model introduced by [56]. It consists of an image encoder $E_{\text{image}}(I)$ and a text encoder $E_{\text{text}}(t)$ that map images $I \in \mathbb{R}^{3 \times W \times H}$ and text descriptions t into a shared embedding space \mathbb{R}^d , with embeddings $e_I, e_t \in \mathbb{R}^d$, where the feature vectors can be directly compared. The image encoder $E_{\text{image}}(I)$ is typically based on a Vision Transformer (ViT) [16] or a ResNet [29] backbone, with an additional layer normalization $\text{LN}(\cdot)$ [1] and projection matrix $P \in \mathbb{R}^{d \times d'}$, while the text encoder $E_{\text{text}}(t)$ is a Transformer [76] with architectural modifications as in [58].

Training CLIP is trained on 400 million image–caption pairs to *align* pairs of image and text representations over batches of size B via a contrastive loss of the following form:

$$\mathcal{L}_{\text{contrastive}} = \frac{1}{B} \sum_{i=1}^B \left[-\log \frac{\exp \left(\text{sim}(E_{\text{image}}(I_i), E_{\text{text}}(t_i)) / \tau \right)}{\sum_{j=1}^B \exp \left(\text{sim}(E_{\text{image}}(I_i), E_{\text{text}}(t_j)) / \tau \right)} - \log \frac{\exp \left(\text{sim}(E_{\text{text}}(t_i), E_{\text{image}}(I_i)) / \tau \right)}{\sum_{j=1}^B \exp \left(\text{sim}(E_{\text{text}}(t_i), E_{\text{image}}(I_j)) / \tau \right)} \right] \in \mathbb{R} \quad (3.1)$$

where

$$\text{cos_sim}(e_I, e_t) = \frac{\langle e_I, e_t \rangle}{\|e_I\|_2 \|e_t\|_2} \in \mathbb{R}$$

is the cosine similarity between the embedding e_I of image I and the embedding e_t of text description t , $\tau \in \mathbb{R}$ is a learnable temperature parameter, and $B \in \mathbb{N}$ is the batch size. The loss encourages matching image–text pairs to have higher cosine similarity than mismatched pairs, which enables zero-shot image classification through cosine similarity by comparing an image’s embedding with embeddings of candidate class names or descriptions.

3.1.2 CLIP ViT Image Encoder $E_{\text{image}}(I)$

The CLIP ViT image encoder is based on a slightly modified ViT architecture. Formally, given an input image $I \in \mathbb{R}^{3 \times H \times W}$, a ViT produces an output $\text{ViT}(I) \in \mathbb{R}^{d' \times (N+1)}$, where d' is the ViT latent dimension and N is the number of patch tokens, with an additional [CLS] token prepended at position 0. Depending on the architecture, the output of the CLIP image encoder $E_{\text{image}}(I)$ used during the training procedure via $\mathcal{L}_{\text{contrastive}}$ is:

$$E_{\text{image}}(I) = \text{LN}(\text{ViT}(I)_{[\text{CLS}]}) \in \mathbb{R}^d \quad (3.2)$$

where LN is layer normalization [1] and $P \in \mathbb{R}^{d' \times d'}$ is a projection matrix specific to CLIP, projecting the ViT output into the shared text-image space.

The [CLS] token The learned [CLS] token enables a single-vector output from $E_{\text{image}}(I)$, which allows direct comparison through cosine similarity with the outputs of the text encoder $E_{\text{text}}(t)$. Crucially, during the optimization process, it is *this* token that is directly optimized with the contrastive loss $\mathcal{L}_{\text{contrastive}}$ in Equation 3.1. During training, the [CLS] token is optimized and processed by the ViT, propagating through the layers while attending to all other tokens during the MSA head computations, analogous to how BERT [13] uses a [CLS] token for sentences. This means that the extracted representation of the [CLS] token from the last layer of the ViT, $E_{\text{image}}(I)$, serves as a pooled semantic summary of the image content.

Applications Thanks to its aligned multimodal representations, the image encoder $E_{\text{image}}(I)$ achieves impressive performance across a wide range of tasks [87, 60, 56]. Relevant for us is also its frozen use in LVLMs such as LLaVA [43], which allows the alignment of image and text tokens. CLIP is available with various ViT configurations (e.g., ViT-B-16, ViT-L-14, ViT-H-14), differing in the number of parameters, patchification strategy, and image resize resolution. This flexibility makes CLIP suitable for a wide range of applications.

3.1.3 ViT Residual Stream $\text{ViT}(I)$

The ViT employed in the image encoder $E_{\text{image}}(I)$ of CLIP follows the standard Transformer architecture [76, 15] with MSA and MLP blocks. Each Transformer layer adds its output to a running residual stream via skip-connections[30], meaning that each individual patch token final embedding can be decomposed as a linear sum of the initial patch embedding plus contributions from each layer’s MSA and MLP layers. A visual explanation can be found at Figure 3.1.

3. BACKGROUND

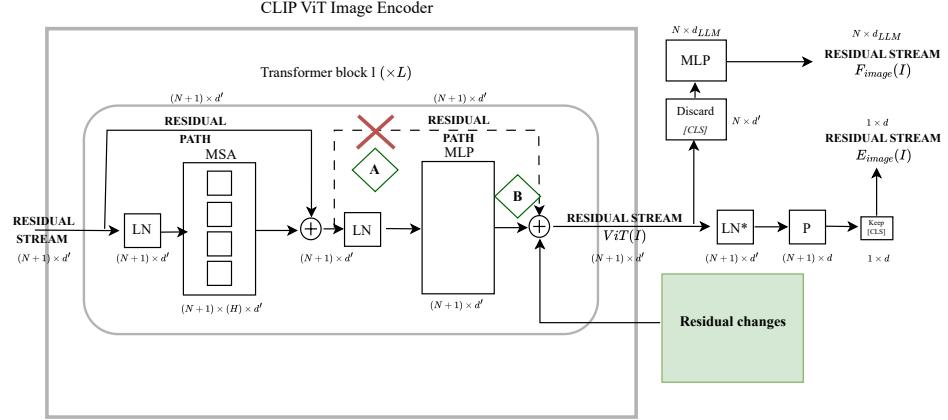


Figure 3.1: Figure showing the concept of residual stream inside the CLIP ViT image encoder, where additive contributions of MSA heads and MLP layers are summed together to produce a final embedding for each image-patch. We mark all three considered residual streams: $ViT(I)$, $E_{\text{image}}(I)$, and $F_{\text{image}}(I)$, which are recovered from different parts of the model pipeline and have different dimensionalities. Notice how the latter two depend on the first one. With letters (A) and (B), we mark the model’s locations where we respectively stop the normal residual stream flow and apply modifications using our methods `QuerySystem` + `PCSelection`.

Residual Stream The residual stream computation has same form for all the positional tokens in the images. Let i be the position of an arbitrary patch-token of the image I . When considering its final output embedding of a ViT, it can be decomposed as a linear combination of contributions from each MSA and MLP layers [15]:

$$ViT(I)_{[i]} = I_{[i]} + \sum_{l=1}^L H_{[i]}^l + \sum_{l=1}^L M_{[i]}^l \in \mathbb{R}^{d'}, \quad (3.3)$$

where $I_i \in \mathbb{R}^{d'}$ is the initial embedding of the image patch i (after positional encoding), $H_{[i]}^l \in \mathbb{R}^{d'}$ is the output of the MSA at layer l for patch i , and $M_{[i]}^l \in \mathbb{R}^{d'}$ is the output of the MLP block at layer l for patch i . This decomposition expresses the final image representation as a sum of per-layer contributions. Additionally, this equation can be further expanded in finer details, as noticed by the authors of [24], by unrolling the individual contribution per MSA head:

$$ViT(I)_{[i]} = [I]_{[i]} + \sum_{l=1}^L \sum_{h=1}^H H_{[i]}^{l,h} + \sum_{l=1}^L M_{[i]}^l \in \mathbb{R}^{d'}, \quad (3.4)$$

where H is the number of heads per MSA layer and $H_{[i]}^{l,h} \in \mathbb{R}^{d'}$ is the output of the MSA head h at layer l for the token i .

3.1.4 CLIP ViT Image Encoder Residual Stream $E_{\text{image}}(I)$

The main application of CLIP is the comparison of different text-image embedding through cosine similarity to determine their semantic alignment. When focusing on the output of its ViT image encoder $E_{\text{image}}(I)$, it is possible to further isolate and analyze the contributions that significantly influence the results of its task when evaluating cosine similarity with other embeddings. A visual explanation of the residual stream can be found at Figure 3.1.

Mean Ablation of MLP and early MSA layers First, we report that empirical evidence shows that mean-ablating [48] the MLP contributions on the ViT residual stream has minimal impact on zero-shot classification accuracy in CLIP [24]. Consequently, it is possible to focus on analyzing only the MSA heads’ outputs $H_{[\text{CLS}]}^l \in \mathbb{R}^{d'}$, especially in the last four layers of the ViT, since mean ablation of earlier layers MSA have been found to impact less significantly the model performances [24].

LN in affine form The layer normalization (LN) [1] can be treated as an affine transformation and therefore can be distributed linearly across the individual residual components. Formally, let $\text{LN}(x) = Ax + b \in \mathbb{R}^{d'}$ be the affine form of the layer norm where:

$$A = \frac{\gamma}{\sqrt{\sigma^2 + \varepsilon}} \in \mathbb{R}^{d' \times d'}, \quad b = -\frac{\gamma \mu}{\sqrt{\sigma^2 + \varepsilon}} + \beta \in \mathbb{R}^{d'},$$

where $x \in \mathbb{R}^{d'}$ is the input token, $\mu, \sigma \in \mathbb{R}$ are the mean and standard deviation of the token, and $\gamma, \beta \in \mathbb{R}^{d'}$ are learned vectors. Moreover, for the sake of conciseness, let $P' = PA \in \mathbb{R}^{d \times d'}$, $b' = Pb \in \mathbb{R}^d$ represent the absorbed effect of both the projection and the normalization on P .

Residual Stream We now revisit the output of the CLIP image encoder $E_{\text{image}}(I)$ (Equation 3.2) for an arbitrary image I . This depends on the decomposed ViT residual stream (Equation 3.4), the affine form of the LN and the projection matrix P .

Putting together our pieces we can rewrite the ViT encoder output $E_{\text{image}}(I)$

3. BACKGROUND

as:

$$\begin{aligned}
E_{\text{image}}(I) &= P \text{LN} \left(\text{ViT}(I)_{[\text{CLS}]} \right) \\
&= P \text{LN} \left([I]_{[\text{CLS}]} + \sum_{l=1}^L \sum_{h=1}^H H_{[\text{CLS}]}^{l,h} + \sum_{l=1}^L M_{[\text{CLS}]}^l \right) \\
&= P \left(A [I]_{[\text{CLS}]} \right) + \sum_{l=1}^L \sum_{h=1}^H P \left(A H_{[\text{CLS}]}^{l,h} \right) + \sum_{l=1}^L P \left(A M_{[\text{CLS}]}^l \right) + b' \\
&= P' [I]_{[\text{CLS}]} + \sum_{l=1}^L \sum_{h=1}^H P' H_{[\text{CLS}]}^{l,h} + \sum_{l=1}^L P' M_{[\text{CLS}]}^l + b' \\
&\approx P' [I]_{[\text{CLS}]} + \sum_{l=1}^L \sum_{h=1}^H P' H_{[\text{CLS}]}^{l,h} + \sum_{l=1}^L P' \bar{M}_{[\text{CLS}]}^l + b' \\
&= \sum_{l=1}^L \sum_{h=1}^H P' \textcolor{red}{H}_{[\text{CLS}]}^{l,h} + \sum_{l=1}^L P' \bar{M}_{[\text{CLS}]}^l + b' \in \mathbb{R}^{d'}, \tag{3.5}
\end{aligned}$$

where $\bar{M}_{[\text{CLS}]}^l \in \mathbb{R}^{d'}$ is the mean-ablation of the corresponding MLP output through a dataset of image D_I . This decomposition expresses the residual stream of $E_{\text{image}}(I)$ as a sum of individual component contributions from the MSA heads $\sum_{l=1}^L \sum_{h=1}^H P' H_{[\text{CLS}]}^{l,h}$ and the mean ablated MLP layers $\sum_{l=1}^L P' \bar{M}_{[\text{CLS}]}^l$. In the last line of the equation, for conciseness, we denote and regroup the term $P' [I]_{[\text{CLS}]} = \bar{M}_{[\text{CLS}]}^0$, since $[I]_{[\text{CLS}]}$ is learned and remains fixed across different images in the dataset D_I . Noticeably, only the red term $\textcolor{red}{H}_{[\text{CLS}]}^{l,h}$ for $l \in [L-3, L]$ (i.e., the last four layers) is not constant when performing zero-shot classification across different images, which is where we base all our analysis.

3.2 Textspan

Our work follow closely the seminal work of [24], which focuses on CLIP [57] text-explainability, exploiting the alignment between text and image representations to characterize internal components of the ViT encoder $E_{\text{image}}(I)$ of the CLIP model via textual descriptions D_T . In particular, the authors exploits the residual stream of the image encoder $E_{\text{image}}(I)$ as in Equation 3.5 where, thanks to their experiments on mean-ablating the MLP layers, the authors restrict their analysis to the MSA heads of the last four layers.

Method Consider the residual stream (Equation 3.5) for an arbitrary projected MSA head output $P' H_{[\text{CLS}]}^{l,h} \in \mathbb{R}^d$ for head h at layer l , which shares the same dimensionality d as CLIP’s representation space. The text-internal-components shared dimensionality can be leveraged to characterize the

function of this individual MSA heads. In particular, a candidate image dataset $D_I = \{I_1, \dots, I_{N_I}\}$ can be passed through the image encoder $E_{\text{image}}(I)$ to derive an output space $A^{l,h} = \{P'H_1^{l,h}, \dots, P'H_{N_I}^{l,h}\} \in \mathbb{R}^{N_I \times d}$ for head h in layer l . Then, a set of text embeddings $D_T = \{t_1, \dots, t_{N_T}\}$ is passed through CLIP’s text encoder $E_{\text{text}}(t) \in \mathbb{R}^d$ to produce a set of embeddings $E_{D_T} = \{E_{t_1}, \dots, E_{t_{N_T}}\} \in \mathbb{R}^{N_T \times d}$. At this point, `TextSpan`, analogous to the Simultaneous Orthogonal Matching Pursuit [73] (SOMP) algorithm [4], is used to select M text embeddings from E_{D_T} . These M selected textual descriptions aim to explain the most variance in $A^{l,h}$, providing a human-interpretable characterization of each MSA head’s activation space. This offers several downstream applications, such as mean-ablation of targeted heads to remove spurious correlations.

Observation This MSA head-level interpretability is enabled by CLIP’s multimodal shared space, where image features and text features reside in the same vector space \mathbb{R}^d . Hence, the authors can use language as a probe into the visual features. As a result of the analysis, certain attention heads exhibit a coherent role in image representation, i.e., M text explanations related to the same concept, such as encoding *location*, *colors*, or *animals* theme. However, for a considerable number of heads, a particular specialization does not emerge with this method, and thus a coherent interpretation of their role is not obtained. Hence, while this analysis yields valuable insights, the specific role of many MSA heads in the model remains unanswered. This leaves open the natural question of whether analyzing whole-head specialization is the optimal level of granularity for interpreting their mechanistic function and performing model interventions.

3.3 Large Vision-Language Models

In this section we present a brief overview of Large Vision-Language Models (LVLMs), with a focus on one of its open-source variant: LLaVA.

3.3.1 LLaVA

Beyond contrastive embedding models like CLIP, other classes of multimodal models integrate vision and language with a different goal: connecting vision encoders and large language models (LLMs) for general-purpose vision and language understanding. A prominent example is Large Language and Vision Assistant (LLaVA) [43], a LVLM that connects a pre-trained *frozen* CLIP vision encoder $E_{\text{image}}(I)$ [56] to a Large Language Model (LLM), specifically Vicuna [54], an instruction-tuned variant of LLaMA [86]. This entire system is jointly fine-tuned on instruction-following datasets, where each instruction includes an image-based task (e.g., “Describe the image” or visual question

3. BACKGROUND

answering). Consequently, LLaVA can interpret multimodal input questions involving both images and text. The adapter projecting image tokens output of $E_{\text{image}}(I)$ is implemented as a projection matrix in [43], or as an MLP in version 1.5 [42]. We consider this last version throughout our work.

3.3.2 Image Features Residual Stream $F_{\text{image}[i]}(I)$

In LLaVA’s architecture, an input image $I \in \mathbb{R}^{3 \times H \times W}$ is first encoded by the ViT into a sequence of visual tokens, from which hidden features are extracted at the penultimate layer:

$$\text{ViT}(I)_{[i]}^{L-1} = I_{[i]} + \sum_{l=1}^{L-1} H_{[i]}^l + \sum_{l=1}^{L-1} M_{[i]}^l \in \mathbb{R}^{(N+1) \times d'} \quad (3.6)$$

This choice is based on the hypothesis that the final layer of CLIP tends to emphasize global image properties, which may be less useful for capturing detailed local features [43]. Importantly, the [CLS] token is discarded., while keeping all the other N token. These visual features are then passed through a small per-token projection network, specifically a two-layer MLP, to align them with the embedding space of the LLM in the space $\mathbb{R}^{d_{LLM}}$. Subsequently, these projected visual tokens are fed into the LLM as a prefix to the text tokens [43]. Mathematically, considering the ViT residual stream (Equation 3.3) output for a patch token at position i , for $i \in [1, N]$, the corresponding image features $F_{\text{image}[i]}(I) \in \mathbb{R}^{d_{LLM}}$ passed to the LLM is extracted as follows:

$$F_{\text{image}[i]}(I) = \text{MLP}(\text{ViT}(I)_{[i]}^{L-1}) = \text{MLP}\left(I_{[i]} + \sum_{l=1}^{L-1} H_{[i]}^l + \sum_{l=1}^{L-1} M_{[i]}^l\right) \in \mathbb{R}^{d_{LLM}}, \quad (3.7)$$

where L is the number of ViT blocks and $\text{ViT}(I)_{[i]}^{L-1}$ indicates the residual stream extracted at the penultimate layer. A visual explanation can be found at Figure 3.1.

Observation Interpreting such models poses new challenges, since the image features are entangled with language generation inside the LLM. In this work, we focus primarily on the tokens-representation pre-MLP adapter, trying to transfer the interpretability from our analysis on the image encoder $E_{\text{image}}(I)$ towards LLaVA, since the two share the same frozen ViT backbone.

3.4 Principal Component Analysis and Eigenfaces

Many interpretation techniques leverage statistical tools to find structure in high-dimensional representations. A simple example of that is Principal Component Analysis (PCA), which also provides a basis to interesting explanation methods such as Eigenfaces.

3.4.1 Principal Component Analysis

Principal Component Analysis (PCA) is a classical dimensionality-reduction introduced by Karl Pearson in 1901 [53], that identifies the directions (linear combinations of original features) along which the data varies the most. Formally, given a dataset of K vectors $x_i \in \mathbb{R}^{d'}$ (e.g. the activation values of a MSA head), PCA finds an orthogonal basis of unit d_1 vectors $c_i \in \mathbb{R}^{d'}$, called principal components (PCs), such that the projected variance across the dataset of vectors $\text{Var}(x_i \cdot c_j)$ is maximized for the first principal component c_1 , second-maximized for the second principal component c_2 , etc. These vectors represent the true intrinsic linear dimensionality of the dataset, that is, the minimal number of variables needed to linearly approximate the data. Furthermore, when the data lies on a linear manifold, the intrinsic dimensionality matches the number of PCs required to fully explain its variance.

Method In practice, we let $X \in \mathbb{R}^{K \times d'}$ be the data matrix, where each row corresponds to a data vector x_i . After centering the data by subtracting the mean of each feature, we obtain the centered matrix $X_{\text{centered}} \in \mathbb{R}^{K \times d'}$. PCs can then be computed via Singular Value Decomposition (SVD) of X_{centered} , factorized as $X_{\text{centered}} = U\Sigma V^\top$, where $U \in \mathbb{R}^{K \times d_1}$ and $V \in \mathbb{R}^{d_1 \times d'}$ have orthonormal columns, and $\Sigma \in \mathbb{R}^{d_1 \times d_1}$ is a diagonal matrix with singular values in decreasing order. The columns of V , denoted by $c_j \in \mathbb{R}^{d'}$, correspond to the PCs, and the squared singular values on the diagonal of Σ^2 are proportional to the variance captured by each principal component, offering a natural ranking.

3.4.2 Eigenfaces

A classic example of PCA for interpretability is the Eigenfaces method used in face recognition [75]. The authors of this work show that by computing the principal components (PCs) of a data matrix consisting of a flattened collection of face images, it is possible to interpret the PCs visually as faces. These so-called eigenfaces *can* capture salient features such as overall lighting, pose, or specific facial parts. For instance, one principal component might encode the difference between a face with glasses and one without, while another might encode lighting direction. In general, eigenfaces combine multiple factors (e.g., lighting and pose together), and are not necessarily straightforward to interpret, although some directions may align well with human intuition. Any given face can then be approximately reconstructed as a linear weighted sum of these eigenfaces, resulting in a unique encoding for each image. This representation can help distinguish between faces and provide an intuitive understanding of how a face is linearly composed under this basis.

Inspiration In general, PCA has been a useful first step in analyzing neural representations for some applications [18, 35, 9]. The main problem for most applications is that the PCs are not interpretable in the same way as eigenfaces, especially when recovered through various network component activations, since they do not share the same dimensionality d with the output and/or involve non-linear processes, making the spaces not directly comparable. In our case, by examining the top PCs of activation values in MSA heads in the residual stream (Equation 3.5) and exploiting their inherent linearity, we are able to directly compare them with text and image embeddings, enabling a meaningful characterization.

3.5 Explainability of ML Models

The literature on explainability of ML models has seen significant progress in recent years, yet it remains a long-standing challenge [81]. Numerous *post hoc* explainability methods have been proposed, and selecting the most suitable approach depends on several factors, including modality, network architecture, interpretability goals, and computational resources. Generally, no universal method exists; instead, different methods provide complementary insights into the inner workings of specific models.

Motivation Even when narrowing the focus on image models there are already many different approaches. In particular, gradient-based saliency methods, such as Grad-CAM [67], which highlight important image regions based on model gradients; perturbation-based methods like occlusion sensitivity [84], which assess pixel importance by observing prediction changes after masking input regions; surrogate interpretability models, notably LIME [63], which approximate complex classifiers with simpler, locally interpretable models; and concept-based methods, such as TCAV [36], which identify high-level semantic concepts encoded in latent spaces.

Goal In this section, we focus on derived explanations rather than actual methods, from different scales of models, narrowing down in specificity, but with the same focus: the ViT encoder of CLIP and LLaVA. The goal is to provide background information and specific existing knowledge tailoring exactly the intuitions and assumptions used in our methods.

3.5.1 Neural Networks

Many phenomena have been observed concerning the explainability of neural networks. Notably, works on the phenomenon of Neural Collapse [38] in deep learning classification have shown that during training, last-layer features from the same class become nearly identical, while features from

different classes achieve maximal linear separability—a trend that intensifies from shallower to deeper layers [33]. Additionally, classification models trained and tested on the same dataset demonstrate greater resilience to feature ablation, supporting the hypothesis that features are redundantly encoded across layers for classification, with redundancy increasing in larger models [2]. Moreover, while models with different architectures may lack compatible weight spaces, ample evidence suggests that models with identical architectures often converge to the same weight basis [32], regardless of weight initialization and other hyperparameters, thereby permitting the merging of separately trained models with the same architecture and enabling combined model capabilities.

