Visual symbolic mechanisms: Emergent symbol processing in vision language models

Rim Assouel*
Mila - Québec AI Institute
Université de Montréal
assouelr@mila.quebec

Declan Campbell*
Princeton Neuroscience Institute
idcampbell@princeton.edu
* Equal contribution.

Taylor Webb
Mila - Québec AI Institute
Université de Montréal
taylor.w.webb@gmail.com

Abstract

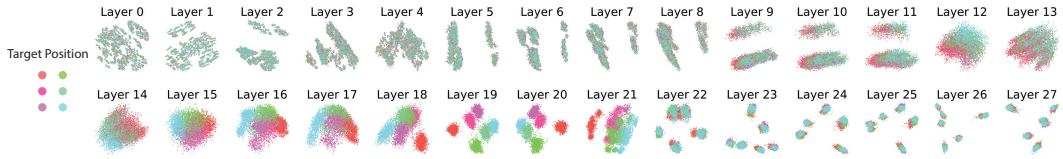
To accurately process a visual scene, observers must bind features together to represent individual objects. This capacity is necessary, for instance, to distinguish an image containing a red square and a blue circle from an image containing a blue square and a red circle. Recent work has found that language models solve this 'binding problem' via a set of symbol-like, content-independent indices, but it is unclear whether similar mechanisms are employed by vision language models (VLMs). This question is especially relevant, given the persistent failures of VLMs on tasks that require binding. Here, we identify a set of emergent symbolic mechanisms that support binding in VLMs via a content-independent, spatial indexing scheme. Moreover, we find that binding errors can be traced directly to failures in these mechanisms. Taken together, these results shed light on the mechanisms that support symbol-like processing in VLMs, and suggest possible avenues for addressing the persistent binding failures exhibited by these models.

1 Introduction

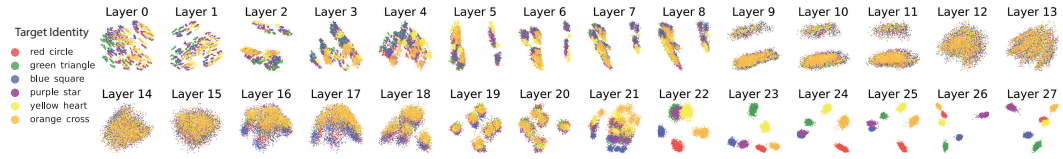
Visual inputs are inherently compositional, often comprising familiar features such as shapes and colors that must be flexibly combined to represent complex scenes. Neural networks, including vision-language models (VLMs), are known to learn such compositional representations, but this introduces the classic 'binding problem'—how to correctly associate features to form coherent object-level representations in context [Treisman and Gelade, 1980, Greff et al., 2020, Campbell et al., 2024, Assouel et al., 2025]. Recent work in language models has revealed emergent symbolic mechanisms, such as content-independent binding IDs, that enable models to track and manipulate entities and attributes in context [Feng and Steinhardt, 2023, Yang et al., 2025]. However, it remains unclear whether VLMs employ similar mechanisms to solve the binding problem for visual entities.

This question is particularly important because many of the persistent failures of VLMs—such as errors in counting or visual search—can be traced to binding errors, where models fail to correctly associate features with objects [Rahmanzadehgervi et al., 2024, Campbell et al., 2024]. Understanding the mechanisms underlying binding in VLMs is therefore crucial for improving their reliability and performance. In this work, we present evidence that VLMs use emergent, symbol-like mechanisms to bind features in multi-object scenes. Specifically, we show that VLMs employ content-independent spatial indices, or position IDs, to organize and retrieve object features. We identify two key mechanisms: (1) **Position ID Heads**, which compute spatial indices for target objects, and (2) **Feature Retrieval Heads**, which use these indices to access object features. Through representational, causal, and intervention analyses, we demonstrate that these mechanisms are central to feature binding—and that their failure directly leads to binding errors. Our findings not only clarify the origins of binding failures in VLMs, but also suggest concrete strategies for improving their symbolic reasoning and scene understanding. In this work we aim to give representational, mechanistic and interventional

CogInterp@NeurIPS 2025: First Workshop on Interpreting Cognition in Deep Learning Models



(a) PCA results coded by target object position.



(b) PCA results coded by target object features.

Figure 2: **Principal Component Analyses.** Hidden state embeddings at last token position projected onto the top 2 principal components. Results are coded either by the (a) position or (b) identity (i.e., features) of the target object.

evidence of the usage of symbol-like position ID by MLLM in order to bind to visual entities in context.

2 Identifying Visual Object Binding Mechanisms

To investigate mechanisms underlying multi-object visual processing, we use a scene description task: given an image with multiple objects and an incomplete caption, the model must describe the remaining objects. This task is challenging for VLMs, especially with many objects or overlapping features, and is closely tied to the binding problem [Campbell et al., 2024]. All experiments in the main text use Qwen2-VL-7B-Instruct [Wang et al., 2024]; generalization to other models is shown in Appendix M and Appendix H.

2.1 Representational Analyses

We first analyze how the model represents target objects when completing captions. Using PCA and representational similarity analysis [Kriegeskorte et al., 2008] (RSA), we find a two-stage process: intermediate layers encode the spatial position of the target object, while later layers encode its features (color, shape). This suggests the model first computes a spatial index, then uses it to retrieve object features (see Figure 14 and Figure 1). Full methodological details are provided in Appendix E.

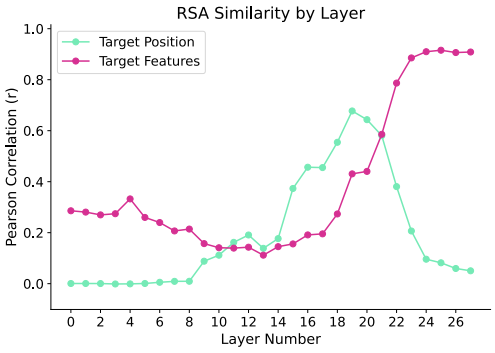


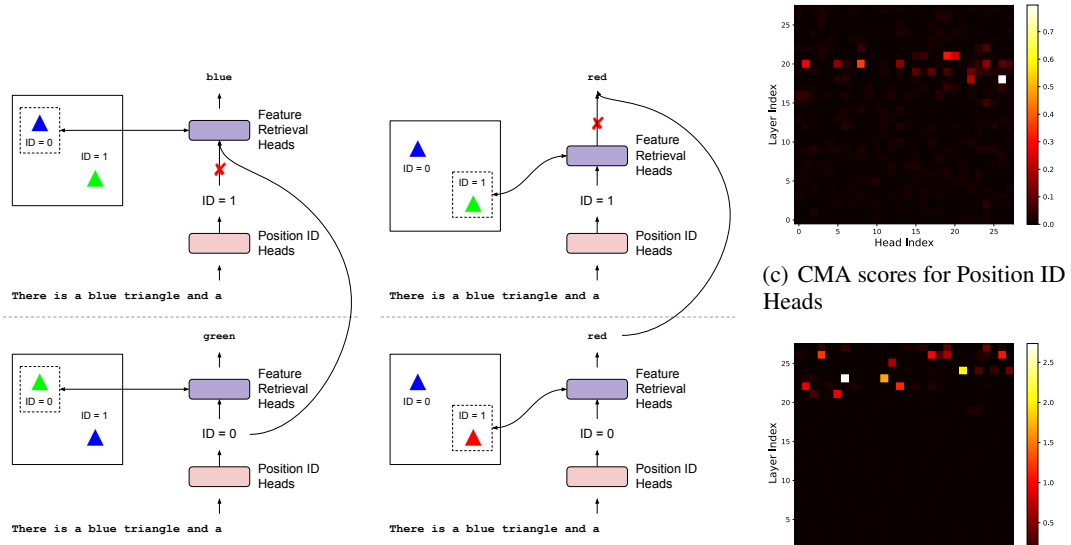
Figure 1: **Representational Similarity Analyses**. RSA results for hidden state at last token position.

2.2 Mechanistic Analyses

To test the hypothesized two-stage mechanism, we perform causal mediation analysis (CMA) targeting attention heads at the last token position. Our goal is to identify distinct sets of heads: "position ID heads" in layers 18–21, causally involved in computing spatial indices, and "feature retrieval heads" in layers 21–27, causally involved in using these indices to retrieve object features. The location of the two sets of heads align with our representational analyses (see Figure 3). Details of CMA setup and scoring are in Appendix D.