3.5.2 Explainability of ViT Models

ViTs [16] exhibit less inductive bias compared to Convolutional Neural Networks (CNNs) [51] since only the MLP layers are local and translationally equivariant, whereas the MSA layers operate globally [16]. In particular, pure MSA layers mix information across all tokens patches, contributing to global aggregation of information, while MLP layers process each token individually, amplifying and refining the details of contributions from the other components [77]. Visual progressive feature analysis in CNNs for vision tasks has been studied extensively for nearly a decade [69, 50]. Evidence suggests that ViTs exhibit similar behavior, with increasingly complex feature representations emerging through successive layers [26]. Additionally, it has been observed through feature visualization that ViTs interpret background information more effectively than CNNs, contributing to their performance, and that ViTs retain spatial information through all layers, except for the final layer, which acts as a learned pooling operator [26]. This indicates that ViTs likely maintain token-wise spatial understanding throughout most of the network, in contrast to CNNs, which progressively lose spatial granularity [72]. In addition, certain MSA heads specialize in class-specific features [27].

3.5.3 Explainability of Multimodal Models

Focusing on multimodal models, it has been hypothesized that semantic concepts are structurally organized in model’s latent space, sometimes linearly [6], with evidence that embeddings of different architectures can be linearly mapped onto one another [46]. Specifically, it has been shown the existence of linear mapping between image and text embeddings from arbitrary models [68]. This concept aligns with the Platonic Hypothesis [32], which posits that neural networks trained with different objectives on varied data and modalities converge toward a shared statistical model of reality in their representations. This hypothesis is further strengthened by the findings of [47], which discovered that the relative cosine similarity of embeddings

with shared anchors can produce similar embeddings across different neural networks trained on parallel datasets, enabling zero-stitching modules across models with differing dimensionalities. Other work explores comparing embedding spaces of different models via Latent Functional Maps, finding similar zero-stitching capabilities [23]. Additionally, it has been shown that multimodal models behave mostly like bags-of-words, independently of input sequence order [83].

3.5.4 Explainability CLIP

Modality Gap In the context of multimodal contrastive models like CLIP, additional considerations arise. For such models, an intrinsic multimodal gap has been observed at initialization in the embedded feature space, which surprisingly persists and often increases throughout training [41], enlarging differences between image and text embeddings. The gap implies that the model does not merge both modalities into a single cluster, as one would expect with contrastive learning, but preserves distinct distributions for images and texts. It is hypothesized that this gap helps preserving modality-specific information [59]. While some studies have attempted to reduce this gap for improved alignment [20, 22], others argue that maintaining a moderate modality gap can benefit downstream tasks by retaining richer modality-specific features, which naturally reflects distinct representations of reality [34, 20].

Latent Space Structure Some studies have highlighted the “cone projection” effect, where encoder-projected embeddings occupy a remarkably constrained region in the embedding space [20]. This phenomenon is especially prominent in the high-dimensional space of CLIP embeddings, where text-text correlations exceed text-image correlations within distinct regions on the unit sphere [6]. It has been noted that neural network layers reduce angular separation between embedding vectors with high probability, resulting in narrower cones at deeper layers [34]. A key feature of CLIP is its relatively angle-preserving embedding space, achieved through cosine similarity during training, which promotes angle-invariant representations [41]. Layer ablation analysis has identified the final attention layer as particularly influential in shaping CLIP’s image representations [24]. Furthermore, when trained as the backbone of CLIP’s image encoder, ViTs demonstrate the ability to learn features more related to abstract categories (e.g., music or morbidity) [26, 24]. Subsequent research using novel entanglement and association metrics has found larger CLIP models to be more interpretable, as their attention heads are more individually disentangled and linked to specific properties [44]. Further evidence shows that CLIP-like models often treat text inputs as bags-of-words, displaying limited sensitivity to word order in captions [22, 37]. Additionally, CLIP embedding spaces are asymmetric [34],

and mean-ablation experiments suggest that earlier layers contribute less to the output representations in the image encoder [24]. Moreover, it has been observed that the ViT in the CLIP encoder produces high-norm token artifacts during inference, mostly in low-informative background areas of images, which appear to serve the role of storing internal computations [10].

Localization of Concepts The presence of multimodal neurons in CLIP has been reported, individual neurons respond to specific high-level concepts across modalities, famously exemplified by a neuron responding to "Spider-Man" both visually and textually [27, 25]. Some later work provides evidence that many MSA heads in the ViT encoder have a clear semantic focus [24].

3.5.5 Explainability of LLaVA.

For novel LVLM models like LLaVA [43], explainability is still in its infancy. These models combine and align two complex components, a visual encoder and a language model, making it challenging to attribute a particular output token or behavior to visual features [70]. Most works focus on the explainability of the LLM part of the LVLM, i.e., after the visual tokens have been processed by the adapter [82, 49]. These tokens are not interpretable through vocabulary embeddings, and when object-specific tokens are removed, the model's accuracy drops significantly, highlighting the individual information contained in each token [49]. Some works find the presence of global-level features (e.g., the number of subjects) in unexpected token positions during the LLM processing, arguing that these are artifacts of the language model's computation rather than direct information from the visual tokens [10]. This observation is used to justify the worse performance of VLMs on image classification tasks compared to their backbone ViT used in CLIP [85]. For this use case, fine-tuning solely the adapter leads to better numerical stability and improved classification performance [85], suggesting that the adapter could be a source of the problem. Additionally, it has been observed that a small subset of image tokens in the LLM part, with high attention scores, are the primary drivers of object hallucination [8]. Other work introduces an adapter that learns to distill a handful of latent query tokens from the image patches, which are then passed to the language model, indicating that not all patch tokens are necessary for the language model [40]. Finally, it has also been observed that token embeddings passed through VLMs are treated primarily as a bag of tokens, meaning that permuting their order has little to no effect on downstream performance [55].

Chapter 4

Related Work

In this section we discuss similar papers and explain the positioning of our work in the field of ViTs text-interpretation expanding from the seminal work of [24] and on the explainability side of LLaVA focusing on the ViT side of the model.

4.1 Text Based Explanation of CLIP Image ViT Encoder

Related Work After Gandelsman et al. (2024) pioneered the text-based decomposition of CLIP’s ViT image encoder [24], subsequent works have expanded on this approach. Balasubramanian et al. (2024) generalized CLIP’s interpretability to arbitrary ViTs by mapping coarse internal features into CLIP’s text embedding space [3]. Madasu et al. (2024) quantified interpretability in CLIP-like models, measuring each head’s concept consistency and disentanglement using the TextSpan method [45]. Gandelsman et al. (2025) explicitly revealed finer neuron-level polysemy in CLIP [25], using text-based explanations. Basile et al. (2024) analyze CLIP ViT heads by decomposing their activations into a residual stream and studying the spectral properties of their principal components, demonstrating that early heads are more general while later ones specialize across datasets [4]. Using TextSpan, they highlight its analogy with SOMP [73], and show that, for some heads, it provides text features analogous to those obtained by applying OMP [52] to the first principal component. They then apply this knowledge to selectively fine-tune singular values to improve zero-classification, focusing primarily on aligning or pruning head-level representation.

Positioning Our method builds upon the work of TextSpan [24], using a mechanistic [77] and mathematically grounded framework that introduces a principled spectral characterization of later-layer CLIP ViT MSA heads.

4. RELATED WORK

Our approach differs fundamentally from prior methods that treat each head as a single, coarse feature extractor [24, 3]. Instead, PCLens focuses on finer-grained principal component analysis of the internal MSA activation space, explicitly addressing head polysemanticity [66]. Unlike existing approaches[4], our framework decomposes each head’s role into multiple interpretable directions that are explained through both text and image modalities, revealing that these directions preserve semantic meaning consistently across both local MSA heads and the final CLIP embedding latent space, compared to other existing approaches only focusing on few PCs and characterizing them with text using OMP [52]. No prior methods also complement the explanation of MSA heads using image embeddings. We advance beyond tools like TextSpan not only by enhancing the explanation of both polysemantic and unisemantic MSA heads, but also by enabling precise, automated control over their latent space. We uncover rich, interpretable semantic directions spanning intuitive (e.g., *black-and-white*), unintuitive (e.g., *happiness-handcraft*), and abstract concepts (e.g., *pair*, *nurturing*). Moreover, using QuerySystem, we identify minimal sets of PCs responsible for encoding specific vision-text concepts across different MSA heads, on which we can act using PCSelection, enabling targeted removal or enhancement of PCs and concepts directly at the representation level without requiring fine-tuning, thereby facilitating both interpretability and mitigation of spurious correlations. Finally, our framework generalizes across ViT models, revealing shared latent space organizational patterns and concept specialization trends in different architectures.

4.2 Explanation of ViT Patch Tokens Effects on LLaVA

To the best of our knowledge, there are currently no studies that explicitly address the interpretation of CLIP-ViT patch tokens before they are processed by LLaVA’s adapter. Some works, however, have explored the explanation of these tokens after the MLP token-wise adapter has been applied in this setting, see Section 3.5.5.

Chapter 5

Explaining the Latent Space of CLIP ViT Encoder $E_{\text{image}}(I)$ MSA Heads Using Text and Images

In this chapter, we present the methods and experiments employed in the *first part of our work*: enhancing interpretation and explanation of the latent space of the MSA heads in the ViT encoder $E_{\text{image}}(I)$ of CLIP using both text embeddings E_{D_t} and image embeddings E_{D_I} .

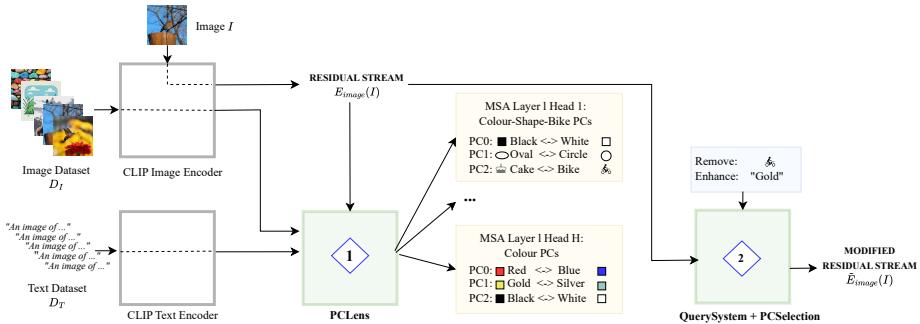


Figure 5.1: In this figure, we highlight our methods: PCLens (1), which interprets principal components using both text and image descriptions, and QuerySystem + PCSelection (2), which apply targeted edits on the residual stream $E_{\text{image}}(I)$ of an image reference I , by removing or enhancing concepts.

5.1 Overall setup

This section outlines the models, datasets, and assumptions used throughout the rest of the chapter.

5. EXPLAINING THE LATENT SPACE OF CLIP ViT ENCODER $E_{\text{IMAGE}}(I)$ MSA HEADS USING TEXT AND IMAGES

5.1.1 Models and Datasets

We conduct our experiments on four CLIP-based ViT models with different patch and model sizes: ViT-B-32, ViT-B-16, ViT-L-14, and ViT-H-14. Depending on the experiment, we evaluate our results on three datasets: ImageNet [12], CIFAR-10 [39], and Waterbirds [65], which manually combines photographs of waterbirds and landbirds with alternating water or land backgrounds. Regarding the qualitative examples, unless specified, the model of reference is the ViT-L-14.

5.1.2 Goal

Our goal is to explain the CLIP ViT encoder $E_{\text{image}}(I)$ residual stream (Equation 3.5) through the interpretation of the intermediate activation matrix of each MSA head in the last four layers, $A^{l,h} = \{P'H_1^{l,h}, \dots, P'H_{N_l}^{l,h}\} \in \mathbb{R}^{N_l \times d}$, for every head h and layer l , across a dataset $D_I = \{I_1, \dots, I_{N_l}\}$ of N_l generic images, where d is the dimensionality of the final CLIP embedding. For this purpose, we make use of CLIP’s text embeddings, $E_{D_T} \in \mathbb{R}^{N_T \times d}$, obtained from a dataset $D_T = \{t_1, \dots, t_{N_T}\}$ of N_T generic text descriptions by passing a sentence dataset through CLIP’s text encoder $E_{\text{text}}(t)$. Additionally, we make analogous use of the image embeddings $E_{D_I} \in \mathbb{R}^{N_l \times d}$ from the dataset D_I , obtained from CLIP’s image encoder $E_{\text{image}}(I)$. Specifically, the two datasets are as follows:

- A general *image dataset* D_I , which we select as a 10K-image subset from ImageNet’s validation set, uniformly sampled across 1,000 classes.
- A general *text dataset* D_T , designed to represent a wide range of concepts. It consists of five *descriptive* sentences for each ImageNet class label, along with 5 sentences for the 2,500 most commonly used words in English. These sentences are generated using *GPT-4o* [21] and phrased in the style of CLIP prompts (e.g., “An image of ...”), ensuring alignment with CLIP’s training distribution. Further details are provided in Appendix A.

5.1.3 Assumptions

For the remainder of this section, we work under the following assumptions:

- **Last 4 MSA layers:** Based on the mean-ablation experiment conducted by [24] on the residual stream (Equation 3.5), it has been shown that only the outputs of the attention heads in the final few MSA layers significantly influence the accuracy of model’s output when considering zero-shot classification. Consequently, we *assume* that these last four layers are enough to capture most of the semantic information of the

5.2. PCLens: Explaining CLIP Image Encoder $E_{\text{image}(I)}$ MSA Heads through their PCs Directions

CLIP image embedding. That meanst that all of our analysis refer to the residual stream of the image encoder $E_{\text{image}}(I)$ at Equation 3.5.

- **General Image Dataset D_I and Text Dataset D_T :** We *assume* that both the ImageNet subset used as D_I and our constructed text dataset D_T are sufficiently broad and various to cover most of the basic concepts that may be encoded by the network. This assumption is necessary to ensure that we are capturing a representative set of interpretable concepts that we can use to explain our individual contribution and already supported in other similar works [24, 4].

5.2 PCLens: Explaining CLIP Image Encoder $E_{\text{image}(I)}$ MSA Heads through their PCs Directions

In this section, we present the details of our method, PCLens, which aims to address the open question related to the polysemanticity of MSA heads that do not exhibit a clear or consistent meaning when their activation space is interpreted using text-based methods. Specifically, we aim to provide a method that explicitly tackles the potential polysemanticity of certain heads, which are activated by different concepts. We hypothesize that the problem with TextSpan lies in analyzing the heads as a *whole*, using multiple texts; in the case of polysemantic heads, this makes it difficult to identify a clear and consistent pattern. Hence, we attempt to address this issue by proposing a *finer-grained* interpretation of MSA heads, achieved by isolating and interpreting the individual principal component directions (PCs) in their output space. These PCs are characterized by associating them with multiple texts and *enhanced* by also incorporating image descriptions in the shared CLIP embedding space, applied to both their directions. The method is highlighted at position (1) at Figure 5.1.

Method We consider the image dataset D_I . For each MSA head h in each layer l , we store their individual activation outputs across D_I to compute the projected activation space $A^{l,h} \in \mathbb{R}^{N_l \times d}$, storing the data row-wise. We make use of the text embeddings $E_{D_T} \in \mathbb{R}^{N_T \times d}$ from D_T and $E_{D_I} \in \mathbb{R}^{N_l \times d}$ from D_I . We then apply PCLens (Algorithm 1) to decompose the activation space $A^{l,h}$ of each MSA head into PCs, sorted by their singular values. Each PC is then interpreted by associating both its direction with the top M most and least similar text embeddings E_{D_T} based on cosine similarity. Additionally, we associate $2M$ image embeddings from E_{D_I} in analogous way, yielding to a total of $4M$ text-image descriptions per PC.

Mind the Gap As discussed in Section 3.5.4, text and image embeddings in CLIP are known to be clustered in distinct regions of the embedding space.

5. EXPLAINING THE LATENT SPACE OF CLIP ViT ENCODER $E_{\text{IMAGE}}(I)$ MSA HEADS USING TEXT AND IMAGES

Algorithm 1 PCLens: Characterizing the role of a MSA head h for layers l through PC decomposition

Require: Activation matrix $A^{l,h} \in \mathbb{R}^{N_l \times d}$ for head h in layer l , text embeddings $E_{D_T} \in \mathbb{R}^{N_T \times d}$, image embeddings $E_{D_I} \in \mathbb{R}^{N_I \times d}$, number of texts-images per PC M

- 1: Perform PCA decomposition on $A^{l,h}$ and retain PCs $p_1, \dots, p_K \in \mathbb{R}^d$.
- 2: **for** p_i in p_1, \dots, p_K **do**
- 3: Retrieve M text embeddings in E_{D_T} and M image embeddings in E_{D_I} with maximal/minimal cosine similarity to p_i
- 4: **end for**

To address this misalignment, when reconstructing the activation space $A^{l,h}$, we mean-center each image embedding using the mean of E_{D_I} , similarly as the authors of [24].

5.2.1 Metric of Choice: Cosine Similarity

We propose cosine similarity between a given PC $p_k \in \mathbb{R}^d$ and a generic CLIP concept embedding $e_c \in \mathbb{R}^d$ as a metric for characterizing the PC and measuring the similarity between the two. While this choice seems to be motivated by CLIP’s training objective, which directly optimizes the cosine similarity between image and text embeddings, it is not immediately evident that such metric would also be meaningful when comparing *internal representations* (e.g., PCs p_k of MSA activation matrices) to *external embeddings* (e.g., e_c).

Motivation To motivate our choice, consider a setting in which the CLIP ViT encoder $E_{\text{image}}(I)$ produces an image embedding $e_I \in \mathbb{R}^d$ of an arbitrary image I , which is then compared to a set of text and image embeddings $E_{D_C} \in \mathbb{R}^{(N_I+N_T) \times d}$ using cosine similarity. This happens during both training and inference (e.g., in zero-shot classification), hence the resulting latent space is naturally structured to work under that metric. We hypothesize that any arbitrary individual PCs $p_k \in \mathbb{R}^d$ of the MSA activation matrices $A^{l,h}$, which captures dominant semantic directions in the activation space, can be aligned and compared with an arbitrary CLIP embeddings $e_c \in E_{D_C}$ in a way that reflects its internal organization of semantic information. Therefore, we aim to ensure that such comparison between PC p_k and CLIP embedding e_c via cosine similarity is already present and enforced during training, thereby providing evidence that the two can be directly compared using such metric. Moreover, we need to ensure that the resulting value of the metric has an interpretable meaning when building the resulting values of the overall training objective between the image e_I and the other representations. What we aim to find is a contribution that depends on both, similarity of the image

5.2. PCLens: Explaining CLIP Image Encoder $E_{\text{image}(I)}$ MSA Heads through their PCs Directions

embedding e_I with the PC p_k and the similarity of the PC p_K with the concept embedding e_c .

Derivation In what follows, we propose a candidate scoring function of the form $\alpha^{l,k} f(p_k, e_c) \in \mathbb{R}$, which quantifies the contribution of each PC p_k to the semantic alignment between an image embedding e_I and a concept embedding e_c . The scalar $\alpha^{l,k} \in \mathbb{R}$, the score, is the projection coefficient obtained by expressing the [CLS] token output $P' H_{[\text{CLS}]}^{l,h} \in \mathbb{R}^d$ (from head h at layer l) as a linear combination of the PCs of its same activation matrix $A^{l,h} \in \mathbb{R}^{N \times d}$ for image I . Intuitively, if the image embedding e_I projects positively onto p_k (i.e. its corresponding coefficient $\alpha^{l,k}$ is positive) and if $f(p_k, e_c)$ is also positive, then the contribution of the function increases the cosine similarity between e_I and the CLIP concept embedding e_c . Conversely, a negative value of $f(p_k, e_c)$ would decrease this similarity and so on for the other cases. Hence, the role of the function $f(p_k, e_c)$ can be interpreted as that of a similarity metric between p_k and the concept embedding e_c , serving as a measure of semantic alignment between the principal component p_k and e_c . We find the presence of such $f(p_k, e_c)$ in the following derivation on the optimized cosine similarity between an image I and the concept embedding e_c :

$$\begin{aligned}
& \cos_sim(E_{\text{image}}(I), e_c) \\
&= \frac{E_{\text{image}}(I) \cdot e_c}{\|E_{\text{image}}(I)\| \|e_c\|} \\
&\approx \frac{(\sum_{l=1}^L \sum_{h=1}^H P' H_{[\text{CLS}]}^{l,h} + \sum_{l=0}^L P' \bar{M}_{[\text{CLS}]}^l + b') \cdot e_c}{\|E_{\text{image}}(I)\| \|e_c\|} \\
&= \frac{\sum_{l=1}^L \sum_{h=1}^H (P' H_{[\text{CLS}]}^{l,h}) \cdot e_c + \sum_{l=0}^L (P' \bar{M}_{[\text{CLS}]}^l) \cdot e_c + b' \cdot e_c}{\|E_{\text{image}}(I)\| \|e_c\|} \\
&\approx \frac{\sum_{l=1}^L \sum_{h=1}^H P' \sum_k \alpha_k^{l,h} (\mathbf{p}_k \cdot \mathbf{e}_c) + \sum_{l=0}^L (P' \bar{M}_{[\text{CLS}]}^l) \cdot e_c + b' \cdot e_c}{\|E_{\text{image}}(I)\| \|e_c\|} \tag{5.1}
\end{aligned}$$

Starting from the cosine similarity definition, we substitute the residual stream representation of $E_{\text{image}}(I)$ (as in Eq. 3.5), apply the linearity of the dot product to decompose the expression into additive components, and finally employ a linear approximation of $H^{l,h}$ using the PCs from its activation matrix $A^{l,h}$.

Looking at the terms in red in the last line of the equation, we notice that we can extract the composed function $\alpha^{l,k} f(p_k, e_c) \in \mathbb{R}$, depending only on p_k and e_c and which has the exact role as described above:

$$f(p_k, e_i) = \frac{p_k \cdot e_i}{\|e_i\|} = \frac{p_k \cdot e_i}{\|e_i\| \|p_k\|} = \cos_sim(p_k, e_i)$$

5. EXPLAINING THE LATENT SPACE OF CLIP ViT ENCODER $E_{\text{IMAGE}}(I)$ MSA HEADS USING TEXT AND IMAGES

The penultimate step follows from the fact that p_k is already normalized.

5.2.2 Recovered Zero-shot accuracy (PCLens vs TextSpan)

In this section, we aim to demonstrate the validity and numerical superiority of PCLens over TextSpan. We show that we can approximate the activation space of the last four layers’ MSA heads, $A^{l,h} \in \mathbb{R}^{N_l \times d}$ —the only image-dependent part of the output—by the space spanned by the top k text embeddings found by our method. The idea is to use these reconstructed values within the residual stream (Equation 3.5) to verify that we can effectively approximate it. We evaluate the performance of PCLens in terms of both zero-shot classification accuracy and the number k of text descriptions required per MSA head, and we compare it with TextSpan. Intuitively, we aim to show that the text embeddings found by our method are able to capture the most significant variation in the image representation at the head level, meaning they correctly approximate the principal components hence the activation matrices $A^{l,h}$.

Setup We reconstruct each of the matrices $A^{l,h}$ using a varying number k of text embeddings. For TextSpan, this corresponds to using a total of k texts per head, following their original experimental setup. For PCLens, to ensure a fair comparison, we use the top *single* text embeddings most correlated with the first k principal components. We evaluate the resulting reconstructions by directly substituting them into the residual stream of each image embedding and measuring zero-shot classification performance on the ImageNet dataset D_I .

Results The results for the different ViT models are shown in Figure 5.2. As illustrated by the heatmap, PCLens reaches a comparable zero-shot baseline accuracy more quickly than TextSpan, thereby supporting the validity of our approach. Intuitively, because fewer text embeddings are required to capture the relevant variance in the heads’ output representations than with TextSpan, our text directions align more closely with the variations in the matrix activation space. These results also indicate that the text embeddings produced by our method are well aligned with the principal components, as they can efficiently approximate the matrix.

Low Heads’ Space Dimensionality Fewer text embeddings are sufficient to reconstruct the output space of the activation matrices $A^{l,h}$, compared to their true dimensionality \mathbb{R}^d . For instance, in the case of ViT-L-14, only 60 text directions are needed, despite the embeddings having a dimensionality of $d = 768$. This suggests that the MSA heads operate in a lower-dimensional subspace, likely encoding compressed or specialized semantic concepts. Interestingly, using only 40 principal components already yields a meaningful

5.2. PCLens: Explaining CLIP Image Encoder $E_{\text{image}(I)}$ MSA Heads through their PCs Directions



Figure 5.2: HeatMap of recovered zero-shot accuracy of TextSpan vs PCLens on different ViT Models. Notice how PCLens converges faster with a smaller number of text-embeddings than TextSpan towards a similar zero-shot baseline accuracy.

approximation of the activation space, achieving 73.11% over 73.88% accuracy for ViT-L-14, hinting that the first few PCs already explains lots of the activation space variance. This result is further supported in Figure 5.3, which shows the mean cumulative variance explained per PC across the MSA heads of the last four layers of the different ViT models. Notice that there is no clear trend in the amount of variance explained across ViTs of different sizes, except for the fact that as few as 40 principal components can already explain up to 90% of the variance.

PCs cosine similarity across ViT Models and layers We aim to show the value of enhancing descriptions of PCs using both image and text embeddings. Figure 5.4 shows a heatmap visualizing the cosine similarity between PCs and both text E_T and image embeddings E_I for ViT-L-14. Additional plots for the remaining models are provided in the Appendix A.2. Despite the presence of *modality gap*, illustrated at the bottom of the figure with the cosine similarity of 10 random permutations of images and texts with each other, clearly that image and text embeddings are inherently more similar within their respective modalities than to their cross-modal counterparts, we

5. EXPLAINING THE LATENT SPACE OF CLIP ViT ENCODER $E_{\text{IMAGE}}(I)$ MSA HEADS USING TEXT AND IMAGES

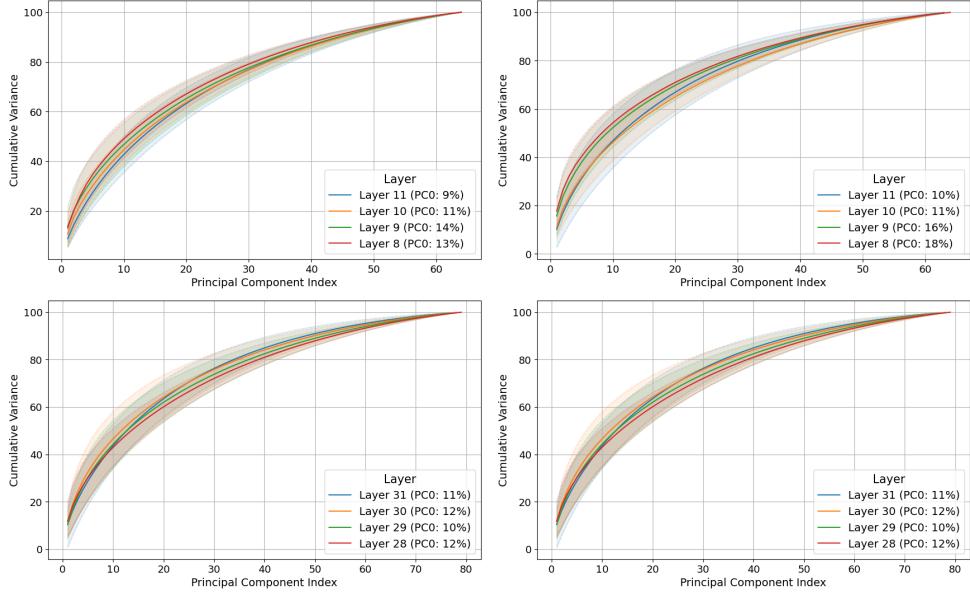


Figure 5.3: Mean cumulative variance explained per PC across heads of the last four layers of the different ViT models. The shaded region represents ± 1 standard deviation, and legends indicate the percentage of variance captured by the first principal component (PC0) per layer.

find similar values of cosine alignment between PCs and both image and text embeddings in later layer. Notable is the high text similarity of layer 21 with the first PC. We will also find in the next Chapter 6 that this layer plays an interesting role in the refinement of image-tokens, presenting a strong alignment with the final residual stream, and suggesting the presence of a strong refinement of these tokens.

5.2.3 (Non)-Polysemantic Heads Role (PCLens vs TextSpan)

In this section we derive a qualitatively comparison on the explainability of individual MSA heads using PCLens and TextSpan, highlighting the superiority of our method in disentangling ambiguous polysemantic heads, even when considering only text-embeddings.

Comparison The results for polysemantic heads of ViT-L-14 are reported in both Figure 5.5 and Figure 2.2. Additional comparison for different ViT sizes are reported in Appendix A.3 (non-polysemantic head) and Appendix A.4 (polysemantic head). TextSpan is designed to find text-embeddings that are captured by the variance of the *whole* head. In this example, for TextSpan, it seems that this head does not relate to a specific concept. However, PCLens demonstrates that this head in fact relates to multiple coherent concepts that

5.2. PCLens: Explaining CLIP Image Encoder $E_{\text{image}(I)}$ MSA Heads through their PCs Directions

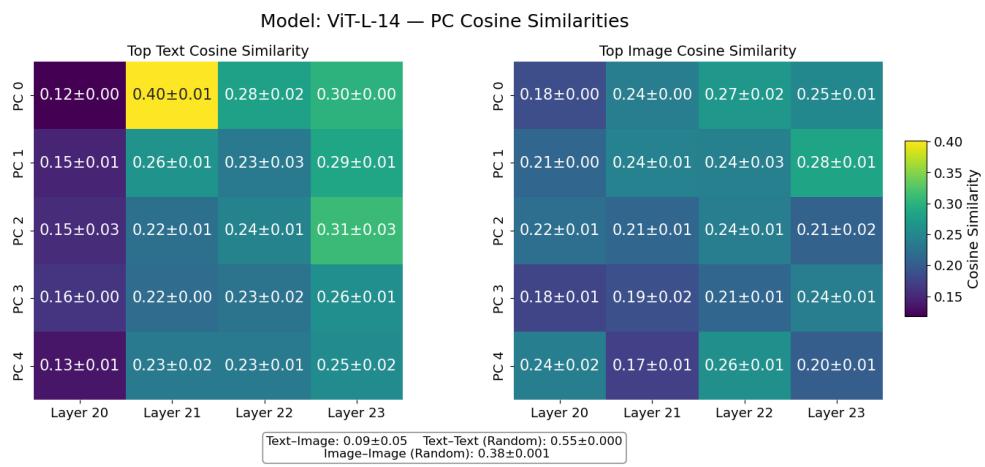


Figure 5.4: Cosine similarities of the top 5 PCs from ViT-L-14 across different layers, computed separately for text embeddings (left) and image embeddings (right). Each cell reports the mean cosine similarity along with the standard deviation across the top ten samples. The analysis includes the last 4 relevant layer, highlighting variations in embedding alignment at deeper layers of the ViT. Reference baseline cosine similarities for final embeddings pairings are provided below for comparison: text-image similarity, random text-text similarity, and random image-image similarity.

are encoded in different directions of variance. PCLens has the ability to disentangle these concepts by decomposing MSA activation space $A^{l,h}$ with different principal components and characterizing them instead. This way, PCLens reveals that this head in fact relates to multiple coherent concepts that are encoded in different directions of variance, such as *white* (PC0), *silver* (PC1), *shadow* (PC2) and *glass/transparent* (PC3) for MSA layer 22 head 14, and *pair* (PC0), *young girl* (PC1), *group* (PC5) and *nurturing* (PC6) for MSA layer 23 head 4. This enhanced explainability highlights the advantage of our method in isolating semantically meaningful directions within the representation space. Moreover, the explainability is clearly enhanced with images explanations of PCs, aligned with the text-descriptions.

Non-Polysemantic We also report an example of a non-polysemantic *Colour* MSA head in Figure 5.6, where we provide additional examples of these specific *color* heads across different ViT models in Appendix . Arguably, even in this case, PCLens provides a qualitatively better explanation of the role of this color head when examining individual PCs, with both text and image, compared to TextSpan. In particular, it is easier to identify a general pattern of the head role when considering a finer-grained decomposition. A.3.

5. EXPLAINING THE LATENT SPACE OF CLIP ViT ENCODER $E_{\text{IMAGE}}(I)$ MSA HEADS USING TEXT AND IMAGES

ViT-L-14 – Layer 22 – Head 14	
A white and silver vehicle. A sunlight casting shadows on a roof. A stainless steel item. A gray and black animal. A clear glass object.	TextSpan A blue and silver object. A brown and white dog. A black and gold object. A grandfather reading a book with glasses. A dark sky filled with stars.
A white and silver vehicle. A brown and white dog. A dog with a white and brown coat. A white and brown dog. A dog with a white and red coat. A sunlight casting shadows over a roof. A cloud with the sun shining behind it. A shadow cast by an object. Shadows cast on a wall. Long shadows cast by trees.	PCLens A shiny silver object. A silver object. A silver-colored metal bar. A shiny silver faucet. A silver trinket. A clear glass object. A clear glass pane. A clear glass. A clear plastic item. A clear soup.
PC 1 (silver)	PC 2 (shadow)
       	       

Figure 5.5: A comparison of TextSpan with PCLens as tools to interpret the role of ViT-L-14 Layer 22 Head 14. For PCLens, each color highlights sentences corresponding to a different principal component, i.e. PC 0 (blue), PC 1 (green), PC 2 (orange), PC 3 (yellow). We also provide image-description of PCLens associated with PC 1 (green) and PC 2 (orange), with emerging concepts of *silver* and *shadow* respectively.

5.2.4 Heads’ Finer-grained Latent Space

In this section, we discuss qualitative results demonstrating the capabilities of PCLens in uncovering and disentangling concepts encoded in the MSA heads. We provide evidence of the explainability benefits of expressing intermediate representations with text-image characterized PCs, revealing insights into the semantic of the latent structure of CLIP ViT encoder representations. Moreover, we show that image representations associated to PCs via PCLens seems to exhibit a similar semantic conceptual structure as the text embeddings, hinting a cross-modal alignment of representations. Further, we will demonstrate the presence of highly specialized PCs directions.

Polysemy Our results provide clear evidences that MSA heads are capable of encoding multiple concepts within their subspace as in Figure 5.5 and Figure 2.2, providing strong evidences for the strategy of superposition [19] in large transformer models. PCLens effectively disentangles these concepts, offering a powerful tool for understanding the latent structure of attention heads and the semantic relationships encoded within.

5.2. PCLens: Explaining CLIP Image Encoder $E_{\text{image}(I)}$ MSA Heads through their PCs Directions

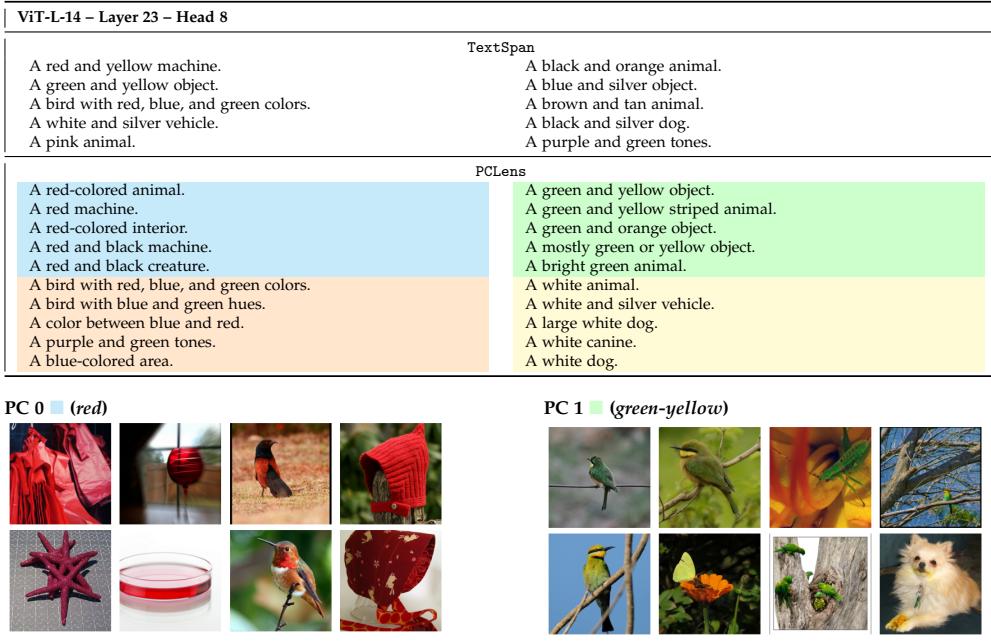


Figure 5.6: A comparison of TextSpan with PCLens for ViT-L-14 Layer 23 Head 8. For PCLens, each color highlights sentences corresponding to a different principal component, i.e. PC 0 , PC 1 , PC 2 , PC 3 , with displayed image-descriptions for PC 0 and PC 1 .

Cross-Modal Alignment of Representation In addition to revealing concepts encoded within the heads of the CLIP image encoder, PCLens also shows hints at cross-modal representational alignment across the different MSA encoder heads. For instance, PC1 of the ViT-L-14 for Layer 22 Head 14, aligns with representations of *silver* from both text and image at Figure 5.5. Additionally, PC2 of the same head aligns to representations of text and images about *shadows*. The fact that the PCs have consistent visual and textual descriptions qualitatively strengthen the validity and interpretability of our method PCLens. Other examples are noticeable in Figure 2.2 for Layer 23 Head 4, which reveal similar dynamics across PC0 and PC6 for the concepts of *pair* and *nurturing*.

Contrastive Directions, Specialization and Abstract Concepts Our findings reveal that principal components are highly specialized. As an example, we can traverse the direction of PC0 in MSA Layer 22 Head 12 in Figure 5.7, where we observe that even opposite ends of a principal component axis can be used to encode distinct concepts, not aligning with human intuition of opposite in this case. PCLens further reveals that the attention heads not only encode simple concept, but also abstract ones. For instance, Figure 2.2

5. EXPLAINING THE LATENT SPACE OF CLIP ViT ENCODER $E_{\text{IMAGE}}(I)$ MSA HEADS USING TEXT AND IMAGES

ViT-L-14 - Layer 22 - Head 12 - PC 0



Figure 5.7: An example of traversal of PC0 of Layer 22 Head 12, where its positive and negative characterization showing meaningful latent space directions of *forest* and *ocean* concepts.

highlights components encoding *pair* and *nurturing* concepts, while Figure 5.8 shows components encoding abstract concepts such as *eating*, *failure*, and *infant*. These results emphasize the capacity of ViT models to capture complex and abstract relationships within their latent spaces, which is especially true for ViT trained with contrastive loss [26].

Layer 22 - Head 9 - PC 0 ('eating')	Layer 22 - Head 15 - PC 3 ('mistake')	Layer 22 - Head 4 - PC 1 ('infant')
A spoon near to a mouth. A spoon of soup lifted to the mouth. Someone savouring a bit of dessert. A tongue tasting food. Food with a bite taken out.	A broken item repaired. Representing quitting or leaving. People giving up their seat on a bus. A computer with an error message. A broken item being repaired.	A baby. An infant taking their first steps. A baby with big, curious eyes. A child taking their first steps. A baby crawling on the floor.

Figure 5.8: Examples of highly specialized PCs in ViT-L/14 across different MSA heads, highlighting the concepts of *eating*, *mistake* and *infant*.

Specialized and Contrasting PCs Directions Across ViT (woman-man) We provide additional qualitative examples across different ViT models to highlight the generalization capability of our approach, PCLens, and to showcase interesting findings in the characterization of specific PCs. Surprisingly, we consistently observe the presence of *woman* and *man* concepts encoded along a single PC, as shown in Figure 5.9. This indicates that in the latent space of these heads, the concepts lie on opposite sides along this direction. Beyond the ethical implications of a latent space where these directions are repre-

5.3. QuerySystem: Automatic Location of Concepts in CLIP ViT Encoder $E_{\text{image}}(I)$

sented as opposites, it is noteworthy that this concept emerges consistently across all model sizes. These findings are complemented by additional results in Appendix A.5, which examine the alignment with human intuition, such as the oppositional concept of *black–white*, as well as cases of misalignment, such as with *happiness* and other concepts, differing across ViT models.

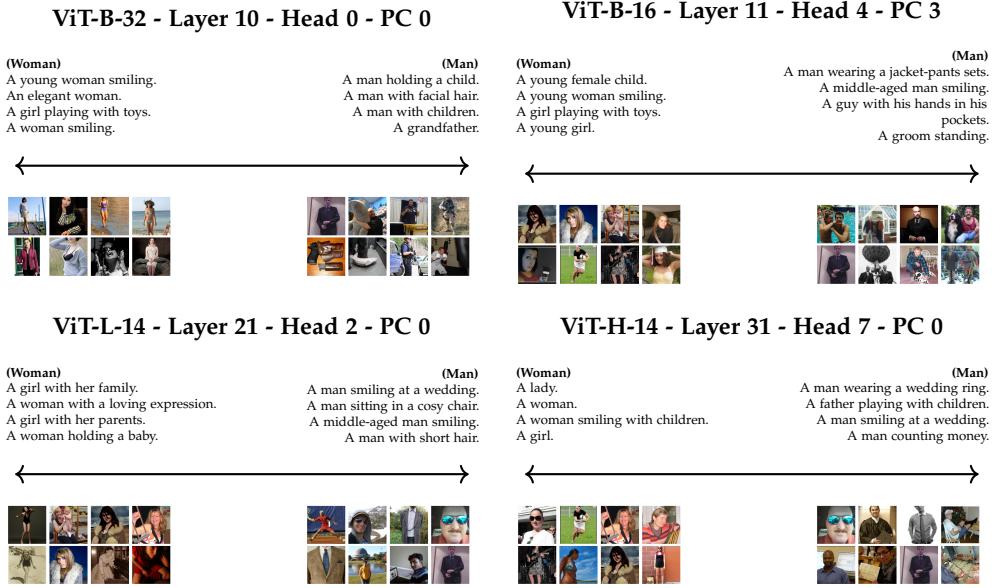


Figure 5.9: Principal components across different CLIP ViT models encoding the concepts of *woman* and *man*. Interestingly, this concept is present in all models, although with slightly different representations, and consistently encoded along a single PC. This indicates that in the latent space of these heads, the concepts lie on opposite sides along this direction.

5.3 QuerySystem: Automatic Location of Concepts in CLIP ViT Encoder $E_{\text{image}}(I)$

In this section, we present our method for identifying the locations within the CLIP ViT encoder where principal components predominantly contribute to the alignment with an arbitrary concept c . Once these aligned PCs are identified, we trace them back to their corresponding MSA layers to determine which attention head is most responsible for encoding the concept. This procedure not only clarifies the role of each attention head in establishing the alignment between any image and the concept c , but also enhances the overall explainability of the model. The method is highlighted at position (2) at Figure 5.1.

5. EXPLAINING THE LATENT SPACE OF CLIP ViT ENCODER $E_{\text{IMAGE}}(I)$ MSA HEADS USING TEXT AND IMAGES

Algorithm 2 Query System: Identifying the most aligned PCs in CLIP ViT Encoder $E_{\text{image}}(I)$ with a concept c

Require: $e_c \in \mathbb{R}^d$ text or image embedding , $\{A^{l,1}, \dots, A^{l,H}\} \subset \mathbb{R}^{H \times N_l \times d}$ activation matrices of the last four MSA layers

- 1: Score = []
- 2: **for** layer l in $L - 3, \dots, L$ **do**
- 3: **for** head h in $1, \dots, H$ **do**
- 4: Decompose $A^{l,h}$ to find top K first principal components p_1, \dots, p_k
- 5: Compute scores s_1, \dots, s_K via the (absolute) cosine similarity of each PC with the concept embedding e_c
- 6: Append s_1, \dots, s_K to Score
- 7: **end for**
- 8: **end for**
- 9: Determine top R scores across Score, and corresponding top PC directions p_1^*, \dots, p_R^*

Method Given an arbitrary concept c , being either text (e.g. "An image representing a heart", "An image representing a failure") or an image, we compute its embedding $e_c \in \mathbb{R}^d$ using respectively the CLIP image encoder or the text encoder. Given the joined set of all the PCs $R = \{c_1, \dots, c_K\}$, with $p_k \in \mathbb{R}^d$, across all the MSA heads $A^{l,h} \in \mathbb{R}^{N_l \times d}$ of the last 4 layers, we employ the procedure described by Algorithm 2. This method traverses the residual stream (Equation 3.4), and indirectly the individual MSA heads contributions, to localize the set S of R most absolute cosine similar PCs with e_c , possibly distributed across different heads, which intuitively builds up the contribution towards that concept c .

5.3.1 Metric of Choice: Absolute Cosine Similarity

We make use of the absolute cosine similarity between a PC $p_k \in \mathbb{R}^d$ and the concept embedding $e_c \in \mathbb{R}^d$ to rank and compare each PCs across all MSA heads. We motivate our choice of such metric in the following paragraph by following a similar approach as the selection of cosine similarity for PCLens.

Motivation Our approach is based on decomposing the cosine similarity between the residual stream (Equation 5.1) of the embedding e_I of arbitrary image I and a concept embedding e_c , where such metric is naturally optimized on during CLIP training. Since each MSA heads output can be approximated by its weighted sum of PCs, the whole residual stream can be decomposed as a weighted sum of all the PCs, which captures in finer-grained details the image contributions toward that concept. Hence, we aim to compare the contributions of each individual PCs to each others, as they capture a continuous sum of values in \mathbb{R} .

Derivation We focus on the cosine similarity measure of an arbitrary image I and a concept embedding e_c . As shown in the residual stream Equation 5.1, the overall alignment can be decomposed into two terms:

- A term representing the sum of the alignments of individual PCs p_k with the concept e_c , where each contribution is weighted by a score $\alpha_k^{l,h}$ (with l indicating the layer, h the head, and k the k th PC).
- A constant term across all images (accounting for the mean ablations of the MLPs and first MSA layers), which is independent of the PCs.

Since our goal is to compare the relative contributions of the PCs, we disregard the constant term and all terms constant between their comparison. Furthermore, to ensure generality over all images, we focus on the image-independent part of the alignment. This means we do not consider the image-specific scaling factors associated with the PCs via $\alpha_k^{l,h}$ in our comparison, setting it to a constant value of one. Hence we are left with the following term:

$$\sum_{l=1}^L \sum_{h=1}^H \sum_k \frac{p_k \cdot e_c}{\|e_c\|} = \sum_{l=1}^L \sum_{h=1}^H \sum_k \text{cos_sim}(p_k, e_k)$$

The last step comes from the fact that p_k is already normalized.

This leaves a last problem; we have characterized the PCs with two distinct concepts, one in its positive direction and one on the negative. Hence, the PC can be aligned with the concept c when it takes both a high positive value or a low negative value. Absolute cosine similarity captures both strong positive and strong negative correlations, ensuring relevant PCs are identified regardless of directionality. Therefore, we make use of this when ranking the individual PC contribution. This means we are left with the following scoring function

$$\sum_{l=1}^L \sum_{h=1}^H \sum_k |\text{cos_sim}(p_k, e_k)|$$

We use this measure to compare the individual PCs between each other into `QuerySystem`.