2.3 Intervention Analyses

In this section our goal is to further validate the use of content-independent, symbol-like indices through some targeted interventions. Our previous analyses predict that the model



(a) CMA targeting Position ID Heads (b) CMA targeting Feature Retrieval Heads

(c) CMA scores for Position ID Heads

(d) CMA scores for Feature Retrieval Heads

Figure 3: **Causal Mediation Analyses.** Illustrations depicting the causal mediation analyses (CMA) targeting (a) Position ID Heads and (b) Feature Retrieval Heads. Results showing CMA scores for (c) Position ID Heads and (d) Feature Retrieval Heads.

uses position IDs to selectively attend to part of the image. Specifically we hypothesize that the feature retrieval heads match the keys of the image patches using these position IDs. We intervened directly on the keys of the image patches in a simple color retrieval task. In this task, the model must complete the following sentence: "In this image, the color of the [shape$_A$] is", where shape$_A$ is the shape of the object at position A. The intervention consists of swapping two objects IDs by adding to the keys the average difference of the keys over the whole dataset based on their position ID. We find that swapping position IDs between objects reliably causes the model to retrieve the swapped features, with this intervention maximally effective when it is applied in the layers where we identified feature retrieval heads (layers 21-28), with intervention efficacy scores ranging from 93%-99% (Figure 5(a)). Intervention procedures and quantitative results are further described in Appendix G. We also show additional interventions results for a **wider range of models on realistic images** in Appendix H.

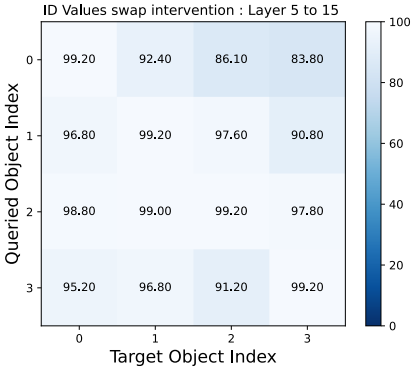
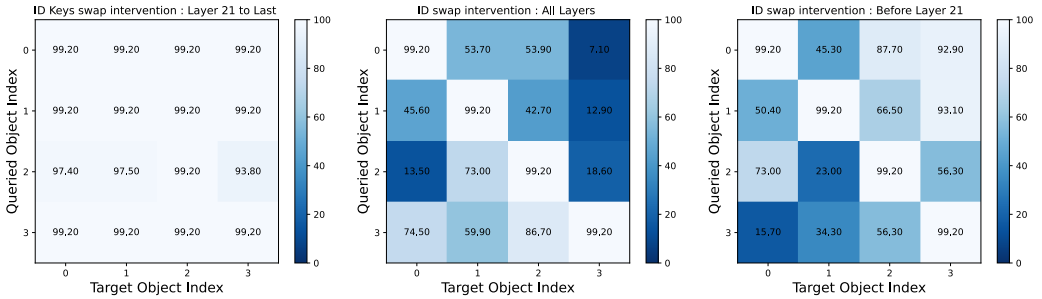


Figure 4: The model is tasked to complete the following template: "The color of the [shape of object index A] is". Object values IDs are swapped between the queried object index A (y-axis) and the target object index B (x-axis).

3 Analysis of Binding Errors

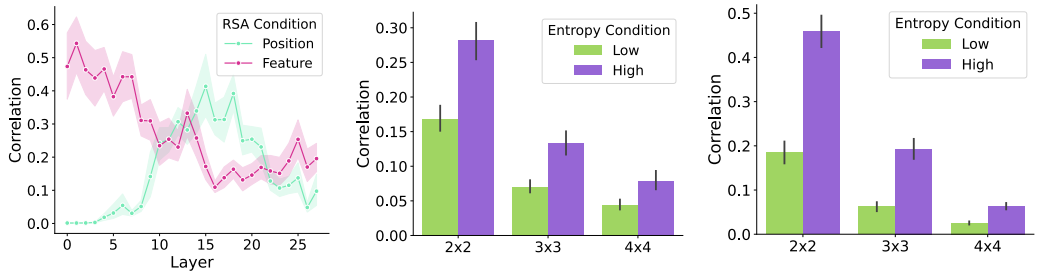
Having identified some of the mechanisms that support binding and symbol processing in VLMs, we next investigated how these mechanisms contribute to binding errors. Specifically, we examined how the model's use of position IDs affects its ability to correctly bind features to objects or locations. For these analyses, we used a completion task with varying numbers of objects and feature entropy levels, two factors that are known to affect the occurrence of binding errors [Campbell et al., 2024].

We first examined the representations at the positions corresponding to objects already listed in the incomplete caption. RSA revealed that earlier layers primarily encode object features, while intermediate layers encode object positions (Figure 6a). This suggests an initial processing stage



(a) Layers with Retrieval Heads (b) All Layers (c) Layers before Retrieval Heads

Figure 5: **Intervention targeting Feature Retrieval Heads.** The model is tasked to complete the following template: "The color of the [shape of object index A] is". Object keys IDs are swapped between the queried object index A (y-axis) and the target object index B (x-axis). Intervention was performed on three sets of layers: (a) layers 21-28, where CMA identified Feature Retrieval Heads, (b) all layers, or (c) layers before layer 21.



(a) RSA at Caption Token (b) Position RSA - Caption (c) Position RSA - Last Token

Figure 6: **Representational Analyses of Binding Errors.** (a) Position and feature RSA across layers for caption tokens describing objects. (b) Position RSA for high vs. low entropy condition, at position of caption tokens. (c) Position RSA for high vs. low entropy condition, at final token position.

in which the model retrieves the position IDs of described objects by matching their features to the image. To further validate this, we adapted the color retrieval intervention (see Section 2.3), this time intervening on the values of the image patches (i.e., the information communicated to the caption tokens). We found that interventions targeting layers 5–15—where position information peaks for caption tokens—significantly altered model behavior (Figure 4), supporting the existence of this early feature-to-position matching stage.

We then asked whether the strength of position ID representations depends on feature entropy. Consistent with the higher binding error rates observed for low-entropy images (where objects share features), we found that position ID representations were weaker in the low-entropy condition compared to the high-entropy condition (Figures 6(b) and 6(c)). This effect was present both during the retrieval of position IDs for objects described in the caption and during the generation of the target position ID at the final token position. We further validate the link between binding errors and altered position ID processing with additional interventions in Appendix K.

Conclusion In summary, our analyses reveal that vision-language models employ emergent, symbol-like mechanisms to solve the binding problem in multi-object visual scenes. By leveraging content-independent position IDs and specialized attention heads, these models can systematically bind features to objects. However, we also show that failures in these mechanisms directly contribute to persistent binding errors, especially in challenging, low-entropy contexts. Understanding and improving these symbolic processes paves the way to more robust representations and training of MLLMs.

References

Anne M Treisman and Garry Gelade. A feature-integration theory of attention. *Cognitive psychology*, 12(1):97–136, 1980.

Klaus Greff, Sjoerd Van Steenkiste, and Jürgen Schmidhuber. On the binding problem in artificial neural networks. *arXiv preprint arXiv:2012.05208*, 2020.

Declan Campbell, Sunayana Rane, Tyler Giallanza, Camillo Nicolò De Sabbata, Kia Ghods, Amogh Joshi, Alexander Ku, Steven Frankland, Tom Griffiths, Jonathan D Cohen, et al. Understanding the limits of vision language models through the lens of the binding problem. *Advances in Neural Information Processing Systems*, 37:113436–113460, 2024.

Rim Assouel, Pietro Astolfi, Florian Bordes, Michal Drozdzal, and Adriana Romero-Soriano. Object-centric binding in contrastive language-image pretraining, 2025. URL https://arxiv.org/abs/2502.14113.

Jiahai Feng and Jacob Steinhardt. How do language models bind entities in context? *arXiv preprint arXiv:2310.17191*, 2023.

Yukang Yang, Declan Campbell, Kaixuan Huang, Mengdi Wang, Jonathan Cohen, and Taylor Webb. Emergent symbolic mechanisms support abstract reasoning in large language models. *arXiv preprint arXiv:2502.20332*, 2025.

Pooyan Rahmanzadehgervi, Logan Bolton, Mohammad Reza Taesiri, and Anh Totti Nguyen. Vision language models are blind. In *Proceedings of the Asian Conference on Computer Vision*, pages 18–34, 2024.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Nikolaus Kriegeskorte, Marieke Mur, and Peter A Bandettini. Representational similarity analysis-connecting the branches of systems neuroscience. *Frontiers in systems neuroscience*, 2:249, 2008.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, et al. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*, 2022.

Eric Todd, Millicent L Li, Arnab Sen Sharma, Aaron Mueller, Byron C Wallace, and David Bau. Function vectors in large language models. *arXiv preprint arXiv:2310.15213*, 2023.