5.4 PCSelection: Targeted Interventions on the ViT Encoder

In this section we present PCSelection at Algorithm 3. This method explains how a set of principal components,

$$S = \{c_1, \dots, p_k\} \subset \mathbb{R}^d,$$

as identified by the `QuerySystem`, selected components can be used to modify the image residual stream to induce meaningful changes in its representation

5. EXPLAINING THE LATENT SPACE OF CLIP ViT ENCODER $E_{\text{IMAGE}}(I)$ MSA HEADS USING TEXT AND IMAGES

Algorithm 3 PCSelection: Targeted actions on the PCs of CLIP image encoder $E_{\text{image}}(I)$ residual stream

Require: Set of PCs $S \subset \{p_1, \dots, p_K\}$, mean-ablated coefficients $\{\bar{\alpha}_k^{l,h}\}$ per MSA head, image $I \in D_I$, CLIP ViT encoder $E_{\text{image}}(I)$, Action $A \in \{\text{retain}, \text{remove}, \text{amplify}\}$

- 1: **if** $A = \text{retain}$ **then**
- 2: Keep all $\bar{\alpha}_k^{l,h}$
- 3: **else if** $A = \text{remove}$ **then**
- 4: Set $S = S^c$
- 5: **else if** $A = \text{amplify}$ **then**
- 6: Set $\bar{\alpha}_k^{l,h} = v \cdot \alpha_k^{l,h}$, $v \in \mathbb{R}$
- 7: **end if**
- 8:
- 9: Compute residual stream $E_{\text{image}}(I)$ from the CLIP encoder.
- 10: **for** $l = 1$ to L **do**
- 11: **for** $h = 1$ to H **do**
- 12: Decompose head activation $A^{l,h} = \sum_k \alpha_k^{l,h} p_k$
- 13: **for** $k = 1$ to \dots **do**
- 14: **if** $p_k \in S$ **then** $\alpha_k'^{l,h} = \alpha_k^{l,h}$ **else** $\alpha_k'^{l,h} = \bar{\alpha}_k^{l,h}$ **end if**
- 15: **end for**
- 16: Reconstruct modified activation $A'^{l,h} = \sum_k \alpha_k'^{l,h} p_k$
- 17: **end for**
- 18: **end for**
- 19: Rebuild modified residual stream:

$$\tilde{E}_{\text{image}}(I) = \sum_{l=1}^L \sum_{h=1}^H P' A'^{l,h} + \sum_{l=0}^L P' \bar{M}_{[\text{CLS}]}^l + b'$$

without any fine-tuning. This process can be used to retain, remove, or amplify the PC contributions (directions of variance) from S , in order to analyze and manipulate the output representation of the image, $E_{\text{Image}}(I) \in \mathbb{R}^d$. This is particularly suited for CLIP’s primary application: performing various tasks based on cosine similarity with CLIP embeddings. This approach enables targeted interventions and further enhances the explainability of the CLIP ViT encoder. We focus specifically on the PCs associated with the last four MSA layers, while the remaining components of the residual stream are mean-ablated, as described in Equation 3.2. The method is highlighted at position 2 at Figure 5.1.

Method We consider the residual stream as described in Equation 3.4 for a given image $I \in D_I$. Let again $S = \{p_1, \dots, p_k\} \subset \mathbb{R}^{K \times d}$ denote

5.4. PCSelection: Targeted Interventions on the ViT Encoder

a selected set of PCs across all MSA heads, and let its complement be $S^c = R \setminus S$, where R is the set of all PCs. In general, we can perform the following operations on individual PC contributions to the residual stream: *retain, remove, amplify (diminish)*. Amplifying (diminishing) is straightforward, achieved by multiplying the contribution of each PC p_k by a scalar value $v \in \mathbb{R}$. Removing or selecting PCs works as follow. To remove the contributions of principal components in the complement set $S^c = R \setminus S$ (while retaining those in S), we process each $p_k \in S^c$ individually. For each, we consider the MSA activation output $A^{l,h} \in \mathbb{R}^{N_l \times d}$ and apply a *mean ablation* [48] strategy: we project the dataset mean (computed over D_I) onto p_k , retaining only this projection. The remaining components in S are preserved by projecting the original activation $A^{l,h}$ onto their respective directions at their corresponding heads.

Following these steps, PCSelection produces a modified residual stream $\tilde{E}_{\text{image}}(I)$:

$$\begin{aligned}
 E_{\text{image}}(I) &= \sum_{l=1}^L \sum_{h=1}^H P' H_{[\text{CLS}]}^{l,h} + \sum_{l=0}^L P' \bar{M}_{[\text{CLS}]}^l + b' \\
 &\approx \sum_{l=1}^L \sum_{h=1}^H P' \sum_k \alpha_k^{l,h} p_k + \sum_{l=0}^L P' \bar{M}_{[\text{CLS}]}^l + b' \\
 &\approx \sum_{l=1}^L \sum_{h=1}^H P' \left(\sum_{k \in S} \color{red}{\alpha_k^{l,h}} p_k + \sum_{k \in S^c} \bar{\alpha}_k^{l,h} p_k \right) + \sum_{l=0}^L P' \bar{M}_{[\text{CLS}]}^l + b' \\
 &= \tilde{E}_{\text{image}}(I) \in \mathbb{R}^d
 \end{aligned} \tag{5.2}$$

Here, $\bar{\alpha}_k^{l,h} \in \mathbb{R}$ denotes the mean score across the dataset D_I per MSA head activation $H^{l,h}$. The only image-dependent component is the score for the selected PCs in S (highlighted in red), which is where we intend to act when amplifying or diminishing the signal. The modified residual stream $\tilde{E}_{\text{image}}(I)$ can then be used in cosine similarity computations as a normal CLIP embedding:

$$\text{cos_sim}\left(\tilde{E}_{\text{image}}(I), e_c\right) = \frac{\tilde{E}_{\text{image}}(I) \cdot e_c}{\|\tilde{E}_{\text{image}}(I)\| \|e_c\|} \tag{5.3}$$

PCSelection + QuerySystem A straightforward application of PCSelection is to use the set of PCs S identified by the QuerySystem to retain, remove or act only on those PCs associated with a specific concept when reconstructing the final output of the CLIP image encoder. In this way, it is possible to remove, amplify, or diminish the contribution of a concept without any finetuning and in an *automatic* way.

5. EXPLAINING THE LATENT SPACE OF CLIP ViT ENCODER $E_{\text{IMAGE}}(I)$ MSA HEADS
USING TEXT AND IMAGES



Figure 5.10: Comparison of top images retrieved for the text concept *heart* retrieved using the first 75 PCs found by QuerySystem, across the various CLIP ViT backbones (rows). In each row, the left column (Full CLIP embedding) shows the top-k normal CLIP most similar image embeddings for the text concept, and the right column (First 75 PCs) shows the corresponding images when using the modified residual stream produced by QuerySystem +PCSelection. We also provide the percentage of active PCs across the active heads of the last four layers.

5.4.1 Qualitative Reconstruction of Concept

In this section we provide qualitative results showing how our `QuerySystem` indeed finds a subset of PCs which are related to a concept c . We provide examples of the results obtained with this procedure, showing that only a few PCs are sufficient to encode a concept, even if it is out-of-distribution from the image dataset D_I .

Setup We use the reference image dataset D_I and reconstruct each image embedding residual stream (5.2) following PCSelection, using the set of PCs S selected for a given concept c from the `QuerySystem`. The resulting reconstructed embedding $\tilde{E}_{\text{image}}(I)$ is compared through cosine similarity with the original concept embedding e_c (Equation 5.1). Intuitively, if the PCs in the set S encode sufficient information about the concept, then the reconstruction of the images where these directions are dominant will still preserve most of their information. This means that if using only a small set of PCs S , distributed across different heads, we still retrieve images associated with the corresponding concept (e.g., images of hearts matching the description "An image of a heart concept"), we have evidences that we are successfully localizing relevant PCs for the concept c within the ViT encoder.

Result The results are shown in Figure 5.10. We observe that the resulting outputs using the 75 PC found by `QuerySystem` are arguably better and more specialized than those of the final CLIP model; when using fewer PCs compared to the standard CLIP approach, we can localize smaller instances of the concept (e.g., a small heart on a teddy bear), which remains undetected by the full model. This specialized localization of concept is consistent across every ViT model, where retaining fewer PCs for the concept *heart* tends to highlight small occurrences of hearts in the images. Notably, CLIP zero-shot accuracy remains effective even when using only a small percentage of active principal components across the active heads of the already mean-ablated model. This provides evidence that our method, `QuerySystem`, indeed identifies meaningful PC directions related to the target concept and that MSA heads are highly specialized. By comparing different model sizes, we can also use this querying process to explore how concepts are encoded in smaller vs. larger models, which in this case does not provide a clear pattern. In Appendix A.6, we provide additional comparison between different ViT model size in reconstructing the concept of *party*.

5.4.2 PCs encode Semantic Meaning for both ViT Heads and CLIP Latent Space

In this section, we aim to show that the directions associated with the PCs found by `QuerySystem` have the same semantic meaning both in their local

5. EXPLAINING THE LATENT SPACE OF CLIP ViT ENCODER $E_{\text{IMAGE}}(I)$ MSA HEADS USING TEXT AND IMAGES

MSA latent space and in the final CLIP embedding. For our experiment, we use the set of PCs S identified by `QuerySystem`, with the goal to demonstrate that manipulating these directions using `PCSelection`, either at the local MSA level or at the global residual stream level, induces meaningful changes in the latent space by either *enhancing* or *forgetting* the concept.

Setup We conduct an ablation study on each class label of ImageNet. We fix one label l at a time (e.g., "goldfinch"), align it with the CLIP description (e.g., "An image of a goldfinch"), and query increasing sizes k of the set S of PCs found by `QuerySystem`. This allows us to localize the positions of PCs encoding the concept l within the various MSA heads of the CLIP image encoder. We then either *remove* or *retain*; when intervening directly within the ViT, this involves following `PCSelection` to either preserve the internal contributions from PCs in S at their respective heads (to retain the concept) or mean-ablate these PCs while retaining all others (to remove the concept). In the case of acting on the full residual stream, this corresponds to either projecting it onto the subspace spanned by the selected PCs S (to retain the concept) or subtracting this projection from the residual stream (to remove the concept). In both settings, we quantitatively evaluate performance using a zero-shot accuracy test, reporting both the overall accuracy over ImageNet and the accuracy on the manipulated labels. The results are averaged over all the individual label tests. Intuitively, if the concept is effectively removed, the total accuracy should degrade slightly, while the accuracy for label l should drop to near zero. Conversely, when retaining only the PCs associated with label l , we expect low overall accuracy but near-perfect performance on the targeted class. This also depends on the choice of k ; if a concept is not fully encoded within the selected k PCs, then it will not be effectively enhanced or removed.

Result The results of the experiment are shown in Figure 5.11, based on the single-label experiments. Additional complementary plots related to individual labels, including more details on the standard deviation across experiments, are provided in Appendix A.7.

Examining both sets of results, it is clear that in both approaches the identified PCs directions are indeed responsible for encoding a given class. Removing them slightly decreases the overall accuracy but completely removes the concept with a restrained number of PCs, depending on the model size; conversely, retaining them allows correct classification of the class label. Notably, when acting on the final CLIP embedding, a small number of PCs (e.g., 45 for ViT-L-14) is enough to recognize the class concept with 90% accuracy, and slightly more (e.g., 65 for ViT-L-14) is sufficient to forget it in 95% of the cases. When directly modifying the ViT encoder, a similar pattern emerges, but with larger numbers required: approximately 1250

PCs to forget and 900 to correctly classify the concept in ViT-L-14. In both cases, forgetting appears to be more difficult, likely due to redundancy, where concepts are encoded across multiple directions. For instance, a “heart” could still be recognized from its red color and shape even without the specific “heart” concept being present. Moreover, in both approaches we observe a similar trend: in larger models, the identified PCs are more precise, likely more specialized, as fewer of them are needed to retain concept-specific information. However, redundancy is also more evident in larger models, making it harder to forget a concept, as indicated by the slower decay in accuracy. This may also be a consequence of the increased capacity of larger models. Interestingly, this is supported by the observation that the overall accuracy on other classes remains higher when more PCs are removed in larger models, again suggesting redundancy in information representation. We also observe a “sweet spot” when retaining a concept: for certain values of k , accuracy on the targeted label exceeds the baseline and surrounding values. As k increases further, misclassification begins to rise again. Another noteworthy observation is that information stored in MLP components is more relevant in larger models; when mean-ablating all components in the MSA heads, larger models still manage to correctly classify a greater portion of the dataset. Lastly, when considering the CLIP final embedding, fewer PCs (i.e. smaller values of k) are needed to effectively act on the concept c . The reason can be motivated by the fact that we act on the whole residual stream at once, projecting out/in contributions for PCs. That means that if a concept is encoded by a similar PC directions p_k in more heads, it is effectively filtered out all at once with, requiring less PCs. When using a larger number of PCs, the accuracy either drops to 0 (removing) or converges to the baseline (keeping), due to the PCs being mostly independent to each other and encoding a similar dimensionality of the full CLIP output.

5.5 Automatic Removal of Spurious Correlations (PCLens vs TextSpan)

In this section, motivated by the findings of Section 5.4.2 on effectively removing PC directions related to a concept c , we investigate whether it is possible to use this localized information in the ViT encoder to increase the overall accuracy on a specific task. In particular, we show how the combination of QuerySystem + PCSelection can be used to automatically remove known spurious correlations in the CLIP ViT encoder without any fine-tuning. We make use of the Waterbirds [65] dataset, where waterbird and landbird foreground images are stitched onto alternating water or land backgrounds. Since ViTs are known to heavily rely on background information, our goal is to remove the information related to these background concepts from the model to improve performance. In TextSpan, the authors manually inspect

5. EXPLAINING THE LATENT SPACE OF CLIP ViT ENCODER $E_{\text{IMAGE}}(I)$ MSA HEADS USING TEXT AND IMAGES

Table 5.1: Comparison of different ViT Models on different methods. Each cell contains both the total accuracy and worst class accuracy across the waterbirds dataset. Cells highlighted in light green indicate that the corresponding method achieves the highest worst class accuracy among all methods for that models, regular green indicates that the corresponding method achieved the highest total class accuracy, while cells highlighted in dark green indicate that the method achieved both the highest worst class accuracy and the highest total accuracy.

Model Name	B-32	B-16	L-14	H-14
Keep Class	64.57, 33.53	79.01, 55.76	74.53, 49.93	77.32, 38.16
Keep Physical	65.45, 35.34	84.66, 54.98	82.22, 70.82	79.67, 37.07
Remove Spurious	66.83, 52.64	83.98, 63.71	88.83, 74.45	84.45, 48.13
TextSpan	64.03, 35.60	82.83, 59.65	81.51, 67.67	80.88, 41.43
Baseline	62.77, 30.02	76.73, 45.63	73.11, 47.67	77.03, 37.22

each head and annotate each head, while, in our approach, we automate this procedure, allowing for finer-grained removal in terms of PCs rather than entire MSA heads.

Setup In our experiment, we perform three different approaches and compare them to TextSpan, aiming to verify which method produces the best results when keeping or removing PCs associated with different concepts. The authors of TextSpan manually characterize each MSA head as encoding a concept and remove those associated with the concepts of “water background” and “land background.” The first method we derive is based on querying the top k PCs associated with the labels “water background” and “land background” using QuerySystem, and removing them from the ViT encoder using our PCSelection (Remove Spurious). We then perform two additional tests: one (Keep Class) in which we query the top k PCs associated with the labels “waterbird” and “landbird” and another (Keep Physical) using labels such as “feet shape” and “beak shape” which describe physical attributes known to differentiate the two classes. For both of these methods, we retain the selected PCs in the CLIP image encoder and ablate all others using PCSelection. We then perform a zero-shot accuracy test on the full Waterbirds test dataset, which comprises 5k images, and measure both the overall accuracy and the worst-class accuracy. The value of k , the number of PCs, is selected via a simple grid search, where we track the value that yields the best worst-class-group accuracy (full results are reported in Appendix A.8).

Results Analyzing the results in Table 5.1, we see that our approach Remove Spurious, which is both automatic and requires no manual intervention,

5.5. Automatic Removal of Spurious Correlations (PCLens vs TextSpan)

largely outperforms the baseline CLIP model as well as the TextSpan, in terms of worst class accuracy and total accuracy. In general, it also performs better than all the other methods for worst class accuracy for all the models. This method can be easily applied to any dataset with known spurious correlations by simply querying the PCs associated with each spurious concept. Empirically, we found that smaller models need more PCs to forget a concept compared to larger ones (i.e. before their worst convergence stabilizes as in Appendix A.8). We hypothesize that the reason may be attributed to the concept being more spread across different directions and heads in smaller models, whereas in larger models the PCs tend to be more specific and localized.

5. EXPLAINING THE LATENT SPACE OF CLIP ViT ENCODER $E_{\text{IMAGE}}(I)$ MSA HEADS USING TEXT AND IMAGES

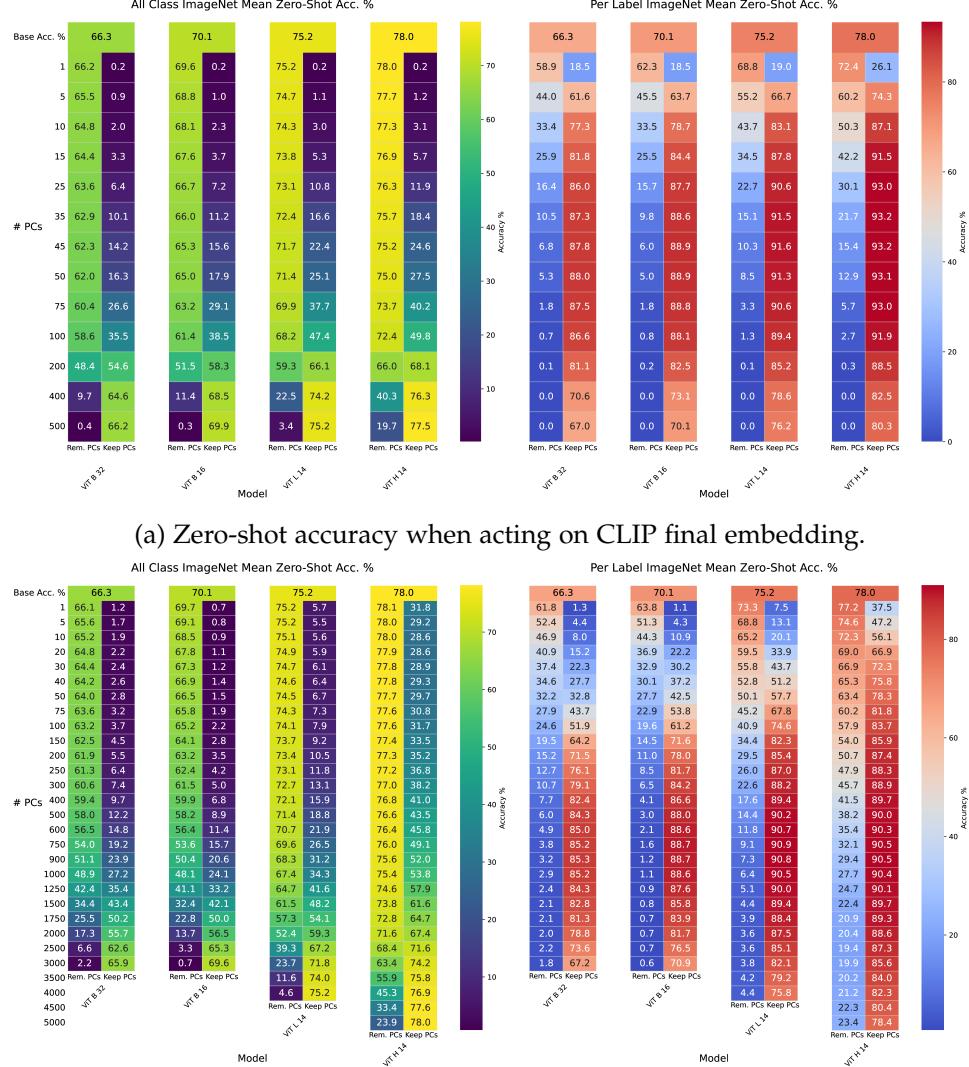


Figure 5.11: Heatmaps of zero-shot accuracy across different ViT models when either removing (left heatmap) or retaining (right heatmap) the contributions of the k most relevant PCs encoding an ImageNet label, as identified by our QuerySystem. Plots at position 5.11a refer to experiments acting on the final CLIP embedding, while plots at 5.11b refer to experiments acting directly on the ViT residual stream. In each heatmap, the left column shows the mean accuracy over the entire ImageNet subset, while the right column shows the mean accuracy on the targeted label.

Chapter 6

Explaining LVLM Image Tokens: Transferring Knowledge from CLIP ViT to LLaVA Image Features $F_{\text{image}}(I)$

In this section, we aim to verify whether the interpretability knowledge acquired in Section 5 on the CLIP ViT encoder can be transferred to LVLM models that utilize the same frozen encoder. Specifically, we focus on the LLaVA-v1.5-7B model [42], which employs the frozen CLIP ViT-L-14-336px encoder to produce image patches and aligns them with a 2-layer MLP in the LLM, Vicuna [54], latent space.

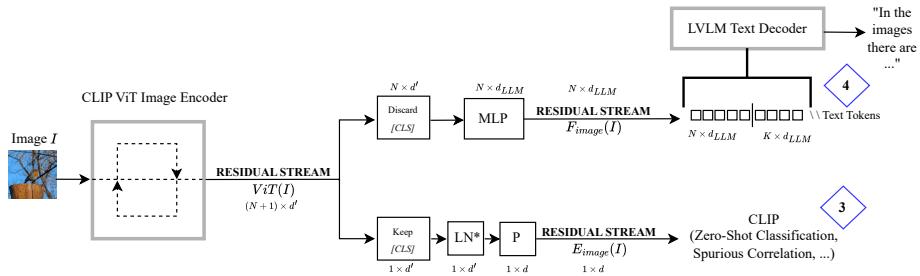


Figure 6.1: This figure highlights the differences between the residual stream of feature images $F_{\text{image}}(I)$ used by LLaVA and the CLIP image encoder’s residual stream $E_{\text{image}}(I)$. Notably, both depend on the hidden residual stream of the CLIP $ViT(I)$, but focus on different tokens: the latter relies on the $[CLS]$ token mapping it in \mathbb{R}^d CLIP embedding space, while the former uses all the other N tokens mapping them on the LLM space $\mathbb{R}^{d_{LLM}}$.

6.1 Overall Setup

This section outlines the models, datasets, and assumptions used throughout the rest of the chapter.

6.1.1 Models and Datasets

We conduct our experiments on the LLaVA-v1.5-7B [42] model, making use of the frozen ViT-L-14-336px encoder. Moreover, we perform our test on the same general image dataset D_I from ImageNet, as in the previous Chapter 5.

Representative Images Throughout the following experiments, we refer to a set of three representative images (hereafter referred to as *Representative Images*) on which qualitative analyses are performed. These images are presented in Figure 6.2 and they have been specifically selected to represent three distinct visual scenarios: an image with a clear, singular subject, in this case the European Robin (Figure 6.2a), an image containing multiple subjects, where ambiguity of classification can arise (Figure 6.2b), and a background-only image of a coloured stone-wall (Figure 6.2c).



(a) A first shot image of an European Robin.
 (b) A pile of a donkey, a dog, a cat and a chicken, from the tale *The Bremen town musicians* [28].
 (c) An image depicting a wall made of colored stones.

Figure 6.2: Three *Representative Images* capturing different visual scenarios: a single-subject image, a multi-subject image, and a background-only image. These images are used in the qualitative experiments of this chapter.

6.1.2 Assumptions

We operate under the following assumptions:

- **Patch tokens are not interchangeable:** When computing the mean ablation value of a patch token at position i from an image $I \in D_I$ at a given layer l , we take the mean over the dataset D_I specifically for that

patch index i . In other words, we do not *assume* that all patch positions i are interchangeable. This distinction is motivated by observed artifacts in the ViT computation: high-norm tokens, often located in low-information background regions, are frequently repurposed for internal computations during inference [10]. Moreover, this choice promotes generality in our analysis.

- **General Image Dataset D_I :** We *assume* that the ImageNet subset used as D_I is sufficiently broad and diverse to cover most of the basic concepts that may be encoded by the model, similarly to the assumption made in Chapter 5.

6.1.3 Key differences between CLIP Residual Stream of CLIP and LLaVA Feature Images

In Figure 6.1, we refer to the residual stream of image features $F_{\text{image}[i]}$ (Equation 3.7) for LLaVA, and the residual stream of the CLIP ViT encoder $E_{\text{image}}(I)$ (Equation 3.5) for CLIP. We first highlight the following two key differences:

- **Different Latent Spaces:** For LLaVA, image features $F_{\text{image}[i]}$ are extracted from the hidden outputs of the ViT encoder at the penultimate layer, and thus lie in $\mathbb{R}^{d'}$. Moreover, in LLaVA, these patch tokens are processed individually via an adapter module, specifically, a two-layer $\text{MLP}(\cdot) : \mathbb{R}^{d'} \rightarrow \mathbb{R}^{d_{\text{LLM}}}$ with GELU activation [31], which projects them into the LLM space $\mathbb{R}^{d_{\text{LLM}}}$ (Equation 3.7). In the original CLIP model, the hidden outputs are first passed through a layer normalization LN , followed by a projection matrix $P \in \mathbb{R}^{d \times d'}$, which compresses them into a lower-dimensional space \mathbb{R}^d (Equation 3.4). This means that it is not possible to trivially use text embeddings to internally characterize individual activation space, since they reside in a latent space with $d \ll d'$.
- **Usage of all patch-tokens besides the [CLS]:** All prior analyses have exclusively focused on the [CLS] token, neglecting the other N patch embeddings, since they are not used for zero-shot classification. In LLaVA, however, the [CLS] token is discarded, and all the other N tokens are used, extracted from the penultimate layer. Hence, it is not obvious whether the previous analysis, such as text-explanations, can be directly applied to these tokens.

6.2 MLP Contributions Dominate the Image Features Residual Stream $F_{\text{image}[i]}(I)$

In this section, we present experiments motivating a surprising finding: the contributions of MLP to the image feature residual stream $F_{\text{image}[i]}$ (Equation 3.7) have the greatest influence on the text output predicted by the LVLM, while MSA layers play a comparatively less significant role. This contrasts with the findings of previous chapters [24] on CLIP models, where the MSA contributions from the final layers were shown to be most relevant to zero-shot classification performance using the residual stream of the [CLS] token (Equation 3.5). This suggests that the two models, or more precisely, the patch tokens in LLava as opposed to the [CLS] token in CLIP, rely differently on the contributions of the ViT encoder.

6.2.1 Interpretation of Mean Ablation (LLava vs CLIP)

Before conducting our experiments, we investigate the semantic meaning of the mean residual stream from our reference ImageNet subset D_I at the output level for both LLava and CLIP. This aims to provide insight into the baseline interpretation of the N tokens as we progressively ablate attention and MLP contributions toward their mean.

Setup The experiments proceed as follows: for LLava, we directly prompt the LVLM with “*Describe in detail the following image.*” and feed it with the mean-ablated [48] tokens across D_I , while for CLIP we simply retrieve the most similar images and texts using the mean-ablation of the [CLS] token.

LLava explanation of the Mean Value

The image features a large, open field with a variety of objects scattered throughout the scene.
There is a mix of textures, including grassy areas, rocks, and possibly some water features.
The overall atmosphere appears to be somewhat barren and arid, with a lack of vegetation.
The image may also contain some structures, such as buildings or other human-made objects, but these are not the main focus of the scene.
Instead, the main focus seems to be on the natural elements present in the image.

6.2. MLP Contributions Dominate the Image Features Residual Stream $F_{image[i]}(I)$

Table 6.1: The descriptive output of LLaVA when asked to describe the mean-ablation of the patch-tokens across D_I . The LVLM output provides a high-level semantic description of a natural scene, highlighting environmental elements like grass and rocks.

Results The corresponding result for LLaVA is reported in Table 6.1, where we are provided with a general description of the mean-ablated hidden tokens, describing a generic scene with many ambiguous and varied elements, highlighting a general meaning. Interestingly, CLIP retrieves one of the most prominent concept from ImageNet with the following top four sentences: "*An image of a small domestic animal,*" "*An image of a domestic pet,*" "*An image of a dog.*" and "*An image of an animal accessory.*" and the top four matching images shown in Figure 6.3. This discrepancy between the two models interpretation of the mean is already noteworthy and highlights a fundamental difference in how the two models interpret mean-ablated representations.



Figure 6.3: Top 4 images from ImageNet with higher cosine similarity with the mean-ablation on D_I of the residual stream $E_{image}(I)$ of the [CLS] token by CLIP.

Vision-Language Expert Classification Prompt

You are a vision-language expert. Analyze the given image and classify it into one of the following categories:
 $\{classes\}$.

Answer with only the most appropriate category in lower case.

Table 6.2: The prompt of LLaVA when asked to perform zero-shot classification on a reference dataset with different classes.

6.2.2 Zero-Shot Accuracy Under Mean/Zero Ablation of MLP and MSA layers (LLaVA and CLIP)

We begin by conducting a benchmark experiment aimed at quantitatively evaluating whether mean ablation of different MSA and MLP layers in the

6. EXPLAINING LVLM IMAGE TOKENS: TRANSFERRING KNOWLEDGE FROM CLIP ViT TO LLaVA IMAGE FEATURES $F_{\text{IMAGE}}(I)$

Table 6.3: Comparative zero-shot classification accuracy of LLaVA and CLIP on ImageNet (left cell value) and CIFAR-10 (right cell value) datasets under progressive mean ablation. Performance is evaluated when MSA, MLP, or both component types are replaced with their mean values up to the indicated layer, highlighting LLaVA’s sensitivity to MLP mean ablation and CLIP’s sensitivity to late-stage MSA mean ablation.

Layer	LLaVA Accuracy			CLIP Accuracy		
	MSA	MLP	MSA+MLP	MSA	MLP	MSA+MLP
0	81.5% 84.0%	81.5% 84.0%	81.5% 84.0%	79.2% 95.1%	79.2% 95.1%	79.2% 95.1%
5	80.0% 82.4%	79.9% 82.9%	81.4% 84.0%	79.1% 94.1%	79.3% 93.6%	79.1% 95.1%
10	68.3% 78.5%	75.4% 84.9%	80.7% 84.0%	79.2% 93.3%	79.4% 93.6%	79.2% 95.2%
15	56.9% 83.6%	58.0% 79.7%	76.5% 82.0%	79.3% 94.1%	79.5% 92.5%	79.3% 95.1%
20	54.4% 81.9%	16.0% 52.4%	14.9% 83.0%	79.3% 88.5%	79.4% 92.8%	79.3% 95.0%
21	53.4% 83.7%	0.2% 28.5%	0.0% 81.0%	79.2% 82.0%	79.4% 93.1%	79.2% 93.0%
22	58.7% 86.4%	0.0% 13.4%	0.0% 26.0%	77.0% 76.7%	79.4% 92.9%	77.0% 86.0%
23	61.1% 88.5%	0.0% 10.2%	0.0% 7.0%	63.7% 57.9%	79.2% 90.1%	63.7% 73.0%
24	60.1% 88.3%	0.0% 7.0%	0.0% 0.0%	23.7% 30.1%	79.1% 90.0%	5.0% 10.0%

residual stream of the ViT encoder, of both the [CLS] token and other image-tokens, yields distinct performance degradation in LLaVA and CLIP under zero-shot accuracy.

Setup The experiment is carried out using the full CIFAR-10 dataset and a subset of 20 randomly sampled, distinct classes from ImageNet D_I . The choice of this dataset size is motivated both by computational constraints, since including all 1,000 class labels in the prompt exceeded the available memory resources, and by the desire to ensure that the LLaVA maintains reasonably high accuracy on the selected subset to be able to monitor meaningful changes in performances. LVLMs generally exhibit noticeably lower zero-shot accuracy compared to their standalone ViT encoder counterparts, with accuracy improving when fewer classes are used [85]. Thus, we select a subset size on which the LVLM performs well enough to yield meaningful results. We use the fixed prompt, shown in Figure 6.2, as previous analyses suggest that classification accuracy is minimally affected by variations in prompt formulation [85]. With the objective of measuring changes in zero-shot classification performance caused by component ablation, without fine-tuning, we perform two types of ablation: one using mean values computed from the general reference dataset D_I (mean ablation), and the other by simply zeroing out the values (zero ablation). We consider three ablation conditions, applied progressively from earlier layers onward: ablation of only attention blocks (MSA), only MLP (MLP), and both (MSA + MLP).

6.2. MLP Contributions Dominate the Image Features Residual Stream $F_{image[i]}(I)$

Table 6.4: Zero-shot accuracy comparison between LLaVA and CLIP on ImageNet (left cell value) and CIFAR-10 (left cell value) datasets with progressive zero ablation. Results show performance degradation as MSA, MLP, or both component types are zeroed out up to the indicated layer, notably highlighting LLaVA’s greater sensitivity to MLP ablation compared to CLIP under this condition.

Layer	LLaVA Accuracy			CLIP Accuracy		
	MSA	MLP	MSA+MLP	MSA	MLP	MSA+MLP
0	81.5% 84.0%	81.5% 84.0%	81.5% 84.0%	79.2% 95.1%	79.2% 95.1%	79.2% 95.1%
1	81.7% 84.4%	82.1% 82.4%	82.1% 82.9%	79.2% 95.1%	79.1% 93.4%	79.0% 93.7%
5	73.2% 80.3%	77.5% 83.7%	80.4% 82.0%	79.2% 93.7%	79.4% 92.0%	79.1% 92.8%
10	49.7% 83.4%	58.6% 81.2%	79.1% 83.5%	79.2% 92.2%	79.4% 92.9%	79.1% 94.0%
15	24.9% 72.3%	24.5% 62.8%	76.6% 83.3%	79.0% 92.3%	79.3% 91.6%	79.1% 94.5%
20	24.9% 66.9%	0.2% 19.7%	53.0% 79.2%	77.9% 86.1%	79.0% 90.8%	79.2% 94.1%
21	23.1% 66.4%	0.0% 6.5%	23.2% 74.2%	74.1% 74.8%	79.1% 91.3%	79.1% 91.2%
22	30.1% 72.7%	0.0% 2.7%	12.8% 46.0%	66.4% 69.5%	79.2% 90.8%	77.8% 89.4%
23	30.4% 72.9%	0.0% 0.0%	0.0% 0.0%	43.8% 39.6%	78.7% 88.5%	66.9% 37.6%
24	30.4% 72.9%	0.0% 0.0%	0.0% 0.0%	4.4% 16.0%	78.6% 92.0%	5.0% 10.0%

Results We report results for both ImageNet and CIFAR-10 within the same table cells, separated by space. The outcomes under mean ablation are presented in Table 6.3, and those under zero ablation in Table 6.4. By analyzing the results, we observe that mean ablation is superior in preserving performance while progressively ablating layers, for both CLIP and LLaVA, especially when ablating MSA components. Interestingly, during these mean ablation studies, LLaVA relies heavily on the contributions of the MLP in the residual streams of its tokens, whereas CLIP exhibits a substantially lower dependency on its MLP, as evidenced by minimal performance degradation. Consistent with the findings in Chapter 5, CLIP relies almost entirely on the MSA contributions from the last four layers for zero-shot accuracy. Conversely, LLaVA appears less dependent on MSA contributions, maintaining relatively high accuracy even when they are ablated. These findings raise the question of whether the ViT processes other tokens differently from the [CLS] token, while also underscoring the value of continuing to use the mean ablation strategy, beyond zero-shot evaluation, to better align with the output distribution.

6.2.3 Evolution of LVLM Text Output under Zero/Mean Ablation (LLaVA)

We complement the quantitative results of the previous section with additional qualitative experiments on mean ablation, to derive further evidences

6. EXPLAINING LVLM IMAGE TOKENS: TRANSFERRING KNOWLEDGE FROM CLIP ViT TO LLaVA IMAGE FEATURES $F_{\text{IMAGE}}(I)$

on the importance of the MLP contributions on LLaVA’s interpretation of the patch-tokens.

Setup We perform two types of mean ablation experiments to analyze their impact on prompt evolution. Specifically, we apply both zero ablation and mean ablation to the full MSA and MLP contributions to the *Reference Images* feature residual stream $F_{\text{image}[i]}(I)$ (Equation 3.7), compare the results with the original output, and examine them qualitatively.

Results The results are reported in Table 6.5, where we observe clear qualitative shifts when applying mean ablations, highlighting the importance of MLP contributions for coherent vision-language alignment. Compared to the original outputs (top row), mean ablations of the MSA frequently produce outputs that remain grammatically and semantically coherent, yet slightly drift from accurately describing the actual scenes. In both MSA ablation cases (MSA Mean Ablation and MSA Zero Ablation), we observe smaller semantic distortions, still generating outputs largely related to the objects in the scene, but missing finer details or mixing up attributes. This suggests that MLP residual connections play a more pivotal role in shaping the descriptive prompt, whereas MSA ablations, though still impactful, have less dramatic consequences. In contrast, ablating the MLP leads to heavily corrupted and nonsensical text (MLP Mean Ablation and MLP Zero Ablation), indicating that completely removing these contributions severely impairs the model’s ability to generate meaningful content. Furthermore, mean ablation consistently yields more intelligible results than zero ablation, reflecting that replacing the residual stream with its mean allows the model to remain more centered on the actual distribution. Overall, our findings brings even stronger evidences that the MLP contribution carries more crucial semantic information for the patch tokens used by LLaVA.

6.2. MLP Contributions Dominate the Image Features Residual Stream $F_{image[i]}(I)$

Ablation Type			
Original	The image features a small bird perched on a wooden post or a tree branch, looking around and observing its surroundings. The bird is positioned towards the center of the scene, with its head facing the viewer. The wooden post or branch appears to be a part of a fence, as there is a fence visible in the background. The scene is set against a blue sky, creating a serene and peaceful atmosphere.	The image features a unique scene where a cat, a dog, and a chicken are all sitting on top of a donkey. The cat is positioned on the left side of the donkey, while the dog is on the right side. The chicken is perched on the donkey's back, close to the center of the scene. The animals seem to be enjoying their time together, creating a whimsical and unconventional moment. The donkey appears to be patiently carrying the three animals on its back.	The image features a colorful wall made of various colored rocks, creating a visually appealing and unique design. The wall is composed of numerous small rocks, each with different colors and sizes, creating a vibrant and diverse appearance. The rocks are arranged in a way that they form a cohesive pattern, making the wall stand out as an artistic and creative display.
MSA Mean Ablation	The image features a bird perched on a wooden post or pole, possibly a telephone pole. The bird is sitting on top of the pole, taking up a significant portion of the scene. The background is a dark blue sky, creating a contrasting backdrop for the bird. The scene appears to be captured in black and white, adding a timeless and classic feel to the image.	The image features a unique scene where a donkey is sitting on top of a horse, creating an interesting and unexpected combination. The horse is positioned in the middle of the scene, while the donkey is placed on its back, both animals seemingly sharing the same space. In the background, there is a chicken standing on a fence, adding a touch of liveliness to the scene. The overall composition of the image is quite intriguing, as it combines ...	The image features a large rock wall with a variety of colors, including shades of blue, purple, and gray. The wall appears to be made of different colored rocks, creating a visually striking and unique pattern. The wall is positioned in front of a building, which adds to the overall composition of the scene.
MSA Zero Ablation	The image features a large, white, and round object, possibly a large pillar or a large piece of art, placed in the center of the scene. The object is surrounded by a wooden platform, giving it a sense of prominence. The platform is positioned on top of a wooden plank, which is in turn placed on a wooden base. The overall composition of the scene creates a sense of balance and harmony.	The image features a large chicken coop with a flock of chickens inside. There are at least 13 chickens visible in the coop, with some of them standing closer to the front and others further back. The chickens are of various sizes, indicating a diverse age range within the flock. The coop is filled with hay, providing a comfortable and safe environment for the chickens.	The image features a large, colorful rock wall with a variety of colors, including shades of blue, white, and gray. The wall appears to be made of different types of rocks, creating a visually striking and unique pattern. The wall is positioned in the center of the image, with a large portion of it visible. The combination of the different colors and the variety of rocks make this wall an interesting and eye-catching feature.
MLP Mean Ablation	The image features a close-up of a person's face with a beard, wearing glasses and a hat. The person appears to be a man with a beard and mustache, possibly a hipster. The man is wearing a hat and glasses, giving him a distinctive look. In the background, there are several other people present, but they are not the main focus of the image. The main subject is the man with the beard and mustache, who stands out as the central figure in the scene.	The image depicts a group of people gathered in a room, with some of them standing close to each other and others scattered throughout the space. There are at least 13 people in the scene, with some of them wearing ties, indicating a semi-formal or formal gathering. The room appears to be a living space, with a couch situated in the background and a dining table nearby. A few chairs are also present in the room, with one near the couch and ...	The image features a group of people gathered together, with some of them wearing headphones. There are at least nine people in the scene, with some standing closer to the foreground and others further back. They appear to be engaged in a conversation or enjoying a shared activity. In the background, there is a TV mounted on the wall, which might be the source of the shared activity or entertainment. A pair of scissors can be seen placed on ...
MLP Mean Ablation	The image features a close-up of a person's face with a beard, wearing glasses and a hat. The person appears to be a man with a beard and mustache, possibly a hipster. The man is wearing a hat and glasses, giving him a distinctive look. In the background, there are several other people present, but they are not the main focus of the image. The main subject is the man with the beard and mustache, who stands out as the central figure in the scene.	The image depicts a group of people gathered in a room, with some of them standing close to each other and others scattered throughout the space. There are at least 13 people in the scene, with some of them wearing ties, indicating a semi-formal or formal gathering. The room appears to be a living space, with a couch situated in the background and a dining table nearby. A few chairs are also present in the room, with one near the couch and ...	The image features a group of people gathered together, with some of them wearing headphones. There are at least nine people in the scene, with some standing closer to the foreground and others further back. They appear to be engaged in a conversation or enjoying a shared activity. In the background, there is a TV mounted on the wall, which might be the source of the shared activity or entertainment. A pair of scissors can be seen placed on ...
MLP Zero Ablation	The man is a myth in the woman's ass.	I'm gonna)	What is the most important?

Table 6.5: Evolution of LLaVA outputs under mean and zero ablation of MSA and MLP layers for the three *Reference Images*. Notable is the difference in impact between MSA and MLP layer ablations, with the former having a less disruptive effect compared to the latter.

6.3 Do MSA and MLP Layer Contributions Attend to the [CLS] Residual Stream Differently?

In this section, we aim to investigate whether the differences in the impact of mean-ablating the MSA and MLP contributions between LLaVA and CLIP arise from variations in how the residual streams of different tokens are formed inside the ViT CLIP encoder. In particular, we compare the composition of the residual stream for the [CLS] token, used for zero-shot classification in CLIP, to that of other tokens employed by LLaVA.

Conjecture If we observe that the magnitude of the MSA contributions in the last layers either dominates or aligns strongly with the residual stream of the [CLS] token, while this is not the case for the residual streams of other tokens, this would provide evidence for why the [CLS] token is particularly susceptible to MSA ablations, whereas the other tokens are not but more susceptible to MLP ones. We hypothesize that most of the [CLS] token information is captured through the global pooling inherent in the MSA contributions, which explains the reduced performance when it is mean-ablated for zero-shot classification in CLIP.

Setup We conduct two distinct experiments aimed at determining whether the differences between the [CLS] token and other tokens are reflected in how the MSA and MLP layers contribute to their respective residual streams. Specifically, we examine how signal strength, norm ratios, and directional alignment, measured by cosine similarity, between each MLP and MSA layer’s output and its residual stream evolve, using the tokens averaged from the reference dataset D_I . Moreover, we also make an analogous comparison between the [CLS] token’s MLP and MSA contributions, from the same D_I , with those of the other tokens to quantify any difference in magnitude or alignment across these components.

Results We report the norm experiment in Figure 6.4a and the cosinect-similarity experiment in Figure 6.4b. Overall, we observe consistent, layer-dependent differences in how the model processes the [CLS] token versus the other tokens, in both the contributions from the MSA and MLP components. When we examine the MSA norm as a ratio of each token’s residual-stream norm, the [CLS] token consistently shows larger magnitude contributions than the other tokens, indicating a greater relative change. This

6.3. Do MSA and MLP Layer Contributions Attend to the [CLS] Residual Stream Differently?

gap, modest in the early layers, becomes more marked in the deeper layers, particularly from layers 20 to 23, implying that the [CLS] token integrates increasingly global information through MSA components, peaking in the final layers. A similar pattern appears for the MLP plot: the [CLS] token again exhibits larger magnitude changes, most noticeably in the first layer and, even more strongly, between layers 20 and 24. There's a spike at layer 0 that can be explained by the fact that the layer 0 [CLS] embedding is learned and not yet image-dependent. The other tokens follow a comparable trajectory, with a pronounced spike at layer 21 that may mark a critical point in representation refinement. Comparing the MSA norm of all tokens to that of the [CLS] token, we find that the absolute magnitude for the other tokens consistently, though slightly, exceeds that of the [CLS] token across all layers. This suggests that, although the [CLS] token is heavily influenced by MSA, the cumulative MSA contribution induces larger absolute changes in the remaining tokens. The same holds for the MLP norm: the combined magnitude of the other tokens again surpasses that of the [CLS] token, especially at layer 21, where the spike is almost eight times larger, pointing to intensive representation refinement. Turning to cosine similarity, the MSA direction of the [CLS] token remains consistently positive and higher than that of the other tokens, indicating steady alignment with its residual stream. By contrast, the other tokens display fluctuating alignments, dipping sharply into negative values around layer 21, which suggests divergence as the model gathers and refines information. The MLP cosine similarities are generally lower and more volatile: the [CLS] token does not consistently outperform the other tokens in terms of alignment, and directional coherence emerges only in the last layers. The other tokens show a sharp positive alignment spike at layer 21, reinforcing the idea of intensive refinement at this depth. Finally, the cosine similarity between the MSA outputs of all tokens and that of the [CLS] token remains high throughout the network, though it declines slightly in the deepest layers, hinting at a gradual loss of directional coherence as global information is pooled. For the MLP outputs, this similarity is minimal at the start, rises sharply at layer 2, then becomes unstable and drops between layers 18 and 22, with another peak at layer 21, indicating growing directional divergence between the [CLS] token and the rest. Lastly, at layer 21, the MLP contributions of all tokens appear to dominate the residual stream, with a high norm ratio of 0.9. These contributions are also highly cosine-aligned with the final residual stream, suggesting that this layer plays a crucial role in the individual refinement of token representations. Following this, at MSA layer 22, at the next ViT block, we observe a substantial increase in the magnitude of the MSA contribution for the [CLS] token. This indicates a strong global pooling of information from all other tokens and helps explaining why the [CLS] token is particularly susceptible to mean ablation of the last MSA layers: its residual stream is largely dominated by MSA contributions that integrate refinement information from the rest of the tokens.

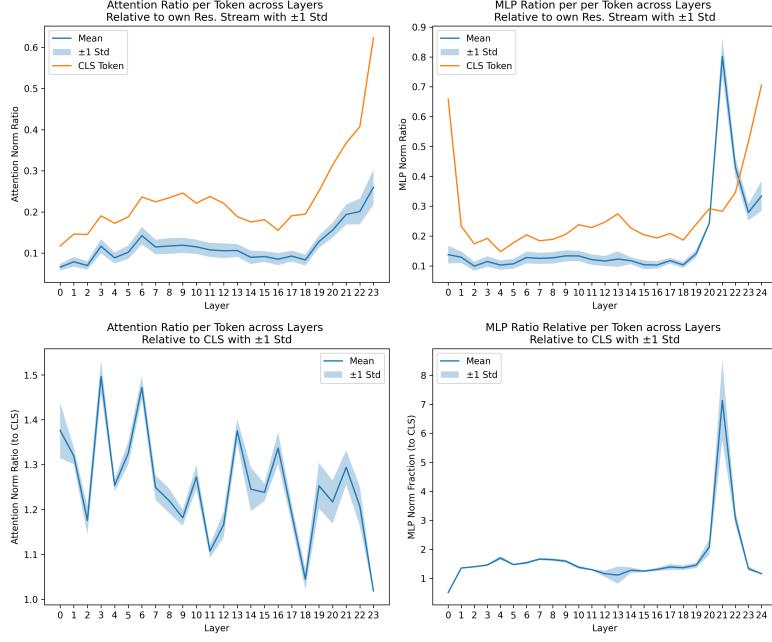
6. EXPLAINING LVLM IMAGE TOKENS: TRANSFERRING KNOWLEDGE FROM CLIP ViT TO LLaVA IMAGE FEATURES $F_{\text{IMAGE}}(I)$

Collectively, these findings highlight structural differences in how the MSA and MLP layers interact with the residual stream across tokens. They underscore the central role of the [CLS] token in progressively encoding global information, particularly in the final layers. However, they do not exhaustively explain why LLaVA patch-tokens are more susceptible to MLP perturbations compared to CLIP [CLS], a question that must be explored further.