Jiahai Feng, Stuart Russell, and Jacob Steinhardt. Monitoring latent world states in language models with propositional probes. *arXiv preprint arXiv:2406.19501*, 2024.

Michael Lepori, Alexa Tartaglini, Wai Keen Vong, Thomas Serre, Brenden M Lake, and Ellie Pavlick. Beyond the doors of perception: Vision transformers represent relations between objects. *Advances in Neural Information Processing Systems*, 37:131503–131544, 2024.

Cheng Tang, Brenden Lake, and Mehrdad Jazayeri. An explainable transformer circuit for compositional generalization. *arXiv preprint arXiv:2502.15801*, 2025.

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

Michal Golovanevsky, William Rudman, Vedant Palit, Ritambhara Singh, and Carsten Eickhoff. What do vlms notice? a mechanistic interpretability pipeline for gaussian-noise-free text-image corruption and evaluation. *arXiv preprint arXiv:2406.16320*, 2024.

Omri Kaduri, Shai Bagon, and Tali Dekel. What's in the image? a deep-dive into the vision of vision language models. *arXiv preprint arXiv:2411.17491*, 2024.

Samyadeep Basu, Martin Grayson, Cecily Morrison, Besmira Nushi, Soheil Feizi, and Daniela Massiceti. Understanding information storage and transfer in multi-modal large language models. *arXiv preprint arXiv:2406.04236*, 2024.

Sunayana Rane, Alexander Ku, Jason Baldridge, Ian Tenney, Tom Griffiths, and Been Kim. Can generative multimodal models count to ten? In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 46, 2024.

Chenhui Zhang and Sherrie Wang. Good at captioning bad at counting: Benchmarking gpt-4v on earth observation data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7839–7849, 2024.

Melanie Mitchell, Alessandro B Palmarini, and Arseny Moskvichev. Comparing humans, gpt-4, and gpt-4v on abstraction and reasoning tasks. *arXiv preprint arXiv:2311.09247*, 2023.

Eunice Yiu, Maan Qraitem, Anisa Noor Majhi, Charlie Wong, Yutong Bai, Shiry Ginosar, Alison Gopnik, and Kate Saenko. Kiva: Kid-inspired visual analogies for testing large multimodal models. *arXiv preprint arXiv:2407.17773*, 2024.

Shuhao Fu, Andrew Jun Lee, Anna Wang, Ida Momennejad, Trevor Bihl, Hongjing Lu, and Taylor W Webb. Evaluating compositional scene understanding in multimodal generative models. *arXiv preprint arXiv:2503.23125*, 2025.

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. URL https://arxiv.org/abs/2210.01936.

Zenon W Pylyshyn. Visual indexes, preconceptual objects, and situated vision. *Cognition*, 80(1-2): 127–158, 2001.

Melvyn A Goodale and A David Milner. Separate visual pathways for perception and action. *Trends in neurosciences*, 15(1):20–25, 1992.

James CR Whittington, Timothy H Muller, Shirley Mark, Guifen Chen, Caswell Barry, Neil Burgess, and Timothy EJ Behrens. The tolman-eichenbaum machine: unifying space and relational memory through generalization in the hippocampal formation. *Cell*, 183(5):1249–1263, 2020.

Randall C O'Reilly, Charan Ranganath, and Jacob L Russin. The structure of systematicity in the brain. *Current directions in psychological science*, 31(2):124–130, 2022.

Taylor W Webb, Steven M Frankland, Awni Altabaa, Simon Segert, Kamesh Krishnamurthy, Declan Campbell, Jacob Russin, Tyler Giallanza, Randall O'Reilly, John Lafferty, et al. The relational bottleneck as an inductive bias for efficient abstraction. *Trends in Cognitive Sciences*, 2024.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *Advances in neural information processing systems*, 33:11525–11538, 2020.

Sindy Löwe, Phillip Lippe, Francesco Locatello, and Max Welling. Rotating features for object discovery. *Advances in Neural Information Processing Systems*, 36:59606–59635, 2023.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *arXiv preprint arXiv:2409.17146*, 2024.

Ian Covert, Tony Sun, James Zou, and Tatsunori Hashimoto. Locality alignment improves vision-language models, 2025. URL https://arxiv.org/abs/2410.11087.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features, 2025. URL https://arxiv.org/abs/2502.14786.

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small. *arXiv preprint arXiv:2211.00593*, 2022.

A Related Work

A number of studies have investigated the emergent mechanisms that support symbol-like processing in language models and other neural networks. This work has identified a number of surprisingly interpretable and structured mechanisms, including induction heads [Olsson et al., 2022], function vectors [Todd et al., 2023], binding IDs [Feng and Steinhardt, 2023, Feng et al., 2024], and emergent symbolic mechanisms [Yang et al., 2025]. Related work has found convergent evidence for such emergent mechanisms in transformer language models and vision transformers that are trained in controlled settings [Lepori et al., 2024, Tang et al., 2025]. Although a number of recent studies have begun to apply mechanistic interpretability techniques to understand VLMs [Neo et al., 2024, Golovanevsky et al., 2024, Kaduri et al., 2024, Basu et al., 2024], it has not yet been established whether VLMs possess emergent mechanisms for symbol processing similar to those that have been identified in text-only language models, as our results suggest.

Several studies have probed the capacity of VLMs to process multi-object scenes, revealing a number of failure modes for tasks such as counting [Rahmanzadehgervi et al., 2024, Rane et al., 2024, Zhang and Wang, 2024], visual search [Campbell et al., 2024], and visual analogy [Mitchell et al., 2023, Yiu et al., 2024, Fu et al., 2025]. Notably, all of these failure modes appear to be directly related to difficulty with binding object features and relations [Yuksekgonul et al., 2023, Campbell et al., 2024, Assouel et al., 2025]. Our results reveal some of the key mechanisms that are involved in these binding failures, suggesting potential avenues for further improvement of VLM architectures and training.

Finally, our findings are closely related to work that has identified the neural and psychological correlates of visual binding. In cognitive science, visual indexing theory [Pylyshyn, 2001] has postulated the existence of content-independent, visual indices that are used for binding object features. In neuroscience, there is a broad distinction between brain regions involved in representing concrete features (e.g., shapes and colors), and brain regions involved in representing space [Goodale and Milner, 1992]. The spatial representations in this latter set of brain regions also appear to be more broadly involved in abstract, symbol-like processing [Whittington et al., 2020, O'Reilly et al., 2022, Webb et al., 2024]. The emergent mechanisms that we have identified in VLMs, in which visual space serves as a content-independent scaffold for binding object features, thus have interesting parallels to findings from cognitive science and neuroscience.

B Discussion

In this work, we have identified a set of emergent mechanisms that support binding and multi-object processing in VLMs. These mechanisms are similar to mechanisms that support symbol-like processes in language models, most notably including binding IDs [Feng and Steinhardt, 2023] and emergent symbolic mechanisms [Yang et al., 2025], but our results extend these findings to visual processing. In particular, we find that symbolic indexing in VLMs is fundamentally organized around a *spatial* scaffold. Across a set of representational, mechanistic, and intervention analyses, we identify convergent evidence for a two-stage process, in which a VLM first computes the spatial index for a target object, and then uses that index to retrieve the object's features. We also identify evidence for an earlier stage that computes the indices corresponding to objects already described in the caption. These results begin to uncover the mechanisms that support symbol processing in VLMs.

Importantly, we also find that the identified visual symbolic mechanisms are directly implicated in the binding failures that limit VLMs' capacity for multi-object processing. We find that position IDs are less accurately represented in conditions that typically lead to binding errors, such as images where multiple objects share features. Furthermore, we find that intervening on these position IDs, by patching them from conditions that are less likely to produce binding errors, can partially rescue model performance. These results suggest that the identified mechanisms are an important target for improving multi-object processing in VLMs. A number of avenues exist for improving the performance of these mechanisms. One approach is to focus on architectural improvements, such as slot-based architectures [Locatello et al., 2020] or other object-centric approaches [Löwe et al., 2023]. Another approach is to focus on explicitly training VLMs to perform spatial indexing, e.g., training them to 'point' to objects via spatial coordinates [Deitke et al., 2024], or to investigate the impact of the pretraining strategy on the downstream capacity to represent objects [Covert et al., 2025, Tschannen et al., 2025]. Whether through improved architectures or new training methods,

an important priority for future work will be to improve the mechanisms that support binding and symbolic processing in VLMs.