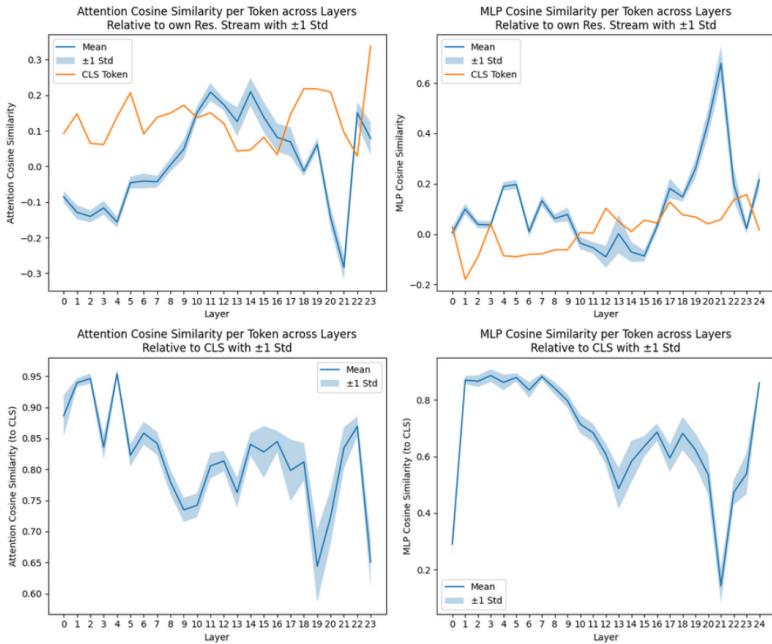
6.4 Concluding Remarks on LLaVA Analysis

Our analysis of the LLaVA model using the frozen CLIP ViT encoder reveals intriguing differences compared to standard CLIP [CLS] token processing. We find a marked reliance on MLP contributions for the patch tokens $F_{\text{image}[i]}(I)$, in contrast to the dominance of MSA contributions observed for the $E_{\text{image}}(I)$ embedding derived from the [CLS] token. While we observe structural differences in how the residual stream forms for the [CLS] token versus other tokens, pinpointing the definitive cause for LLaVA’s distinct behavior remains challenging. We *hypothesize* that the LLaVA adapter, $\text{MLP}(\cdot)$, which connects vision tokens to the LLM’s latent space, plays a significant role, potentially filtering out global MSA contributions while preserving localized MLP information. This hypothesis is partly supported by preliminary findings where amplifying MSA contributions minimally affected LLaVA’s output stability. Understanding this potential filtering mechanism, particularly the role of the adapter $\text{MLP}(\cdot)$, is crucial for interpreting how visual information is translated for language models in architectures like LLaVA and remains an open question requiring further investigation beyond the scope of this thesis. Consequently, we conclude our exploration here, acknowledging that transferring interpretability insights directly from CLIP’s ViT encoder to LVLMs like LLaVA is non-trivial. Potential hypotheses and directions for further research, including a more thorough investigation of the adapter’s filtering role, are discussed in the Discussion and Future Work Chapter 7.

6.4. Concluding Remarks on LLaVA Analysis



(a) Ratios of the MLP (left) and MSA (right) L2-norm contributions across layers, relative to the entire residual stream. In the top two plots, the orange line represents the [CLS] token, whereas the blue line shows the mean ratio for the remaining N tokens, with the standard deviation indicated by the shaded band. The bottom plot presents, layer by layer, the ratio of this N -token mean to the [CLS] token.



(b) Cosine similarities of the MLP (left) and MSA (right) contributions across layers, measured relative to the entire residual stream. In the top two plots, the orange line corresponds to the [CLS] token, whereas the blue line shows the mean similarity for the remaining N tokens, with the shaded band indicating one standard deviation. The bottom plot presents, layer by layer, the similarity between the mean of these N tokens and the [CLS] token.

Figure 6.4: Combined figure showing the ratios of norms per image-token (top) and cosine similarities (bottom) of each MLP and MSA layer's contributions to the hidden residual stream.

Chapter 7

Discussion and Future Work

This chapter discusses our findings, highlighting the methodological choices used throughout our experiments, together with the assumptions, and proposes further ideas for future work.

7.1 Dataset Considerations

Scope and Characteristics of Selected Datasets A core aspect shaping this work is the choice of datasets. The primary image dataset D_I is derived from ImageNet [12], and the text dataset D_T is constructed based on its classes and common English words, aiming for broad but feasible coverage. While selected for their diversity and common use as benchmarks, assuming they offer general applicability, it is acknowledged that all datasets possess inherent characteristics and may exhibit certain biases. For example, the prominence of the *dog* concept emerges during the CLIP mean-ablation analysis (Section 6.2.1). Consequently, the principal components p_k identified by PCLens and their associated semantic characterizations are naturally influenced by the specific data distributions within D_I and D_T . Interpretations and the specific directions identified could potentially differ if alternative datasets with different distributional properties were used. Furthermore, quantitative evaluations were primarily performed on ImageNet, CIFAR-10 [39], and Waterbirds [65]. Although standard benchmarks, their specific characteristics mean they may not represent the full spectrum of real-world visual scenarios or all types of spurious correlations one might encounter.

Future Work: Exploring Dataset Diversity Future research could productively extend the analysis by employing more varied datasets beyond the ImageNet subsets and standard benchmarks used here. Utilizing datasets with distinct visual characteristics (e.g., medical imaging, satellite data) or from specific domains could reveal how semantic concepts are encoded differently or expose dataset-specific structures learned by the model, such as

the observed *woman-man* PC. This would provide a richer understanding of the generalizability of the semantic directions identified by the PCs p_k . Evaluating the proposed intervention methods and overall model performance on a broader range of downstream tasks and datasets exhibiting different statistical properties is also crucial for assessing the robustness and wider applicability of the findings.

7.2 ViT Interpretation and Intervention Methodology

Methodological Choices and Assumptions Our primary method, PCLens, leverages PCA to decompose the activation space $A^{l,h}$ of MSA heads. PCA is chosen for its ability to identify dominant linear structures within the data. In the context of this work, it proves effective in addressing polysemy, as supported by both quantitative and qualitative results. As a linear technique, however, PCA primarily captures linear relationships and might not fully represent more complex, non-linear structures if they exist within the activation space’s true semantic manifold. The characterization of components p_k and the ranking mechanism in QuerySystem also rely significantly on cosine similarity. This metric is selected and motivated due to its direct relevance to CLIP’s training objective and its ability to capture angular alignment, as motivated in Section 5.2.1. While effective for the analyses performed, focusing on angular alignment might overlook other geometric or semantic nuances in the high-dimensional space \mathbb{R}^d . Exploring alternative similarity or distance metrics thus remains a potential avenue for future investigation. Furthermore, the analysis of the residual stream $E_{\text{image}}(I)$ (Equation 3.5) followed the precedent set by [24] in approximating MLP contributions via mean ablation ($\bar{M}_{[\text{CLS}]}^l$). This methodological choice, empirically supported for CLIP’s zero-shot tasks, allows for a focused analysis on the MSA heads’ contributions. It is acknowledged that this simplification might place less emphasis on subtler MLP roles, a point potentially underscored by their apparent importance in the LLaVA context. Similarly, the decision to focus the analysis on the last four MSA layers was based on prior findings indicating their critical role in CLIP’s zero-shot accuracy [24]. Contributions from earlier layers, while deemed less critical for the specific tasks examined, represent another area that could be explored in future work.

Intervention Methodology and Scope The proposed PCSelection method demonstrates positive results for tasks like removing spurious correlations (Section 5.5). Its current implementation involves selecting the number k of principal components via grid search, optimizing for worst-group accuracy on the target dataset. This process is inherently tied to the specific dataset and evaluation metric chosen. Future work could focus on developing more principled or automated methods for determining an optimal or effective

k based on different criteria beside optimization on the problem-metric. Investigating how the required number or the nature of the most relevant PCs varies across different ViT models or target concepts would also yield valuable insights. Additionally, refining the evaluation methodology beyond zero-shot accuracy could allow for a more nuanced quantification of the degree of concept amplification or suppression achieved by interventions acting on the modified embedding $\tilde{E}_{\text{image}}(I)$.

Model Architecture Generalizability The experiments in this work are conducted on four specific CLIP ViT architectures (ViT-B-32, ViT-B-16, ViT-L-14, ViT-H-14). While trends across model sizes are noted (e.g., regarding concept specialization and redundancy), the extent to which these specific findings generalize to other ViT variants or different families of contrastive models beyond CLIP warrants further study. A more in-depth comparative analysis across a wider range of model sizes and potentially different contrastive training frameworks could provide deeper insights into how model scale impacts internal representational structures and the emergence of specialized components like key PCs p_k in different layers or heads, thereby clarifying the relationship between model capacity and representational organization.

7.3 LLaVA: Bridging ViT and Language Models

Challenges in Transferring Interpretability from CLIP to LLaVA A key finding emerged regarding the transferability of insights when extending the analysis to LLaVA (i.e. LLaVA-v1.5-7B). Interpretations derived from analyzing CLIP’s global [CLS] token representation $E_{\text{image}}(I)$, which appears heavily reliant on MSA contributions for zero-shot tasks, did not directly translate to explaining LLaVA’s processing of individual patch tokens $E_{\text{image}[i]}(I)$. Within the LLaVA framework, MLP contributions seemed more influential for these tokens in shaping the final textual output, whereas MSA contributions, even when experimentally amplified, had a less discernible impact. The precise role of the LLaVA adapter $\text{MLP}(\cdot) : \mathbb{R}^{d'} \rightarrow \mathbb{R}^{d_{\text{LLM}}}$ in mediating this observed difference, potentially by filtering or transforming ViT features in specific ways, remains an important open question. This highlights the challenges in directly applying interpretability methods developed for the CLIP encoder to downstream LVLM architectures and suggests that findings may not automatically generalize across different model types or tasks, even when they share components like the ViT encoder.

Future Work: Ideas for Explaining the ViT-LLM Interaction in LLaVA Understanding precisely how LLaVA utilizes features from the shared ViT encoder remains a crucial area for future research. Several avenues warrant exploration to elucidate this interaction. A primary focus could be on

7. DISCUSSION AND FUTURE WORK

investigating the adapter’s role. A plausible hypothesis is that the LLaVA adapter learns to prioritize localized, token-like information, from the MLP components, $M_{[i]}^l$, possibly because this aligns better with the expected input format of the LLM’s tokenizer, while potentially filtering out more global, attention-derived information originating from the MSA components, $H_{[i]}^l$. Rigorously testing this hypothesis is essential and could involve analyzing the adapter’s learned weights, probing the information content of the final features $F_{\text{image}[i]}(I)$ compared to the pre-adapter ViT output $\text{ViT}(I)_{[i]}^{L-1}$, or examining why amplified MSA signals, $H_{[i]}^{l,h}$, appear attenuated later in the LLaVA pipeline. Further research could delve into analyzing the separability of components before they are processed by the adapter. Applying dimensionality reduction techniques like PCA or clustering methods to the pre-adapter outputs of the MLP $M_{[i]}^l$ and MSA $H_{[i]}^l$ components in the $\mathbb{R}^{d'}$ space might reveal their structural distinctness. Quantitative methods, such as supervised classification or distance metrics, could further clarify how separable these representations are, potentially shedding light on the adapter’s filtering behaviour. Another direction involves designing experiments aimed at isolating component contributions more directly. Focusing on token-wise prediction tasks, like predicting attributes from individual $F_{\text{image}[i]}(I)$ features under different ablation conditions (e.g., ablating all MLPs versus all MSAs), could help quantify the specific information carried by each component type for the non-[CLS] tokens utilized in LLaVA. Finally, examining token transformations induced by the adapter could provide valuable insights. Analyzing the self-similarity of individual patch tokens $I_{[i]}$ before and after processing by the adapter $\text{MLP}(\cdot)$, particularly when comparing scenarios with and without MSA ablations, might clarify how the adapter modifies token identity and whether it selectively preserves or discards information related to attention mechanism modifications.

Addressing these points can be crucial for interpreting how visual information is translated and utilized within LVLMs like LLaVA, representing a significant step towards understanding their complex multi-modal latent space structures.

Chapter 8

Conclusion

This final chapter consolidates the key findings and contributions presented throughout this work. We revisit the primary research questions, summarize the methodologies developed, and highlight the main outcomes of our investigation into the spectral explainability and concept-level interventions within the CLIP ViT encoder, as well as the implications for downstream on LLaVA.

8.1 Conclusions

In this work we addressed the challenge of interpreting the internal mechanisms of CLIP’s ViT image encoder $E_{\text{image}}(I)$, particularly the role of its MSA heads in composing the residual stream, and explicitly elucidating the role of polysemantic heads. Leveraging the shared image-text embedding space \mathbb{R}^d , we introduced PCLens, a spectral and mechanistic explainability framework. PCLens decomposes the activation space $A^{l,h}$ of individual MSA heads into principal components p_k , which are then characterized bidirectionally using both text embeddings E_{D_T} and image embeddings E_{D_I} .

Our analysis, conducted across multiple ViT model sizes, revealed that this finer-grained decomposition uncovers highly specialized, semantically meaningful directions p_k within the ViT residual stream $\text{ViT}(I)$. These directions often correspond to intuitive concepts (e.g. *color*, *objects*) as well as more abstract ones (e.g. *pair*, *nurturing*), demonstrating its capacity to tackle the ambiguity of polysemantic heads where methods like TextSpan fails. Furthermore, we provided evidence that these semantic directions p_k maintain coherence between the local latent activation space of MSA heads and the final CLIP embedding latent space \mathbb{R}^d . Quantitatively, PCLens achieves superior reconstruction of zero-shot classification accuracy compared to TextSpan using fewer textual descriptors per head.

Building upon this decomposition, we developed QuerySystem, an automated procedure to locate the principal components p_k across multiple MSA

8. CONCLUSION

heads that are most strongly associated with an arbitrary input concept c , represented by its embedding e_c . This tool highlights the distributed nature of concept encoding, revealing both redundancy and specialization within the ViT architecture. This can also help to find and elucidate semantic bias, hidden in problematic directions in the latent space, such as the controversial directions of opposite concepts (e.g. *woman-man*). Complementing this, PCSelection enables targeted, concept-level interventions directly on the ViT’s residual stream. By selecting a set S of relevant PCs identified by QuerySystem, PCSelection can retain, remove, or amplify specific concept contributions within the final reconstructed image embedding residual stream $\tilde{E}_{\text{image}}(I)$ without any model fine-tuning. The efficacy of this approach is demonstrated through improved worst-group accuracy on the Waterbirds benchmark by automatically identifying and removing PCs associated with spurious background correlations.

Finally, we extend our investigation to LVLMs, specifically LLaVA, which utilizes a frozen CLIP ViT encoder. We observe that transferring interpretability insights between the CLIP ViT encoder and downstream LLaVA is not as straightforward as it might seem. We uncover a notable asymmetry in component contributions: while the [CLS] token representation $E_{\text{image}}(I)$ in standard CLIP is dominated by MSA contributions when evaluated on zero-shot accuracy benchmarks, the N image patch features $F_{\text{image}[i]}(I)$ used by LLaVA appear to be more heavily influenced by MLP contributions when generating textual outputs. We provide evidence of this behaviour both qualitatively and quantitatively, when facing the problem of performing zero-shot class accuracytokentokentokentoken. This suggests structural differences in how information is processed between the N tokens and the [CLS] token, which we analyze in terms of norm ratio and cosine similarity. Finally, we outline future work directions to elucidate this difference, hypothesizing that the main issue lies in the way the adapter $\text{MLP}(\cdot)$, which connects the CLIP vision encoder image embeddings to the LLM text embeddings, processes each individual patch token.

Appendix A

Appendix

A.1 New General Text Dataset D_I

GPT CLIP Model Descriptions Prompt

Generate 5 very simple descriptions for the class '{class_name}', tailored for a CLIP model. Each description should aim to capture the concept as visually as possible. If the concept can be divided into meaningful subparts, provide one description for each subpart to maximize coverage. The final description should clearly summarize the concept in a short and simple way.

For example:

For "shark":
An image of a large fish.
An image of a gray animal.
An image of a shark swimming.
An image of a predator underwater.
An image of a shark.

For "banana":
An image of a yellow fruit.
An image of a curved object.
An image of a banana on a table.
An image of a peeled banana.
An image of a banana.

For "seasons":
An image of green trees in summer.
An image of red and orange leaves in autumn.
An image of snow on the ground in winter.
An image of flowers blooming in spring.
An image of the four seasons.

For "freedom":
An image of a bird flying in the sky.
An image of a person with open arms.
An image of an empty road.
An image of a waving flag.
An image of freedom.

Descriptions should prioritize clear, visual traits or representations. When dealing with abstract concepts, break them into smaller, clear components and provide visual or textual representations. Output only the descriptions as plain text, separated by line breaks (\n).

In this section, we explain how we generate the new text dataset D_T used in our work. The aim is to make it as general as possible, similar to how we consider ImageNet to be broad. To achieve this, we query GPT-4o-mini [21] with the prompt in Box A.1 for each one of the following concepts:

A. APPENDIX

- Each label of ImageNet (1,000 labels)

- The top 2,000 most frequently used words in the English dictionary.

Our goal is to obtain four different visual descriptions and one plain description for each of these concepts, ensuring that they capture diverse aspects of the world in various ways. The choice of redundancy comes from our idea of providing possible synonym descriptions, allowing for a conceptual characterization of the principal components. We remove duplicates in the resulting dictionary, which ultimately comprises 14861 unique sentences that we add to D_I .

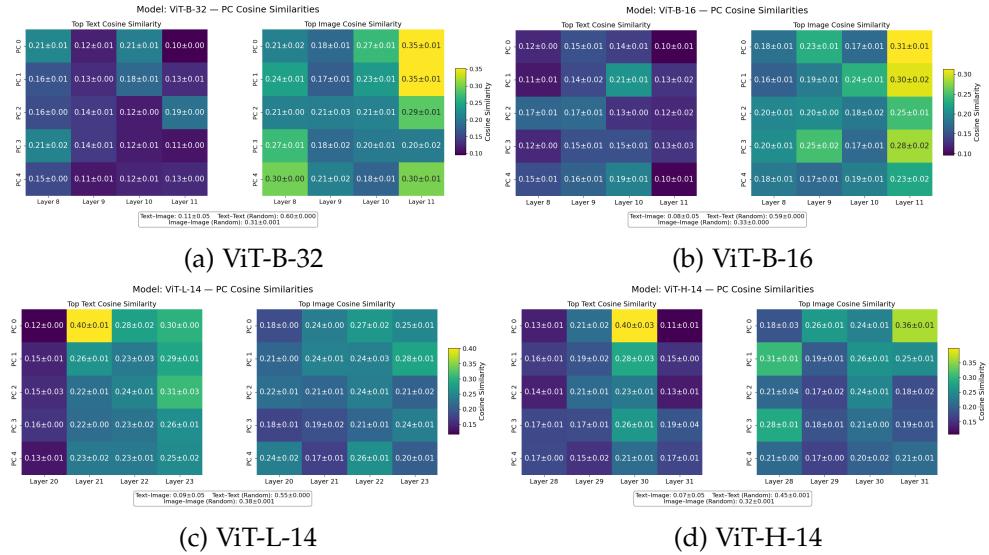


Figure A.1: Cosine similarities of the top 5 PCs from ViT-B-32 (top left), ViT-B-16 (top right), ViT-H-14 (bottom left), and ViT-L-14 (bottom right) across different layers, computed separately for text and image embeddings. Each cell reports the mean cosine similarity along with the standard deviation across samples. The plot covers the last four relevant MSA layers, highlighting multi-modal embedding alignment variations at these deeper layers of each ViT architecture. Reference baseline cosine similarities for final CLIP embedding pairings are provided below for comparison: text-image similarity, random text-text similarity, and random image-image similarity.

A.2 Cosine Similarity of PCs with CLIP Image-Text Embeddings across different ViT Encoders

In this section, we provide additional plots (Figure A.1) showing the cosine similarity of the top five PCs with CLIP image-text embeddings from the last

A.3. Non-Polysemantic Colour Heads across ViT Encoders (PCLens vs TextSpan)

four layers, across different ViT encoders. Each plot is complemented with the global random text-text, text-image, and random image-image cosine similarity scores computed over the embeddings of D_I and D_T of the CLIP model corresponding to that ViT encoder.

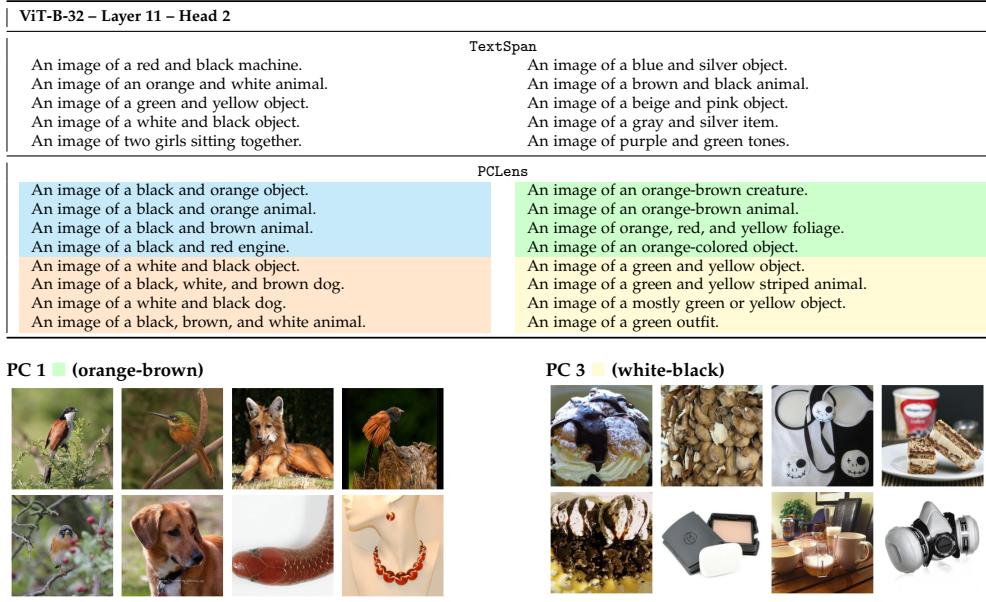


Figure A.2: Comparison of TextSpan with PCLens for ViT-B-32 Layer 11 Head 2 on a *colour* head. For PCLens, each color highlights sentences corresponding to a different principal component, i.e. PC 0 , PC 1 , PC 2 , PC 3 , with displayed image-descriptions for PC 0  and PC 3 .

A.3 Non-Polysemantic Colour Heads across ViT Encoders (PCLens vs TextSpan)

In this section, we provide examples of non-polysemantic *colour* MSA heads, i.e. those associated with a single concept. While this scenario is particularly favorable for TextSpan, we want to emphasize the strengths of our method, PCLens, even under these ideal conditions for the baseline. Specifically, we observe how the image and text characterizations of the PCs align and complement each other, enhancing the interpretability of the head and providing a finer explanation; single PC is more precise and clearer, as it is associated with a single color rather than multiple. Surprisingly, the *colour* head appears across all ViT models. We present the results for ViT-B-32 in Figure A.2, ViT-B-16 in Figure A.3, ViT-L-14 in Figure 2.2, and ViT-H-14 in Figure A.4. Notably, across models, individual MSA heads specialize in different shades

A. APPENDIX

ViT-B-16 – Layer 11 – Head 10	
	TextSpan
An image of an orange and white animal. An image of a red and black creature. An image of the color green. An image of a black and yellow animal. An image of a white and gray animal.	An image of a brown and tan animal. An image of the color blue. An image of a beige and pink object. An image of a red and yellow machine. An image of purple and green tones.
	PCLens
An image of a black and orange animal. An image of a black and brown animal. An image of a black and red engine. An image of a brown and black animal. An image of a green and orange object. An image of a bright green animal. An image of red, yellow, and green lights. An image of a green outfit.	An image of a bird with blue and green hues. An image of a bird with black and blue feathers. An image of a gray and black animal. An image of a black animal. An image of a color between blue and red. An image of a red and white vehicle. An image of a dog with a white and red coat. An image of a black and red engine.
PC 1  (black-blue)	PC 4  (red-pattern)
   	   
   	   

Figure A.3: Comparison of TextSpan with PCLens for ViT-B-16 Layer 11 Head 10 on a *colour* head. For PCLens, each color highlights sentences corresponding to a different principal component, i.e. PC 0 , PC 1 , PC 2 , PC 3 , with displayed image-descriptions for PC 1  and PC 4 .

of colours and multiple colours heads are present across the same ViT model.