C Data Description

This section details the synthetic image datasets generated for Representational Similarity Analysis (RSA), patching experiments, and Principal Component Analysis (PCA). All images use colored shapes on grids with 28×28 pixel patches, where each object occupies a 2×2 patch area (56×56 pixels). RSA and patching experiments were conducted using datasets generated by placing objects along a 2×2, 3×3, and 4×4 grid, while PCA analyses used a 3×2 grid configuration.

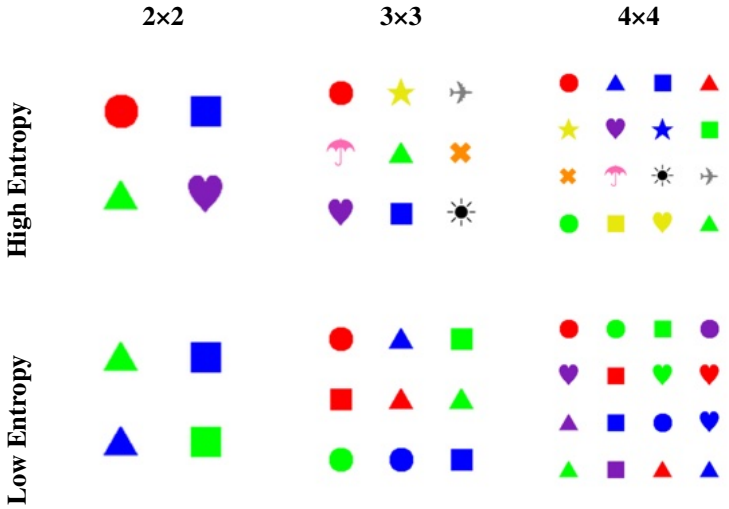


Figure 7: Stimulus conditions

Dataset Generation Protocol: For each dataset, N unique objects (color-shape conjunctions) match the number of grid positions. Each image contains one target object at a fixed position, with the remaining N-1 objects randomly permuted across remaining positions. We generate K trials for each combination of object identity and grid position, yielding K × N × N total trials per dataset.

RSA and Activation Patching Datasets: We created three dataset types varying in grid size and entropy. Low entropy datasets use all possible conjunctions from limited color and shape sets, while high entropy datasets select the most distinct conjunctions from larger sets. The 2×2 grid (280×280 pixels; 4 objects) uses 2 colors × 2 shapes for low entropy and the 4 most distinct conjunctions from 4 colors × 4 shapes for high entropy (both using 100 trials/combination; 1600 total). The 3×3 grid (392×392 pixels; 9 objects) selects 9 conjunctions from 3 colors × 3 shapes for low entropy and 9 objects with unique color and shape for high entropy (both using 50 trials/combination; 4,050 per condition). The 4×4 grid (504×504 pixels; 16 objects) combines 4 colors × 4 shapes for low entropy and selects 16 conjunctions from 9 colors × 9 shapes for high entropy (both using 20 trials/combination; 5,120 per condition).

PCA Dataset: The 3×2 grid configuration (392×932 pixels) uses 6 specific color-shape conjunctions: red circle, green triangle, blue square, purple star, yellow heart, and orange cross. Objects are placed in the leftmost and rightmost columns of a 3×3 grid layout with the center column empty, generating 200 trials/combination, resulting in 7,200 total trials.

D Causal Mediation Analysis

In this section we further detail the procedure to generate the CMA plots in section 2.2. Our goal in this analysis was to identify the attention heads that are causally involved in 1) computing the spatial position of the target object, and 2) using this spatial position as a content-independent index (i.e., position ID) to retrieve the color of the target object. We refer to the heads that perform these computations as *position ID heads* and *feature retrieval heads* respectively.

In order to isolate these two sets of heads, we designed two conditions. Each CMA condition is defined by a clean context c, a patched context c^*, a clean answer token a, and an expected answer token a^*. Given a model M, we measure the causal mediation score as defined by [Wang et al., 2022]:

$$s = (M(c^*)[a^*] - M(c^*)[a]) - (M(c)[a^*] - M(c)[a]) \tag{1}$$

where $M(c)[a]$ corresponds to the logits of token a at the output of the model evaluated on input c. Intuitively, this score measures the extent to which patching activations from another context into context c (yielding the patched context c^*) has the expected effects on the model's outputs. We performed causal mediation on a simpler version of the scene description task involving only two objects. All CMA scores are averaged across 50 samples for each condition.

Condition 1 : Shared Target Feature - Switched Target ID In this condition (Figure 3(a)), we aimed to isolate the attention heads responsible for computing the position ID of the target object. To do so, we patched from an alternative context in which the objects from the clean context appeared in opposite positions. We hypothesized that switching the position of the objects would induce a corresponding switch in the position IDs assigned to those objects. The prompt for the alternative context was the same as the prompt for the clean context: "This is an image with a [color A] [shape A] and a".

Importantly, we only patched the output of attention heads at the final position in the prompt. Our prediction was that the position ID assigned to the target object (object B) in the alternative context would be the same as the position ID for object A in the clean context, and therefore that when this position ID was used to retrieve the features of the target object from the image tokens in the clean context, the features of object A would be retrieved. The expected output for this condition was thus "[color A]". By contrast, this condition should *not* induce a change in the behavior of the feature retrieval heads, as the output of these heads directly reflects the features of the target object, and the target object is the same in both the corrupted and clean contexts (object B).

The CMA scores for this condition (Figure 3(c)) revealed a set of position ID heads primarily located in layers 18-21. Notably, this result is consistent with the high position RSA scores for these layers (Figure 1), and the high degree of separability for object position visible in the PCA results for these layers (Figure 2(a)).

Condition 2 : Different Target Feature - Same Target ID In this condition (Figure 3(b)), we aimed to isolate the attention heads responsible for retrieving the features of an object based on the previously computed position ID. In this condition, one of the objects in the alternative context was the same as clean context (object A), and appeared in the same position, but the features of the object in the other position were changed (i.e., this object was changed from object B in the clean context to object C in the alternative context). The prompt was again the same for both contexts: "This is an image with a [color A] [shape A] and a". We hypothesized that the feature retrieval heads should output the features corresponding to the object from the alternative context (object C), and thus the expected answer was "[color C]". This condition should not result in a change in the behavior of the position ID heads, since the position of the objects remained the same, and therefore the same position ID should be assigned to the target object.

The CMA scores for this condition (Figrue3(d)) revealed a set of feature retrieval heads primarily located in layers 21-27, immediately following the position ID heads. This result is consistent with the presence of a two-stage process, in which the output of the position ID heads is used by the feature retrieval heads to retrieve the features of the target object. This is also consistent with the results of our representational analyses (Figure 1), in which the representation of the features of the target object peaked in later layers.

E Representation Similarity Analysis

To probe the internal representations of Qwen 2VL, we use Representational Similarity Analysis (RSA) to assess how the model's representational geometry aligns with objects' spatial positions and semantic features (color and shape). RSA quantifies alignment between representational spaces by comparing pairwise representational similarity matrices (RSMs), allowing us to evaluate whether model representations are structured according to spatial or semantic information.

E.1 Token Sources for Analysis

Our RSA approach targets two distinct sources of token-level activations that we hypothesize encode object-level information. The first source is the last token in the sequence, positioned immediately before the description of the target object. This token represents a critical decision point where the model must have integrated all relevant information to generate the appropriate object description. Analyzing this single token position yields RSMs of shape [1, T, T], where T represents the number of trials.

The second source consists of the comma tokens that punctuate each object description in the prompt. Specifically, we analyze the comma tokens that follow each of the N-1 object descriptions preceding the target. These tokens serve as natural boundaries between object descriptions and we find that they encode information about the preceding object in the description. This multi-token approach produces RSMs of shape [N-1, T, T], enabling us to track how object representations evolve through the depth of the model. For both token sources, we extract activations from two locations within the model: the residual stream and the attention block outputs.

Model RSM Construction We construct model RSMs by computing pairwise cosine similarities between activations across samples. The specific extraction process depends on the source of the activations. For the residual stream, we extract activations directly at the specified token positions, yielding a single representation per layer. For the attention blocks, we extract the output vectors from each attention head h in layer l, providing multiple representations per layer. Given a set of activations $\text{Act}(i, t)$ for token i and trial t (where the activation may come from either the residual stream or a specific attention head), we compute pairwise similarities across all trial pairs. Each entry in the resulting RSM reflects the similarity of representations across two trials.

Target RSM Construction To evaluate what type of information these representations encode, we construct two types of target RSMs. The position-based RSM captures spatial relationships using ground-truth (x, y) coordinates $\text{coord}(i, t)$ of object i in trial t, with normalized Euclidean distances:

$$\text{RSMpos}[i, t_1, t_2] = 1 - \frac{D(\text{coord}(i, t_1), \text{coord}(i, t_2))}{\max t_1, t_2 D} \tag{2}$$

The feature-based RSM represents semantic similarity through visual attributes. We construct separate matrices for color and shape attributes, then combine them:

$$\text{RSMcolor}[i, t_1, t_2] = \mathbb{1}(\text{color}(i, t_1) = \text{color}(i, t_2)) \tag{3}$$

$$\text{RSMshape}[i, t_1, t_2] = \mathbb{1}(\text{shape}(i, t_1) = \text{shape}(i, t_2)) \tag{4}$$

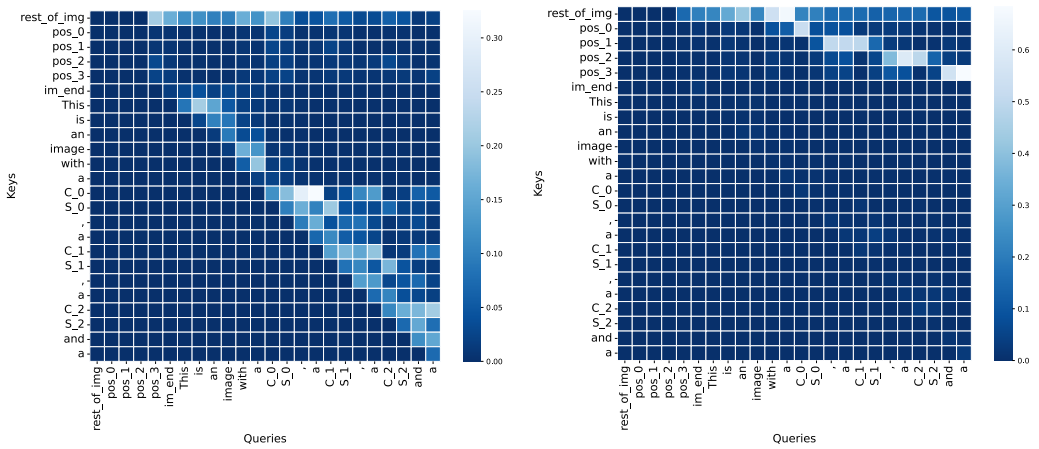
$$\text{RSMfeat}[i, t_1, t_2] = \frac{1}{2}(\text{RSMcolor}[i, t_1, t_2] + \text{RSM}_{\text{shape}}[i, t_1, t_2]) \tag{5}$$

Quantifying Alignment We quantify the alignment between model representations and object features by computing Pearson correlations between model RSMs and each target RSM. These correlations produce scalar similarity scores that vary depending on the source of the activations — for residual stream analysis, we obtain one score per layer, while for attention block analysis, we obtain one score per head per layer. By comparing these alignment patterns across our two token sources and between the residual stream and attention mechanisms, we can trace how different types of object information are encoded and transformed throughout the model's processing hierarchy.

F Attention Analyses

In order to further validate the role of the two sets of identified heads, we analyzed their attention profiles (Figure 8). We plotted the average attention profile of the top-5 heads (according to their CMA score) for each group of heads revealed by the CMA. We used a version of the scene description task with images composed of 4 objects with distinct colors, arranged in a 2x2 grid. Each object spanned 4 image patches.

For the position ID heads (Figure 8(a)), we found that, when generating the description of a target object, they attended primarily to the tokens describing the features of the previous object in the



(a) Attention Profile of Position ID Heads (b) Attention Profile of Feature Retrieval Heads

Figure 8: **Attention Analyses.** Attention profile analyses (attention scores) for (a) position ID and (b) feature retrieval heads. The attention profile is averaged over heads with the top-5 CMA scores. "pos_n" refers to the image patches at the position of object n. "C_n" and "S_n" refer to the tokens describing the color and shape, respectively, of object n in the prompt.

prompt. For instance, when generating the description of object 1 (the "a" token just before "C_1" and "S_1"), these heads attended primarily to the description of object 0 in the prompt (the tokens for "C_0", "S_0", and the following comma). This suggests that the position ID for the previous objects in the prompt are represented at the positions where those objects are described (e.g., at "C_0" and "S_0"). The position ID heads need this information to compute the position ID for the next object (to avoid mentioning the same object twice).

We found that the feature retrieval heads (Figure 8(b)) primarily attended to the image patches corresponding to the target object that was being described. For instance, when generating a description of object 1 (the "a" token before "C_1" and "S_1"), these heads attended primarily to the image patches for object 1 ("pos_1"). This is consistent with our hypothesis that these heads are responsible for retrieving the features of the target object.

G Intervention Protocol on Visual Patches

We used images with 4 objects positioned in a 2x2 grid. Each object spanned four image patches, and had a unique shape and color, making it possible to identify individual objects based on either of their visual features.

Our goal was to intervene on the feature retrieval heads by replacing the position ID for object A with the position ID for object B. An effective intervention should result in the retrieval of the features for object B instead of object A. Since we hypothesize that the feature retrieval heads use position IDs to attend to specific parts of the image, we intervened on the keys of the image patches corresponding to objects A and B. Specifically, we extracted the position ID difference between two positions A and B (denoted as $\mathrm{D}_{A->B}$) by averaging the difference between the object keys at those positions. Let x^l_A and x^l_B be the mean of key projections from layer l spanning the object patches at positions A and B in an image x. We extracted the average difference between two positions A and B as follows:

$$\mathrm{D}^l_{A->B} = \mathbf{E}_x[x^l_B - x^l_A] \tag{6}$$

In our experiments, our goal was to swap two objects that were initially located at positions A and B, by intervening on their keys such that:

$$K^l_{A'} = K^l_A + \mathrm{D}_{A->B} \tag{7}$$

$$K^l_{B'} = K^l_B - \mathrm{D}_{A->B} \tag{8}$$

where $K^l_A \in \mathbb{R}^{4\times d}$ is the key projection of the 4 image tokens in layer l spanning the object at position A.

Our approach to swapping objects by adding or subtracting the position ID difference $D_{A->B}$ is similar to the linear decomposition of binding functions proposed in [Feng and Steinhardt, 2023]. Similar to that work, we hypothesize that the function for binding position IDs to features can be represented as a simple addition of vectors.

Results Figure 5 shows the results of the intervention analysis targeting the feature retrieval heads. The results show the effectiveness of the intervention (the extent to which swapping the object keys induces a corresponding swap in the color retrieved by the model) for each pair of objects, as a function of the layers at which the intervention is performed. We found that this intervention is maximally effective when it is applied in the layers where we identified feature retrieval heads (layers 21-28), with intervention efficacy scores ranging from 93%-99% (Figure 5(a)). By contrast, this intervention was far less effective when applied to all layers (Figure 5(b), or to the layers preceding the appearance of the feature retrieval heads (Figure 5(c)).

H Intervention on Realistic Images.

(a) PUG Image (b) Pug Image flipped

Figure 9: Example of PUG Image. A valid incomplete caption for the scene description would be '*In this image there are two animals. A grey animal and a'* and the model is expected to predict red.

In order to validate the existence of the position-ID related mechanisms that we identified in our paper we performed additional experiments using the PUG (Photorealistic Unreal Graphics) environment, which allows for the generation of images that capture important properties of real-world images, including three-dimensional structure, occlusion, and lighting/shadows. We generated a dataset of images that each contained two distinctly colored 3D animals in realistic backgrounds. We randomly jittered the positions of the animals, but ensured that each image has a clear leftmost and rightmost animal (to ensure that we have ground-truth labels for the position IDs that the models will assign). An examplar image is given in Figure 9

Our hypothesis predicts that position IDs will be assigned based on left vs. right position, independent of object semantics. As such a horizontal flip of the image should swap the position ID assignments because the spatial order is reversed.

To confirm that hypothesis we designed an intervention experiment where the model predicts the color of a missing object in a caption (same as the CMA setting in Section 2.2 and Appendix **??**) based on its leftmost/rightmost label. Our position ID hypothesis predicts that by intervening on the position ID layers, we can steer the model to predict the color of either of the animals based solely on their leftmost/rightmost labels (using the position ID difference described in Appendix 2.3 but this

time we intervene on the last token instead of the image patches keys). We tested the following two conditions:

- Left -> Right: The prompt is "There are two colored animals in this image. A [color of right] animal and a." We add the difference ( ) to steer the model's answer towards repeating the color of the rightmost animal.
- Right -> Left: The prompt is "There are two colored animals in this image. A [color of left] animal and a."We add the difference ( ) to steer the model's answer towards repeating the color of the leftmost animal.

The intervention was highly effective (>90%)for both conditions and all models as shown by the intervention accuracy in table 1.