A.4 Polysemantic MSA Heads across ViT Encoders (PCLens vs TextSpan)

In this section, we provide examples of various polysemantic MSA heads, where it becomes evident that using only TextSpan does not yield clear results on their role. Our method, PCLens, effectively disentangles these representations. We observe that the image and text characterizations of the principal components align and complement each other, thereby enhancing the interpretability of the MSA heads. We present the results for ViT-B-32 in Figure A.5, ViT-B-16 in Figure A.6, ViT-L-14 in Figure 2.2, and ViT-H-14 in Figure A.7.

A.5 PCs Directions across ViT Encoders (QuerySystem)

In this section, we provide additional examples of characterization of principal components across different ViT encoders, highlighting the strength of our method, QuerySystem, in identifying a given concept c . For each model, we present the first PC found by our method related to a given concept c ,

A.6. Comparison of Concept Reconstruction (*Party*) across ViT Encoders (QuerySystem and PCSelection)

ViT-H-14 – Layer 29 – Head 2	TextSpan	PCLens
An image of a gray and silver item. An image of a black automobile. An image of the color brown. An image of a dark green animal. An image of a dark blue bird.	An image of a dark purple object. An image of a golden object. An image of a cat with a mixture of orange and black fur. An image of the word ‘theory’ written on a piece of paper. An image of a brownish-gray furred animal.	
An image of a gray and black animal. An image of a black and gray object. An image of a black and silver dog. An image of a black and silver device. An image of the color brown. An image of a brown food. An image of a brown creature. An image of a brown and tan animal.	An image of a gray and silver item. An image of a silver-gray creature. An image of a gray mechanism. An image of a silver-grey object. An image of a black and gold object. An image of a golden-colored object. An image of a golden object. An image of a golden-colored canine.	
PC 0 ■ (black-gray)	PC 3 □ (gold-on-black)	

Figure A.4: A comparison of TextSpan with PCLens for ViT-H-14 Layer 29 Head 2 on a *colour* head. For PCLens, each color highlights sentences corresponding to a different principal component, i.e. PC 0 ■, PC 1 ■■, PC 2 ■■■, PC 3 □, with displayed image-descriptions for PC 0 ■ and PC 3 □.

showing the top four text descriptions and top eight images associated with each PC direction, as determined by PCLens. In Figure A.8, we illustrate an example where different ViT models exhibit, in different MSA heads, similar semantic direction in the MSA latent space, encoding the concepts of *black* and *white*, which aligns with human intuition of opposites. Conversely, in Figure A.9, we provide an example for the concept of *happiness*, where the opposite PC directions encode diverging concepts across ViT models, deviating from the human intuition of opposites. Notably, no PC across the various ViTs presents a distinct direction for the concept of *happiness*, but rather for a subset of this concept (e.g. smiling, party, etc.).

A.6 Comparison of Concept Reconstruction (Party) across ViT Encoders (QuerySystem and PCSelection)

In this section, we present an additional qualitative comparison of the reconstructions of the concept of *party* (Figure A.10) across the four ViT model sizes, using QuerySystem and PCSelection. The behaviour for the concept *party* differs: each model appears to encode distinct fine-grained aspects of a party. Specifically, ViT-B-32 retrieves “drinking” scenes with fewer

A. APPENDIX

ViT-B-32 – Layer 11 – Head 11	
	TextSpan
An image of a Saharan horned viper. An image of a pan flute. An image of a spaghetti squash. An image of a person skillfully juggling balls. An image of a spider in a circular web.	An image of a trilobite. An image of weeds growing through cracks in old stone. An image of a grid of rebar in the ground. An image of a folded origami crane. An image of braille on a book page.
	PCLens
An image of an eastern diamondback rattlesnake. An image of a Saharan horned viper. An image of an alligator lizard. An image of a snake with scales. An image of a twisted bread product. An image of a twisted tree trunk or branch. An image of a bread with flour dusting on top. An image of a shoreline with waves.	An image of a brain coral. An image of a potato with rough skin. An image of a large boulder. An image of a ball-shaped dough. An image of strings and hammers inside a piano. An image of tension wires on a bridge. An image of a handle with a bundle of strings. An image of a grid of rebar in the ground.
PC 1  (rough-skin)	PC 3  (string-pattern)
	

Figure A.5: A comparison of TextSpan with PCLens for ViT-B-32 Layer 11 Head 11 . For PCLens, each color highlights sentences corresponding to a different principal component, i.e. PC 0 , PC 1 , PC 2 , PC 3 , with displayed image-descriptions for PC 1  and PC 3 .

PCs, whereas ViT-B-16 emphasises “group” scenes. Remarkably, the larger ViT-H-14 produces reconstructions containing more dynamic elements that match the overall notion of a party, suggesting that it likely contains more specialized components for this concept. Interestingly, only a small portion of the components is active during the generation of the reconstructed residual stream, showing that fewer directions are already able to encode a complex concept such as *party*.

A.7 PCs encode Semantic Meaning for both ViT MSA Heads and CLIP Latent Space (QuerySystem and PCSelection)

In this section, following the experiment and results of Section 5.4.2, we provide additional line-plots showing the accuracy of the different ViT models vs the increasing number of PCs used, where the fading shape represents one standard deviation of the data. The plot related to the final CLIP embedding is provided at Figure A.11, while the one for the internal MSA heads of the ViT at Figure A.12.

A.8. Optimal Number of PCs across the Methods for Removing Spurious Correlation (QuerySystem and PCSelection)

ViT-B-16 – Layer 11 – Head 9	
An image of velvet fabric. An image of leather. An image of a knitted object. An image of a metal and wood object. An image of a group of five people.	TextSpan An image of you. An image of a sandwich on a plate with lettuce and tomato. An image of a mesh-covered top. An image of the number two. An image of the four seasons.
An image of a knitted object. An image of a knitted textile. An image of a knitted garment. An image of a knitted fabric. An image of leather. An image of a hardened piece of leather. An image of a leather encasement. An image of a leather object.	PCLens An image of a four-stringed object. An image of a flat, two-dimensional shape with four sides. An image of the four seasons. An image of a hat made from felt. An image showing 60 small divisions on a clock. An image of a group of five people. An image of an object with four right angles. An image of a six-stringed object.
PC 0  (knitted)	PC 2  (leather)
       	       

Figure A.6: A comparison of TextSpan with PCLens for ViT-B-16 Layer 11 Head 9. For PCLens, each color highlights sentences corresponding to a different principal component, i.e. PC 0 , PC 1 , PC 2 , PC 3 , with displayed image-descriptions for PC0  and PC2 .

A.8 Optimal Number of PCs across the Methods for Removing Spurious Correlation (QuerySystem and PCSelection)

In this section, we provide additional plots illustrating the selection of the number of principal components k through a simple grid search, tracking the worst-group accuracy for all methods in the experiment of Section 5.5. Detailed per-model comparisons of the optimization process under each condition are presented in the following figures, where the dotted line represents the maximum accuracy achieved during the process, while the dashed line indicates the Pareto-optimal solution that balances both total and worst-class accuracy. Figure A.13 (Keep PCs Class) displays, for each ViT model, the worst-class accuracy (left) versus the total accuracy (right) when retaining principal components aligned with the class label. Figure A.14 (Remove PCs Spurious) shows analogous plots when removing components associated with the spurious water–land–background concept. Figure A.15 (Keep PCs Physical) presents the same plots under the constraint of preserving principal components correlated with physical attributes.

A. APPENDIX

ViT-H-14 – Layer 31 – Head 11	
	TextSpan
An image of a small cat and a large dog next to each other. An image of an infant taking their first steps. An image of a large wild cat. An image of a big, gentle dog. An image of three axes forming a 3D space: X, Y, and Z.	An image of the concept of most. An image illustrating the concept of a bit. An image of a child with a guardian. An image of pop. An image representing the concept of east.
	PCLens
An image of a dog with a water-resistant coat. An image of a pet dog. An image of a well-built dog. An image of a muscular dog. An image of a cat with a long whiskered face. An image of a slender-bodied cat. An image of a cat with a bushy tail. An image of a cat with a muscular build.	An image of an infant taking their first steps. An image of a baby. An image of a baby with big, curious eyes. An image of a small baby bed. An image of a grey fox. An image of a maned wolf. An image of a fox. An image of a tall, fox-like animal.
PC 0 (dog)	PC 3 (fox-like)

Figure A.7: A comparison of TextSpan with PCLens for ViT-H-14 Layer 31 Head 11. For PCLens, each color highlights sentences corresponding to a different principal component, i.e. PC 0 , PC 1 , PC 2 , PC 3 , with displayed image-descriptions for PC 0 and PC 3 .

Table A.1: Comparison zero-shot accuracy of different ViT models on CIFAR10 datasets using a subset of PCs from QuerySystem, querying physical description of differences between the selected class labels. Each cell has the baseline accuracy on the top and the resulting experiment accuracy on the bottom, together with the number of PCs used by QuerySystem for each class label.

Model Name	ViT-B-32	ViT-B-16	ViT-L-14	ViT-H-14
CIFAR10	93.40% 93.90% (1698)	94.50% 95.10% (2016)	96.70% 97.10% (2598)	98.20% 98.30% (2829)

A.9 Prompt Enhancing by Visual Explanation (QuerySystem and PCSelection)

In this section, motivated by the finding from Section 5.4.2 regarding the enhanced accuracy observed for a label 1 when using a set of principals components selected by QuerySystem, we investigate whether it is possible to improve the model’s performance on certain classification tasks by using only these specific PCs during the reconstruction of the residual stream. The goal is to guide the model to recognize specific classes by explicitly instructing it

A.9. Prompt Enhancing by Visual Explanation (QuerySystem and PCSelection)

ViT-B-32 - Layer 11 - Head 2 - PC 0

(Black-brown)	(White)
A black and orange object.	A white and silver vehicle.
A black and orange animal.	A white outerwear.
A black and brown animal.	A white vehicle.
A black and red engine.	A white and tan animal.

ViT-B-16 - Layer 11 - Head 10 - PC 0

(Black)	(White)
A black and orange animal.	A white and silver vehicle.
A black and brown animal.	A white outerwear.
A black and red engine.	A white-coated dog.
A brown and black animal.	A person in a white coat.

ViT-L-14 - Layer 21 - Head 0 - PC 0

(Black)	(White-colour-dotted)
A group of people black-dressed.	A bride in a white dress.
A black animal.	A white and tan animal.
A solid black dog.	A dog with a white-tan pattern.
A shiny, black creature.	A dog with a white and red coat.

ViT-H-14 - Layer 30 - Head 0 - PC 0

(Black)	(White)
A black automobile.	A person in a white coat.
A group of people black-dressed.	A white vehicle.
A solid black dog.	A white accessory.
A black device.	A white and tan animal.

Figure A.8: The first principal component across different CLIP ViT models, found by QuerySystem, encoding the concepts of *black* and *white*. Interestingly, this concept is consistently present across all models, with slightly different text-image associations. This PC direction encodes a semantic axis in the latent space that aligns with human intuition.

ViT-B-32 - Layer 8 - Head 6 - PC 5

(Party)	(Expression)
People laughing and playing.	A person with a serious look.
Young people laughing.	A person with a focused look.
Family members laughing.	A person with a worried look.
A group of children laughing.	Focused eyes gazing straight ahead.

ViT-B-16 - Layer 10 - Head 7 - PC 2

(Smiling kids)	(Hand-craft)
A woman smiling with children.	A hammer striking a nail.
A family taking a selfie.	An instrument's being adjusted.
A child with wide eyes.	A person tightening screws.
A child looking up at the sky.	A teapot pouring tea into a cup.

ViT-L-14 - Layer 22 - Head 15 - PC 1

(Smiling)	(Repairing)
A middle-aged man smiling.	A chain breaking apart.
A man smiling at a wedding.	A broken item being repaired.
A customer giving a thumbs up.	A mold being filled with substance.
A confident man with followers.	A broken machine.

ViT-H-14 - Layer 30 - Head 9 - PC 2

(Smiling)	(Sunset)
A joyful person holding money.	A sunset marking the end of a day.
A person holding a drink in hand.	A sunset a the end of the day.
A cheerful person waving hello.	A sunset, marking the end of a day.
A person laughing and playing.	A path to an unknown horizon.

Figure A.9: The first principal component across different CLIP ViT models, found by QuerySystem, encoding the concept of *happiness*. Interestingly, this concept is present in all models, but the opposing sides of the principal component differ in their associated concepts. In most cases, the opposites have semantically unintuitive meanings from a human perspective.

A. APPENDIX

on which physical characteristics to focus on when differentiating between classes. For example, within the CIFAR10 dataset, a frog can be distinguished from other classes by attributes such as: "Smooth, moist amphibian skin and strong hind legs, poised for agile leaps on land or water."

Setup We perform a test on the CIFAR10 dataset, where we first derive unique distinguishing characteristics for each label. This is done by prompting GPT-4o [21] to generate descriptions focusing on visual details unique to each specific class. Subsequently, we employ our `QuerySystem` to identify the corresponding subset of PCs from the CLIP ViT Encoder related to these distinguishing features. These selected PCs are then used to perform zero-shot classification.

Results We find that using this PC subset results in a slight performance improvement for all tested models, with the largest overall increase being 0.60 percentage points in total accuracy, achieved by the ViT-B-16 model. It is noteworthy that utilizing such a filtered feature space, derived only from distinguishing characteristics, can still maintain high overall accuracy. We also observe that different models require varying numbers of PCs to achieve their peak accuracy in this setup, with larger models generally utilizing a greater number of PCs.

A.9. Prompt Enhancing by Visual Explanation (QuerySystem and PCSelection)

A Party

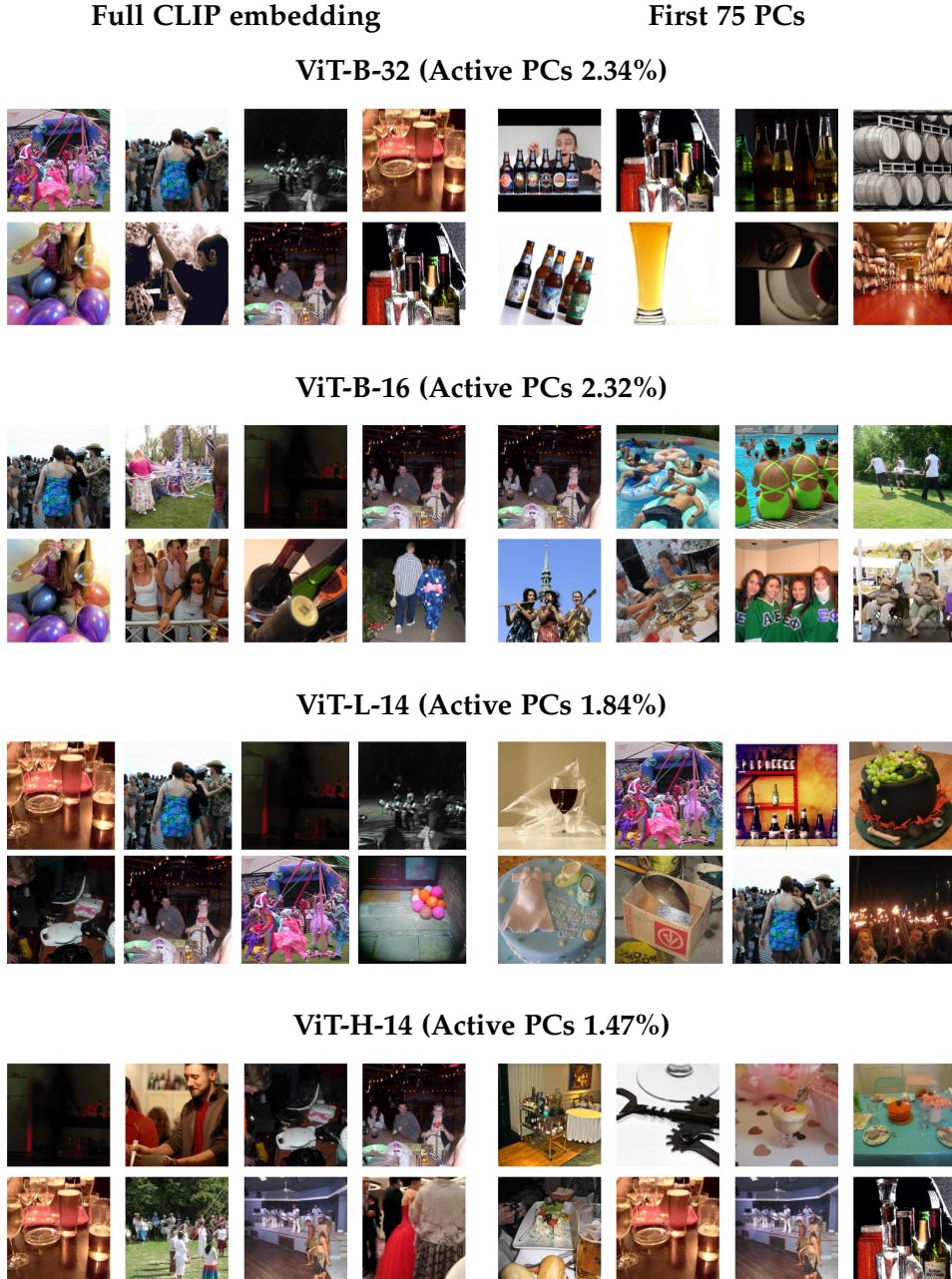


Figure A.10: Comparison of top images retrieved for the text concept *party* using the first 75 PCs found by QuerySystem, across the various CLIP ViT backbones (rows). In each row, the left columns (Full CLIP embedding) shows the top-k normal CLIP most similar image embeddings for the text concept, and the right columns (First 75 PCs) shows the corresponding images when using the modified residual stream produced by QuerySystem + PCSelection. We also provide the percentage of active PCs across the active MSA heads of the last four layers.

A. APPENDIX

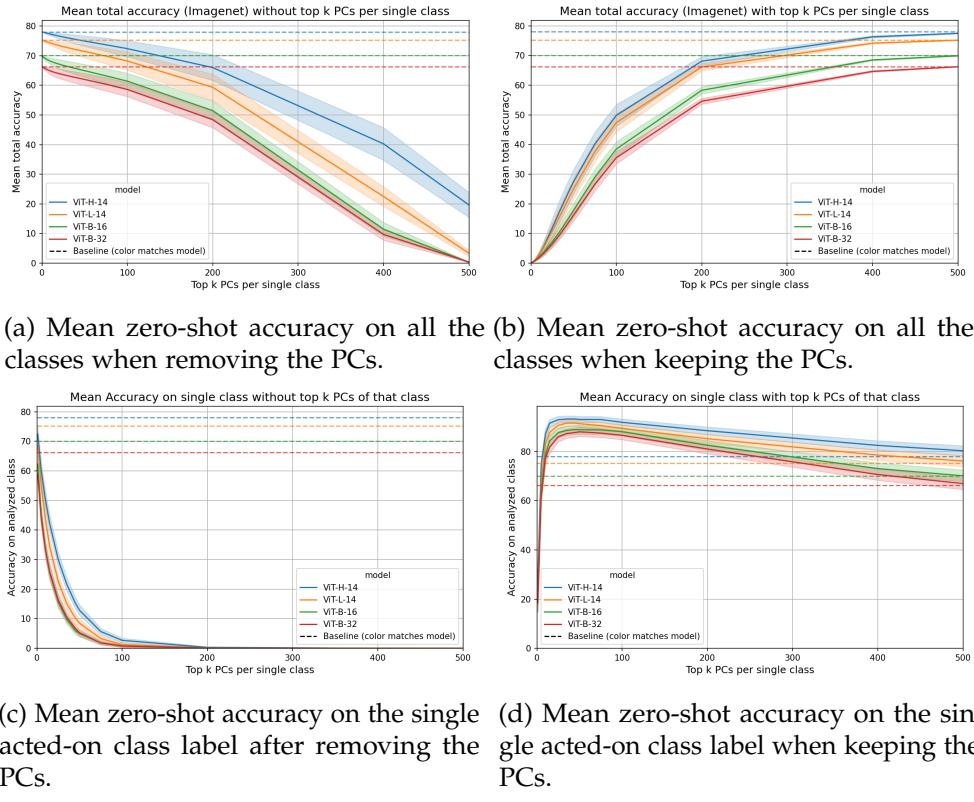
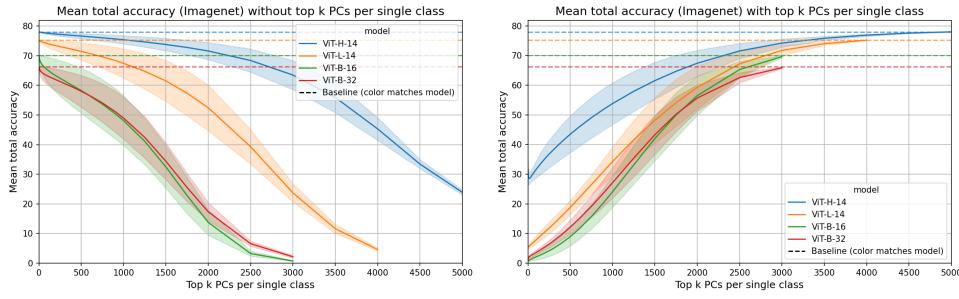
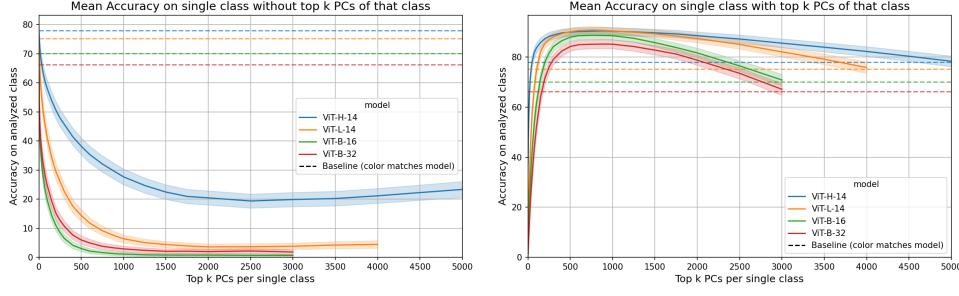


Figure A.11: Additional plots illustrating the semantic enhancement and forgetting of concepts when measuring zero-shot accuracy on ImageNet. In these plots we are acting on the final CLIP embedding projecting features in and out using the PCs found by QuerySystem.

A.9. Prompt Enhancing by Visual Explanation (QuerySystem and PCSelection)



(a) Mean zero-shot accuracy on all the classes when removing the PCs. (b) Mean zero-shot accuracy on all the classes when keeping the PCs.



(c) Mean zero-shot accuracy on the single acted-on class label after removing the PCs. (d) Mean zero-shot accuracy on the single acted-on class label when keeping the PCs.

Figure A.12: Additional plots illustrating the semantic enhancement and forgetting of concepts when measuring zero-shot accuracy on ImageNet. In these plots we are acting on the individual position of the PCs across the MSA heads with PCSelection using the PCs found by QuerySystem.