Direction / Model	Qwen2-vl-7b	Llava-Onevision-7b	Llava-1.5-7b	Llava-1.5-13b	Qwen2.5-vl-3b	Qwen2.5-vl-7b
Left → Right	94.3%	100%	100%	95.8%	100%	100%
Right → Left	97.2%	95.7%	93.0%	100%	100%	95.8%

Table 1: Intervention Performance comparison on PUG across models and directions.

I Additional CMA Results

The analysis of binding errors in Section 3 revealed the presence of an additional processing stage prior to the position ID and feature retrieval heads. This stage occurs at the positions of the tokens describing the objects in the incomplete caption, and involves the accumulation of information about the spatial position of those objects, with this information peaking around layer 15. Our hypothesis is that this stage involves matching the semantic information (color and shape) about the object described in the caption to tokens in the image, resulting in the retrieval of the position ID corresponding to that object. This information is needed to keep track of which objects have already been described.

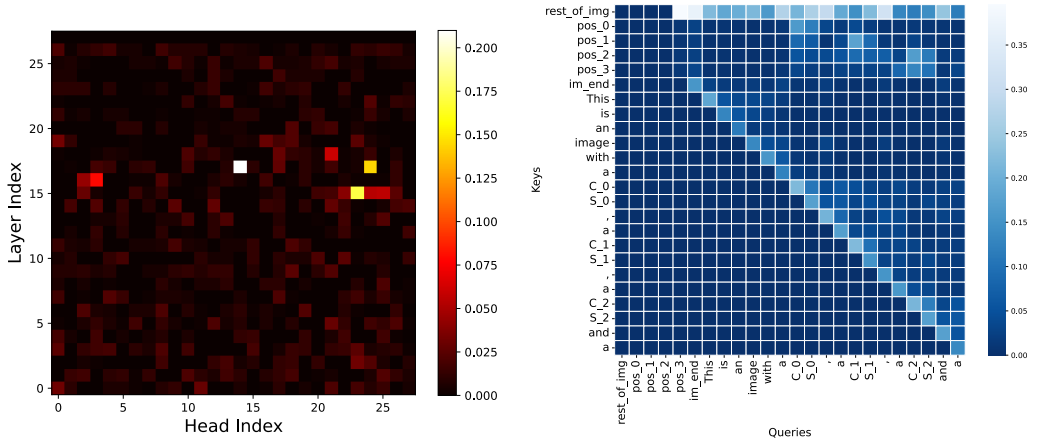
We designed an additional CMA to target this hypothesized process. We used the same condition as illustrated in Figure 3(a), where we aimed to isolate the attention heads responsible for computing the position ID of the target object, but patching was performed at the position of the token describing the color of the first object in the incomplete caption (e.g., the word 'blue' in the prompt in Figure 3(a)). The prompt for the alternative context was the same as the prompt for the clean context: "This is an image with a [color A] [shape A] and a". We hypothesized that switching the position of the objects in the image would induce a corresponding switch in the position IDs assigned to those objects in the caption tokens, and that this CMA would therefore target attention heads responsible for computing the position IDs of the objects described in the caption. We refer to these attention heads as *semantic matching heads*, because they match semantic information in the prompt to patches in the image to retrieve the corresponding position ID.

The CMA scores for this condition (Figure 10(a)) revealed a set of heads primarily located in layers 15-17. Notably, this result is consistent with the high position RSA scores for these layers as shown in Figure 6(a), and provides further evidence for the existence of a preliminary processing stage where the position IDs for the objects in the incomplete caption are computed. We also report attention pattern analyses for this group of heads in Figure 10(b). The aggregate attention profile for these heads shows attention directed from the tokens describing an object in the caption to the image tokens for that objects, consistent with the hypothesized semantic matching process.

We also looked at the attention profiles for individual semantic matching heads. Although some heads (Figures 12(a) and 12(b)) resembled the aggregate pattern (attending primarily to the object that was being described), other heads (Figures 12(c)-12(f)) displayed a more complex pattern, and this pattern appeared to be related to an intrinsic ordering for the position IDs. We discuss this issue in more detail in the following section.

J Position IDs are intrinsically ordered

Our experiments revealed evidence that the position IDs possessed an intrinsic ordering. Specifically, we found that the model had a preference to process the position IDs in raster order. For instance,



(a) CMA scores for Semantic Matching Heads (b) Attention profile of Semantic Matching Heads

Figure 10: **Semantic Matching Heads** (a) Results for Causal Mediation Analysis (CMA) targeting semantic matching heads. (b) Attention profile analyses (attention scores) for semantic matching heads. The attention profile is averaged over the top-10 CMA scores.

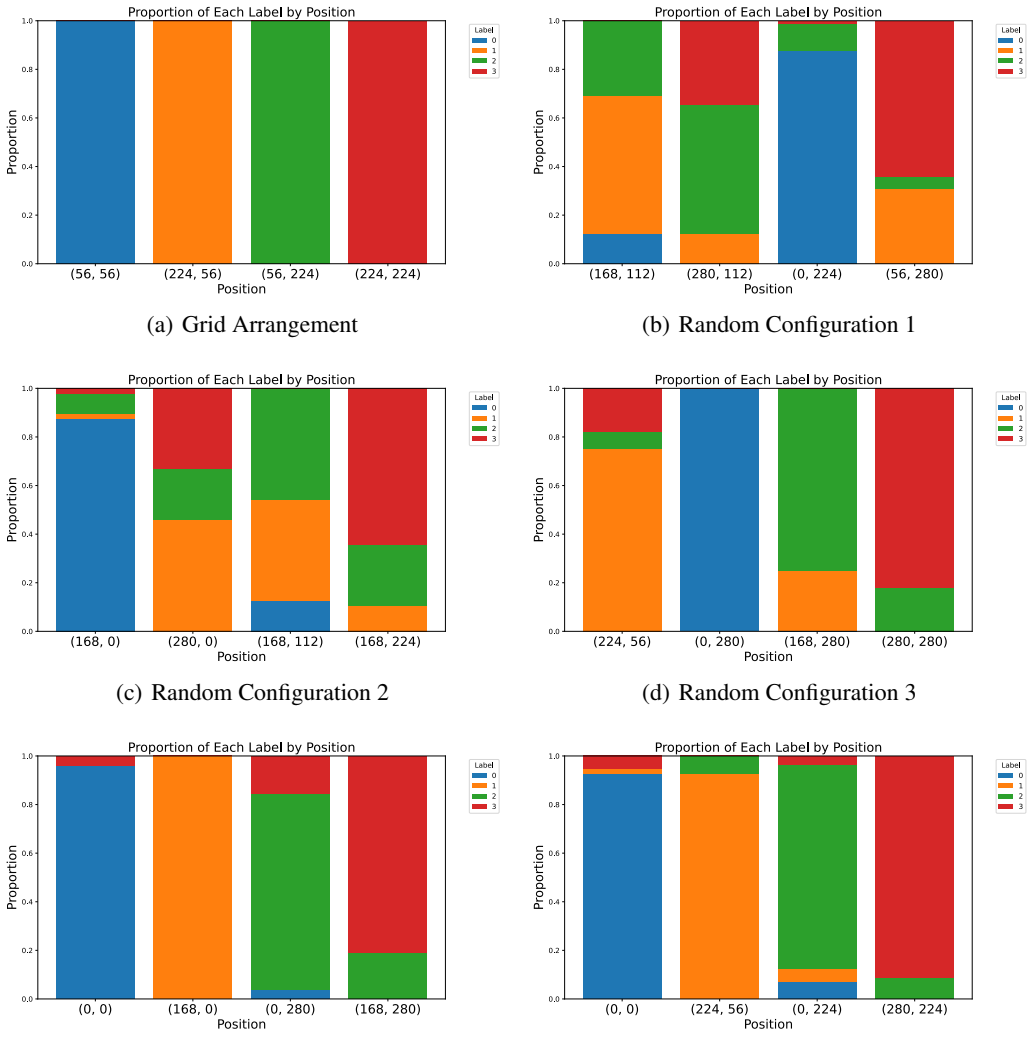
when given an image containing 4 objects arranged in a regular 2x2 grid, and the following prompt, "The four objects present in this image are 1.A ", we found that the model described the objects in raster order (upper left, upper right, lower left, lower right) for 100% of test images (sampling different feature values for the four objects).

To more systematically test whether position IDs were intrinsically ordered, we performed a series of experiments with different spatial configurations. In these experiments, each image contained four objects with the same shape, but distinct colors. For each image, we also created images with all possible permutations of the same set of objects (permuting the set of objects across the same set of four spatial locations). In addition to a regular 2x2 grid arrangement, we also created images with random spatial configurations.