A. APPENDIX

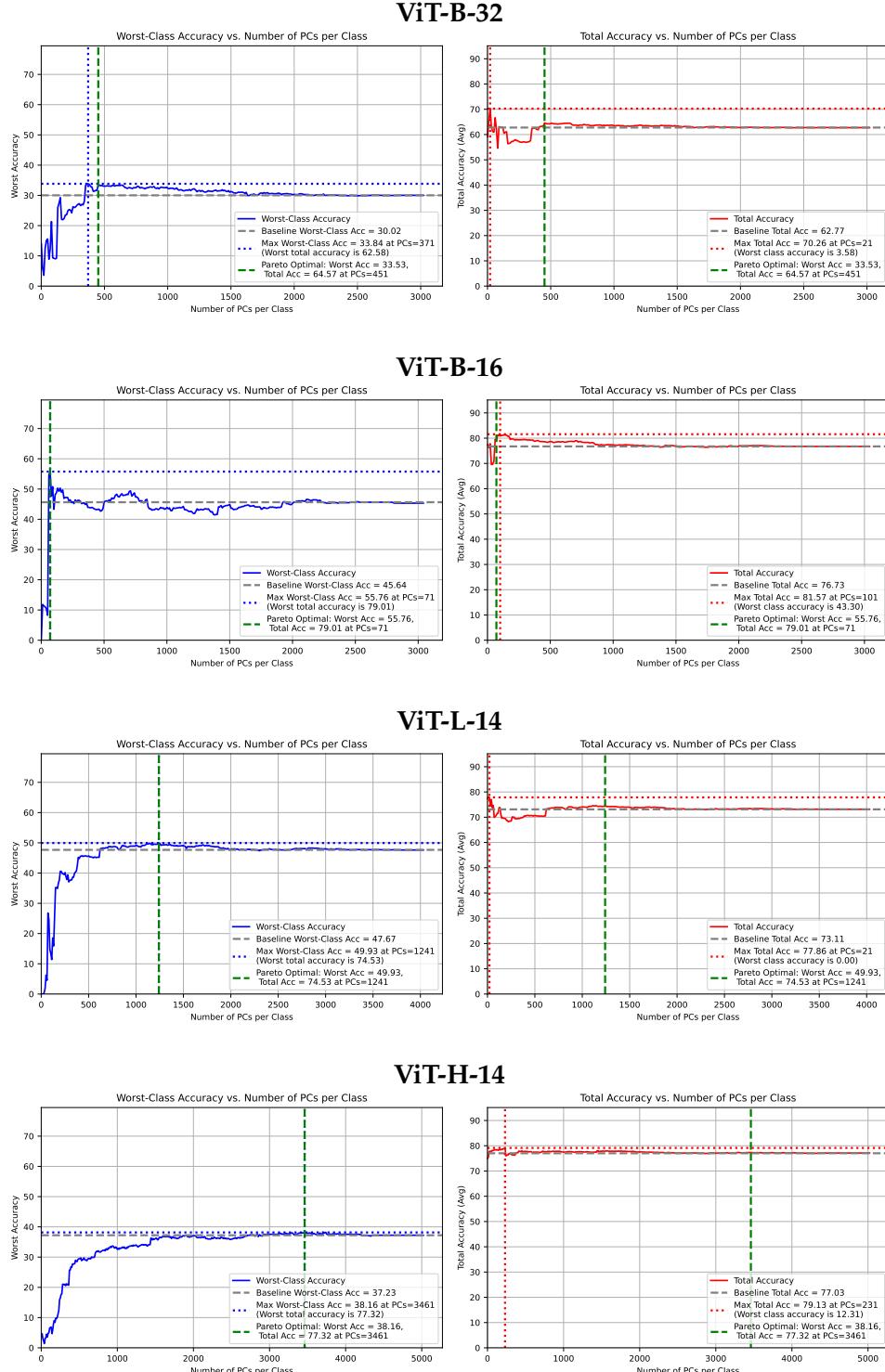


Figure A.13: Comparison of worst-class and total accuracy under the “Keep PCs (class)” condition for the four models. In each row, the left plot shows the worst-class accuracy, and the right plot shows the total accuracy for the corresponding model. In each plot, the dotted line represents the maximum accuracy achieved during the process, while the dashed line indicates the Pareto-optimal solution that balances both total and worst-class accuracy.

A.9. Prompt Enhancing by Visual Explanation (QuerySystem and PCSelection)

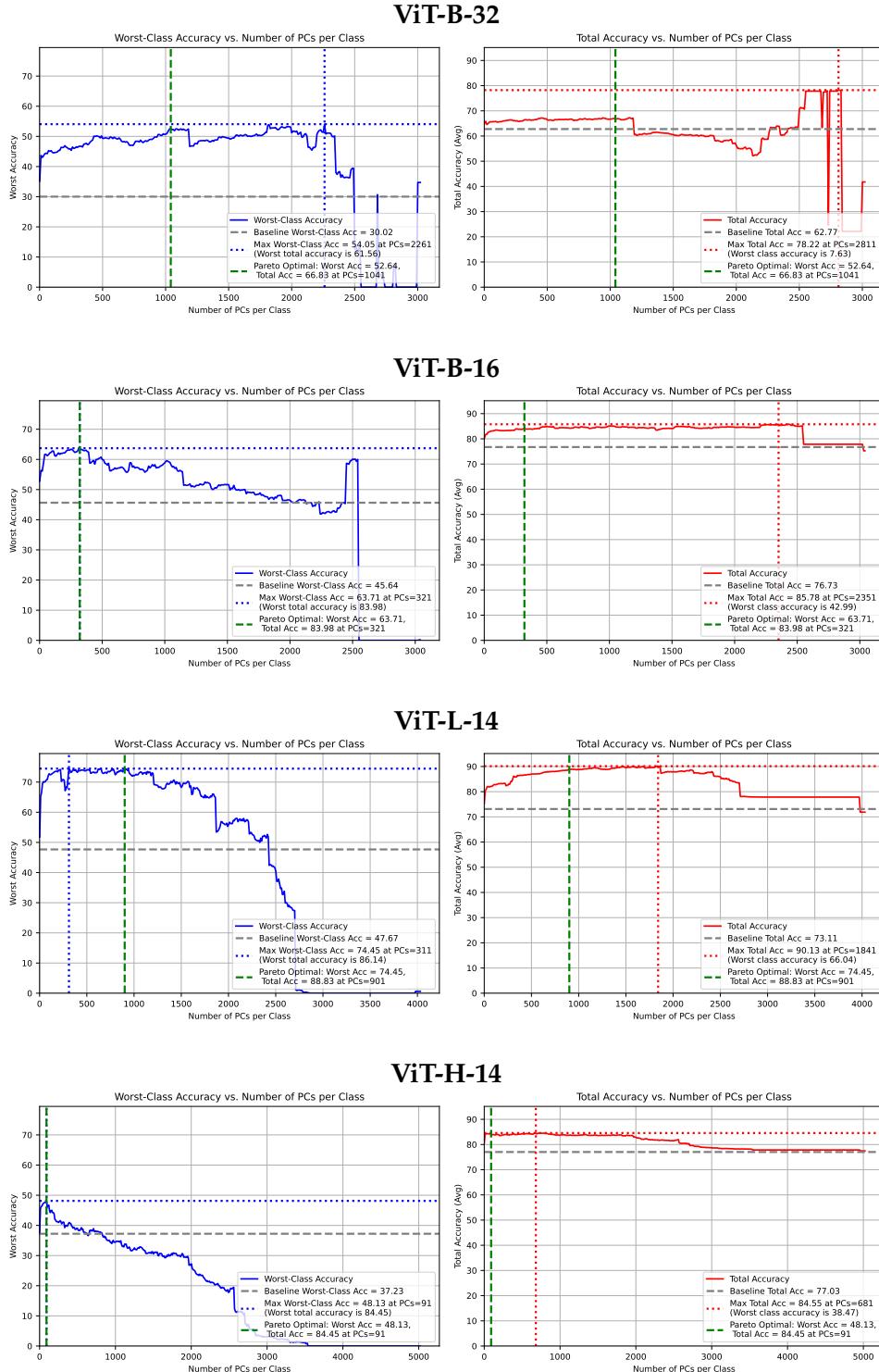


Figure A.14: Comparison of worst-class and total accuracy under the “Remove PCs (spurious)” condition for the four models. In each row, the left plot shows the worst-class accuracy, and the right plot shows the total accuracy for the corresponding model. In each plot, the dotted line represents the maximum accuracy achieved during the process, while the dashed line indicates the Pareto-optimal solution that balances both total and worst-class accuracy.

A. APPENDIX

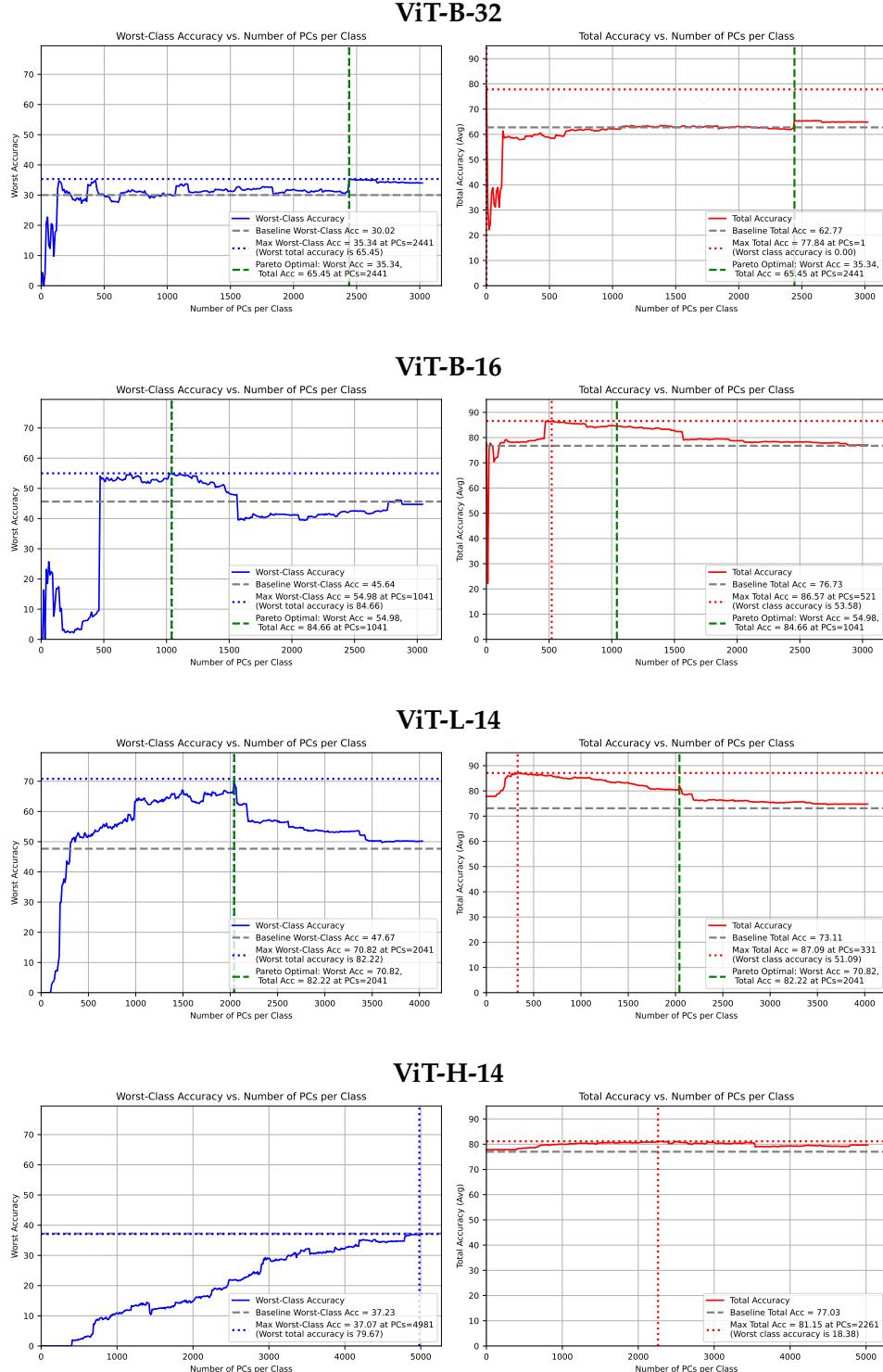


Figure A.15: Comparison of worst-class and total accuracy under the “Keep PCs (physical)” condition for the four models. In each row, the left plot shows the worst-class accuracy, and the right plot shows the total accuracy for the corresponding model. In each plot, the dotted line represents the maximum accuracy achieved during the process, while the dashed line indicates the Pareto-optimal solution that balances both total and worst-class accuracy.

Bibliography

- [1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016.
- [2] Sriram Balasubramanian, Samyadeep Basu, and Soheil Feizi. Decomposing and interpreting image representations via text in vits beyond clip, 2024.
- [3] Sriram Balasubramanian, Samyadeep Basu, and Soheil Feizi. Decomposing and interpreting image representations via text in vits beyond CLIP. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.
- [4] Lorenzo Basile, Valentino Maiorca, Luca Bortolussi, Emanuele Rodolà, and Francesco Locatello. Residual transformer alignment with spectral decomposition, 2024.
- [5] Khaled Bayoudh, Raja Knani, Fayçal Hamdaoui, and Abdellatif Mtibaa. A survey on deep multimodal learning for computer vision: advances, trends, applications, and datasets. *The Visual Computer*, 38(8):2939–2970, 2022.
- [6] Usha Bhalla, Alex Oesterling, Suraj Srinivas, Flavio P. Salmon, and Himabindu Lakkaraju. Interpreting clip with sparse linear concept embeddings (splice), 2024.
- [7] Alexander Binder, Grégoire Montavon, Sebastian Bach, Klaus-Robert Müller, and Wojciech Samek. Layer-wise relevance propagation for neural networks with local renormalization layers, 2016.
- [8] Liwei Che, Tony Qingze Liu, Jing Jia, Weiyi Qin, Ruixiang Tang, and Vladimir Pavlovic. Eazy: Eliminating hallucinations in lvlms by zeroing out hallucinatory image tokens, 2025.

BIBLIOGRAPHY

- [9] Yunkai Dang, Kaichen Huang, Jiahao Huo, Yibo Yan, Sirui Huang, Dongrui Liu, Mengxi Gao, Jie Zhang, Chen Qian, Kun Wang, Yong Liu, Jing Shao, Hui Xiong, and Xuming Hu. Explainable and interpretable multimodal large language models: A comprehensive survey, 2024.
- [10] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers, 2024.
- [11] Arun Das and Paul Rad. Opportunities and challenges in explainable artificial intelligence (xai): A survey. *arXiv preprint arXiv:2006.11371*, 2020.
- [12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.
- [13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.
- [14] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.
- [15] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2021.
- [16] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.
- [17] Mengnan Du, Ninghao Liu, and Xia Hu. Techniques for interpretable machine learning. *Communications of the ACM*, 2019.
- [18] Fabian Eitel, Anna Melkonyan, and Kerstin Ritter. Feature visualization for convolutional neural network models trained on neuroimaging data, 2022.
- [19] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition, 2022.

Bibliography

- [20] Sedigheh Eslami and Gerard de Melo. Mitigate the gap: Investigating approaches for improving cross-modal alignment in clip, 2024.
- [21] OpenAI et. all. Gpt-4 technical report, 2024.
- [22] Abrar Fahim, Alex Murphy, and Alona Fyshe. It's not a modality gap: Characterizing and addressing the contrastive gap, 2024.
- [23] Marco Fumero, Marco Pegoraro, Valentino Maiorca, Francesco Locatello, and Emanuele Rodolà. Latent functional maps, 2024.
- [24] Yossi Gandelsman, Alexei A. Efros, and Jacob Steinhardt. Interpreting clip's image representation via text-based decomposition, 2023.
- [25] Yossi Gandelsman, Alexei A. Efros, and Jacob Steinhardt. Interpreting the second-order effects of neurons in clip, 2024.
- [26] Amin Ghiasi, Hamid Kazemi, Eitan Borgnia, Steven Reich, Manli Shu, Micah Goldblum, Andrew Gordon Wilson, and Tom Goldstein. What do vision transformers learn? a visual exploration, 2022.
- [27] Gabriel Goh, Nick Cammarata, Chelsea Voss, Shan Carter, and Gabriel Hernandez. Multimodal neurons in artificial neural networks, 2021.
- [28] Jacob Grimm and Wilhelm Grimm. The bremen town musicians. In *The Complete Fairy Tales of the Brothers Grimm*. Bantam Books, New York, 2003. Originally published as **"Die Bremer Stadtmusikanten"** in *Kinder- und Hausmärchen*, 2nd ed. (1819).
- [29] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.
- [30] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.
- [31] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus), 2023.
- [32] Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. The platonic representation hypothesis, 2024.
- [33] Jiachen Jiang, Jinxin Zhou, and Zhihui Zhu. On layer-wise representation similarity: Application for multi-exit models with a single classifier, 2024.
- [34] Qian Jiang, Changyou Chen, Han Zhao, Liqun Chen, Qing Ping, Son Dinh Tran, Yi Xu, Belinda Zeng, and Trishul Chilimbi. Understanding and constructing latent modality structures in multi-modal representation learning, 2023.

BIBLIOGRAPHY

- [35] Sachin Karmani, Thanushon Sivakaran, Gaurav Prasad, Mehmet Ali, Wenbo Yang, and Sheyang Tang. Kpca-cam: Visual explainability of deep computer vision models using kernel pca, 2024.
- [36] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav), 2018.
- [37] Darina Koishigarina, Arnas Uselis, and Seong Joon Oh. Clip behaves like a bag-of-words model cross-modally but not uni-modally, 2025.
- [38] Vignesh Kothapalli. Neural collapse: A review on modelling principles and generalization, 2023.
- [39] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*, 2009.
- [40] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models, 2023.
- [41] Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning, 2022.
- [42] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2024.
- [43] Haotian Liu, Hanyu Peng, Zheyu Yu, and Yizhou Li. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023.
- [44] Avinash Madasu, Yossi Gandelsman, Vasudev Lal, and Phillip Howard. Quantifying and enabling the interpretability of clip-like models, 2024.
- [45] Avinash Madasu, Mert Yuksekgonul, Alexey Dosovitskiy, and Andrea Vedaldi. Quantifying and enabling the interpretability of clip-like models, 2024.
- [46] Mazda Moayeri, Keivan Rezaei, Maziar Sanjabi, and Soheil Feizi. Text2concept: Concept activation vectors directly from text, 2023.
- [47] Luca Moschella, Valentino Maiorca, Marco Fumero, Antonio Norelli, Francesco Locatello, and Emanuele Rodolà. Relative representations enable zero-shot latent space communication, 2022.

Bibliography

- [48] Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability, 2023.
- [49] Adam Neo and et al. Towards interpreting visual information processing in vision-language models, 2025.
- [50] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2017.
- [51] Keiron O’Shea and Ryan Nash. An introduction to convolutional neural networks, 2015.
- [52] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Conference Record of the 27th Asilomar Conference on Signals, Systems and Computers*, volume 1, pages 40–44, Pacific Grove, CA, November 1993.
- [53] Karl Pearson. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.
- [54] Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. Instruction tuning with gpt-4, 2023.
- [55] Jianing Qi, Jiawei Liu, Hao Tang, and Zhigang Zhu. Beyond semantics: Rediscovering spatial awareness in vision-language models, 2025.
- [56] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021.
- [57] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.
- [58] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI*, 2019. Accessed: 2024-11-15.
- [59] Sameera Ramasinghe, Violetta Shevchenko, Gil Avraham, and Ajanthan Thalaiyasingam. Accept the modality gap: An exploration in the hyperbolic space, 2024.

BIBLIOGRAPHY

- [60] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022.
- [61] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *arXiv preprint arXiv:2102.12092*, 2021.
- [62] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144, 2016.
- [63] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier, 2016.
- [64] Zahra Sadeghi, Roohallah Alizadehsani, Mehmet Akif CIFCI, Samina Kausar, Rizwan Rehman, Priyakshi Mahanta, Pranjal Kumar Bora, Ammar Almasri, Rami S. Alkhawaldeh, Sadiq Hussain, Bilal Alatas, Afshin Shoeibi, Hossein Moosaei, Milan Hladík, Saeid Nahavandi, and Panos M. Pardalos. A review of explainable artificial intelligence in healthcare. *Computers and Electrical Engineering*, 118:109370, 2024.
- [65] Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization, 2020.
- [66] Adam Scherlis, Kshitij Sachan, Adam S. Jermyn, Joe Benton, and Buck Shlegeris. Polysemanticity and capacity in neural networks. 2022.
- [67] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2):336–359, October 2019.
- [68] Ashish Seth, Mayur Hemanu, and Chirag Agarwal. Dear: Debiasing vision-language models with additive residuals, 2023.
- [69] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps, 2014.
- [70] Gabriela Ben Melech Stan, Estelle Aflalo, Raanan Yehezkel Rohekar, Anahita Bhawandiwalla, Shao-Yen Tseng, Matthew Lyle Olson, Yaniv Gurwicz, Chenfei Wu, Nan Duan, and Vasudev Lal. Lvlm-interpret: An interpretability tool for large vision-language models, 2024.

Bibliography

- [71] Xin Tang, Jiawei Zhang, Yichun He, Xinhe Zhang, Zuwan Lin, Sebastian Partarrieu, Emma Bou Hanna, Zhaolin Ren, Hao Shen, Yuhong Yang, Xiao Wang, Na Li, Jie Ding, and Jia Liu. Explainable multi-task learning for multi-modality biological data analysis. *Nat. Commun.*, 14(1):2546, May 2023.
- [72] Umberto M. Tomasini, Leonardo Petrini, Francesco Cagnetta, and Matthieu Wyart. How deep convolutional neural networks lose spatial information with training, 2022.
- [73] Joel A. Tropp, Anna C. Gilbert, and Martin J. Strauss. Algorithms for simultaneous sparse approximation. part i: Greedy pursuit. *Signal Processing*, 86(3):572–588, 2006. Sparse Approximations in Signal and Image Processing.
- [74] M.A. Turk and A.P. Pentland. Face recognition using eigenfaces, 1991.
- [75] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.
- [76] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.
- [77] Martina G. Vilas, Timothy Schaumlöffel, and Gemma Roig. Analyzing vision transformers for image classification in class embedding space, 2023.
- [78] Jiaqi Wang, Hanqi Jiang, Yiheng Liu, Chong Ma, Xu Zhang, Yi Pan, Mengyuan Liu, Peiran Gu, Sichen Xia, Wenjun Li, et al. A comprehensive review of multimodal large language models: Performance and challenges across different tasks. *arXiv preprint arXiv:2408.01319*, 2024.
- [79] Lehan Wang, Haonan Wang, Honglong Yang, Jiaji Mao, Zehong Yang, Jun Shen, and Xiaomeng Li. Interpretable bilingual multimodal large language model for diverse biomedical tasks. *arXiv preprint arXiv:2410.18387*, 2024.
- [80] Hanguang Xiao, Feizhong Zhou, Xingyue Liu, Tianqi Liu, Zhipeng Li, Xin Liu, and Xiaoxuan Huang. A comprehensive survey of large language models and multimodal large language models in medicine. *arXiv preprint arXiv:2405.08603*, 2024.
- [81] Wenli Yang, Yuchen Wei, Hanyu Wei, Yanyu Chen, Guan Huang, Xiang Li, Renjie Li, Naimeng Yao, Xinyi Wang, Xiaotong Gu, Muhammad Bilal Amin, and Byeong Kang. Survey on explainable ai: From approaches,

BIBLIOGRAPHY

- limitations and applications aspects. *Human-Centric Intelligent Systems*, 3(3):161–188, Sep 2023.
- [82] Linli Yao, Lei Li, Shuhuai Ren, Lean Wang, Yuanxin Liu, Xu Sun, and Lu Hou. Deco: Decoupling token compression from semantic abstraction in multimodal large language models. *arXiv preprint arXiv:2405.20985*, 2024.
 - [83] Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it?, 2023.
 - [84] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks, 2013.
 - [85] Yuhui Zhang, Alyssa Unell, Xiaohan Wang, Dhruba Ghosh, Yuchang Su, Ludwig Schmidt, and Serena Yeung-Levy. Why are visually-grounded language models bad at image classification?, 2024.
 - [86] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023.
 - [87] Ziqin Zhou, Bowen Zhang, Yinjie Lei, Lingqiao Liu, and Yifan Liu. Zegclip: Towards adapting clip for zero-shot semantic segmentation, 2023.