The results of these experiments are shown in Figure 11. Each panel shows the results for a different spatial configuration. The spatial positions are listed in raster order along the X axis. The Y axis shows the proportion that each of these locations is described first (blue), second (orange), third (green), or fourth (red) by the model. For the regular 2x2 grid (Figure 11(a)), the objects were described in raster order 100% of the time. The results for the other spatial configurations were noisier, but a tendency toward describing the objects in raster order can still be observed.

We also found that the attention pattern for some semantic matching heads seemed to be influenced by this intrinsic ordering. Figure 12(c) shows the attention profile for Head 17.14 (i.e., head 14 in layer 17). Rather than attending to the object currently being described in the caption, this head primarily attends to the *next* object described in the caption (e.g., this head attends from the tokens describing object 0 in the prompt to image tokens for object 1). Head 19.22 displayed a similar behavior (Figure 12(e)), anticipating the next object described in the caption. Notably, the behavior of these heads differed when the caption did not list the objects in raster order. Head 17.14 attended to the object that came after (in terms of raster order) the object currently being described in the caption (Figure 12(d)). For Head 19.22, after briefly attending to the object currently being described in the caption, attention was directed back to object 0. This suggests that Head 19.22 was anticipating that the caption would be continued by listing the first object in raster order not already described in the prompt (object 0). These results suggest that the semantic matching heads coordinate this intrinsic ordering through a heterogeneous set of mechanisms at the level of individual heads. Understanding how these mechanisms work together to control the order in which objects are described is an interesting topic for future work.

Intrinsic Order Intervention To further test whether position IDs were intrinsically ordered, we performed an intervention experiment. This experiment was similar to the intervention experiment described in Section 2.3 that targeted the feature retrieval heads by intervening on the keys of the



(a) Grid Arrangement

(b) Random Configuration 1

(c) Random Configuration 2

(d) Random Configuration 3

(e) Random Configuration 4

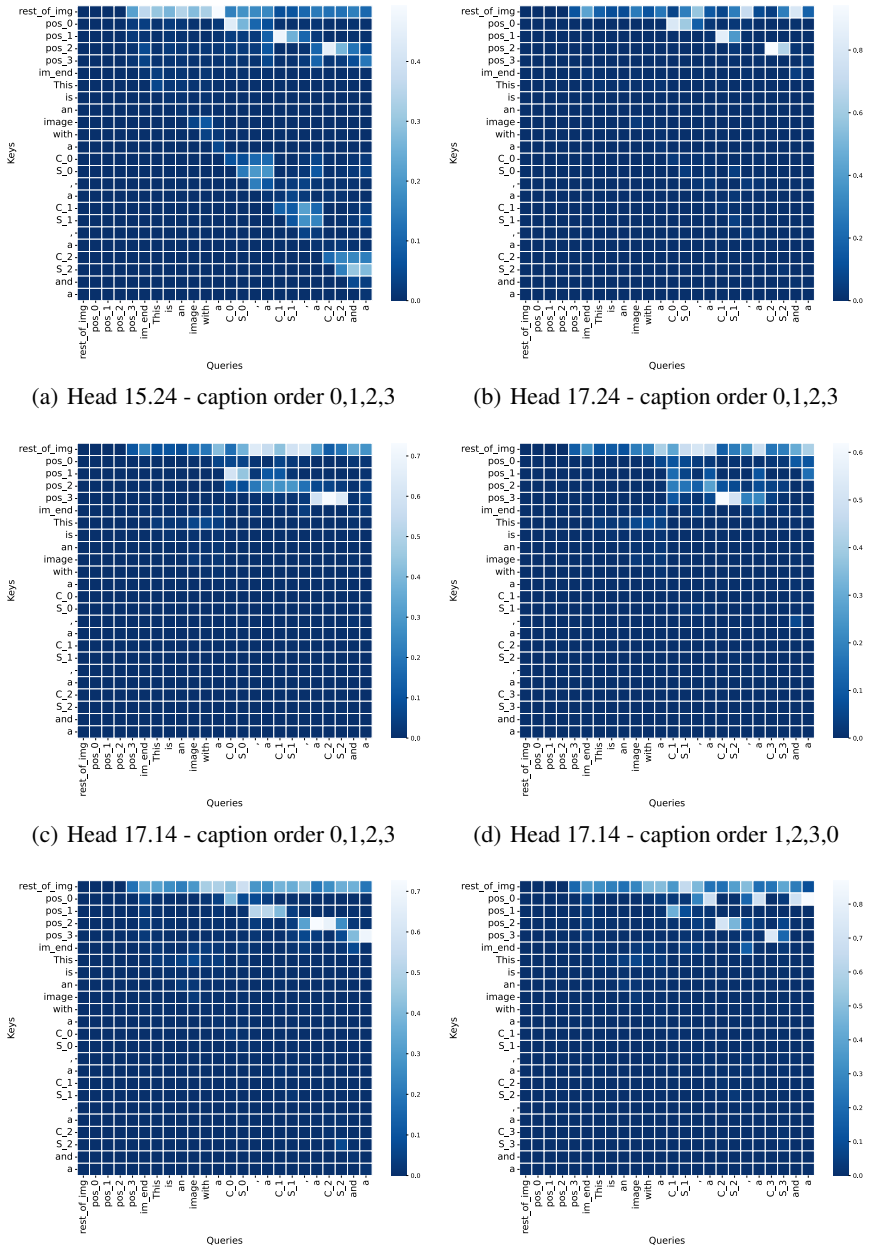
(f) Random Configuration 5

Figure 11: Scene description order for different spatial configurations.

image patches. To test whether the position IDs were intrinsically ordered, we performed this intervention together with the following open-ended prompt: "The four objects present in this image are 1.A ", thus allowing the model to describe the objects according to its own intrinsic ordering. We swapped the position ID of each object with the positions corresponding to the first generated object (e.g., at coordinates (56,56)) and measured the accuracy of the swap by identifying the change in the first generated objects. We found that this intervention was 100% effective, further confirming that the position IDs were intrinsically ordered.

K Binding Errors Interventions

Following our RSA findings which indicated compromised positional representations in low-entropy settings, we performed an intervention analysis to causally test whether substituting these representations with more robust versions from high-entropy conditions could recover task performance. Specifically, during the low-entropy scene description task, we replaced the position of the last token in the residual stream at layer 18 with the average representation derived from the high-entropy trials that share the same position ID target. To evaluate the efficacy of this intervention, we compared accuracy on the completion task between low-entropy trials with and without the intervention, across



(a) Head 15.24 - caption order 0,1,2,3

(b) Head 17.24 - caption order 0,1,2,3

(c) Head 17.14 - caption order 0,1,2,3

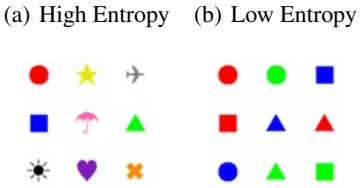
(d) Head 17.14 - caption order 1,2,3,0

(e) Head 19.22 - caption order 0,1,2,3

(f) Head 19.22 - caption order 1,2,3,0

Figure 12: **Attention analyses for individual Semantic Matching Heads.** Attention profile analyses (attention scores) for selected individual semantic matching heads, given different ordering of objects in caption. Heads are identified by their layer and head indices (e.g., Head 15.24 refers to layer 15, head 24). The indices used to describe the order of the objects in the caption refer to raster order, i.e. index 0 is the object in the upper left, and index 3 is the object in the lower right.

3x3 and 4x4 grid configurations. This intervention recovered approximately half of the performance gap observed between the original low-entropy and high-entropy trials in both the 3x3 and 4x4 conditions (Table 1). The success of this intervention provides causal evidence that degraded position IDs are a key factor contributing to impaired multi-object processing in low-entropy visual scenes.



	High Entropy	Low Entropy	Intervention
3x3	0.99 ± 0.01	0.89 ± 0.01	0.94 ± 0.01
4x4	0.79 ± 0.02	0.59 ± 0.02	0.70 ± 0.02

Table 2: Scene description task performance in the 3x3 and 4x4 grid arrangement. We report accuracy for retrieval of missing object in both the high and low entropy settings. The last column corresponds to intervention on residual stream embedding for layer 18 at last token position.

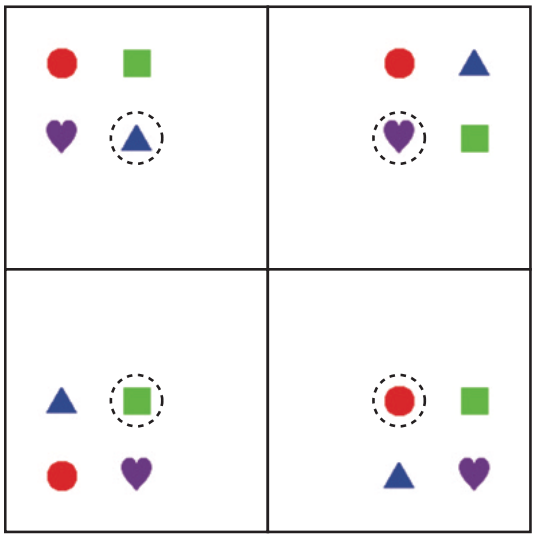


Figure 13: Stimulus Condition for the Relative vs Absolute Position IDs characterization

L Relative vs. Absolute Position Analysis

To determine whether the positional representations identified in intermediate layers use relative or absolute position encoding, we designed a controlled experiment using a 3×3 object grid. We created four different 2×2 sub-arrangements of objects, each positioned within different quadrants of the larger grid (Figure 13). Crucially, across all four arrangements, one object was always placed at the same absolute grid location (the center position).

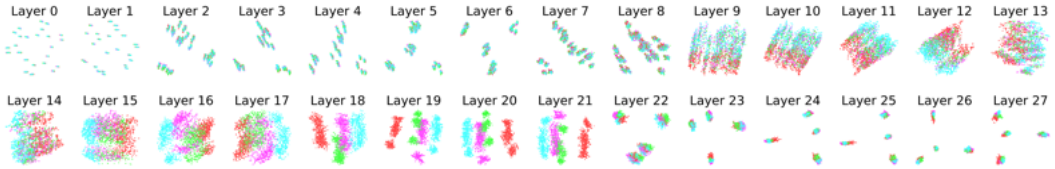
We performed PCA analyses at two levels of granularity. First, we analyzed all objects across conditions, plotting them according to either their relative grid position (Figure 14(a), eg. *top left*), their absolute grid position (Figure Figure 14(b)) or their semantic features (Figure 14(c)). Second, we focused specifically on the central object (circled in dotted lines in Figure 13), coloring these representations by either relative grid condition (Figure 14(d)) or semantic identity (Figure 14(e)).

If VLMs use absolute position encoding, the central object's position ID should cluster together regardless of grid arrangement, since it occupies the same absolute location across conditions. If VLMs use relative position encoding, the central object's position ID should vary systematically with the surrounding spatial context.

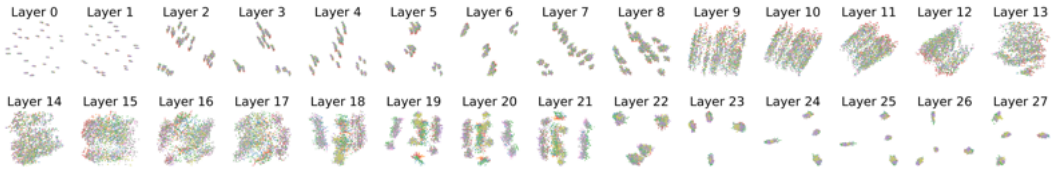
Our PCA results showed that representations of the central object clustered primarily by grid arrangement rather than remaining invariant across conditions. This provides evidence that VLMs employ a relative position encoding scheme, where spatial indices are computed based on local spatial relationships between objects rather than absolute grid coordinates.

M Additional Models

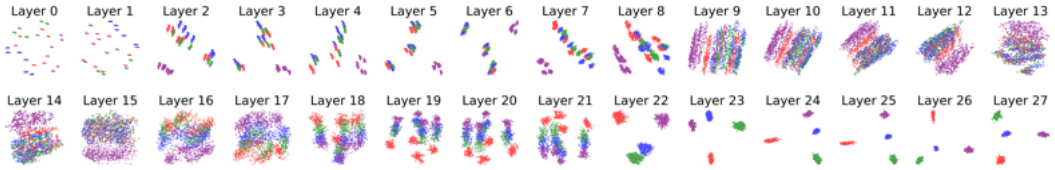
We replicated the same RSA and PCA analyses from the main paper across a range of different open source VLMs (Qwen2-vl, LLaVa 1.5, LLaVa-Onevision-7b, and Qwen 2.5) and model scales (Qwen



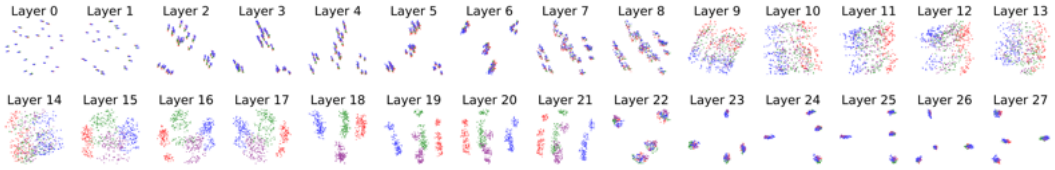
(a) PCA results coded by target object relative position for all objects.



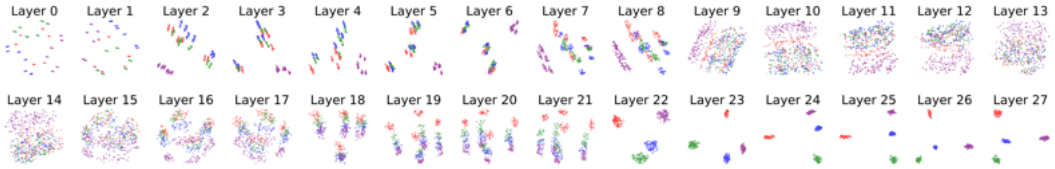
(b) PCA results colored by target object index for all objects.



(c) PCA results colored by target object features for all objects.



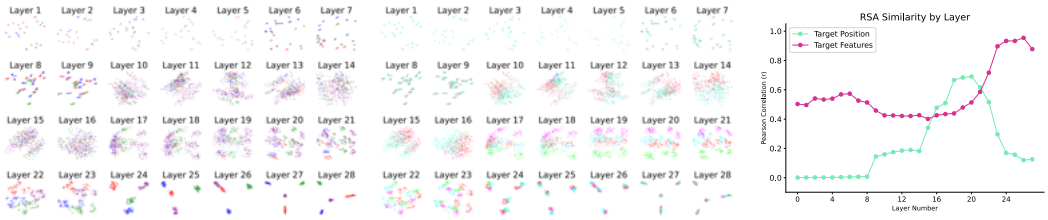
(d) PCA results colored by target object relative position for central object.



(e) PCA results colored by target object features for central object.

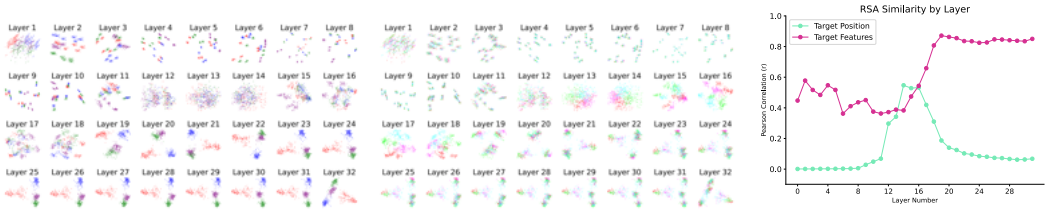
Figure 14: **Principal Component Analyses.** Stimulus condition shown in Figure 13. PCA results for all objects. (e-g) PCA results for central object only. Hidden state embeddings at last token position projected onto the top 2 principal components, coded by relative position, absolute position, or object features.

2.5 3b, 7b,). We find convergent evidence for the same two stage processing across all models and model scales.



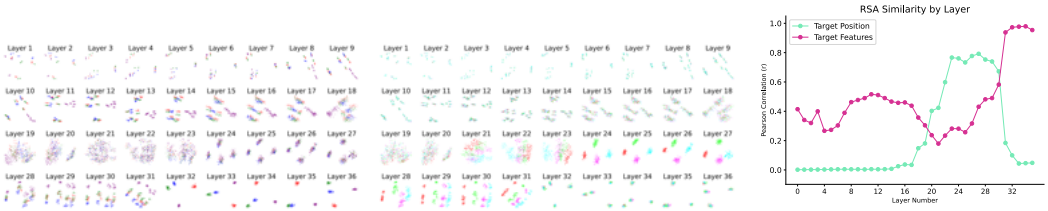
(a) Feature PCA (b) Position PCA (c) RSA

Figure 15: **Llava One Analysis.** (a) Feature PCA, (b) Position PCA, and (c) RSA analysis for LLava One.



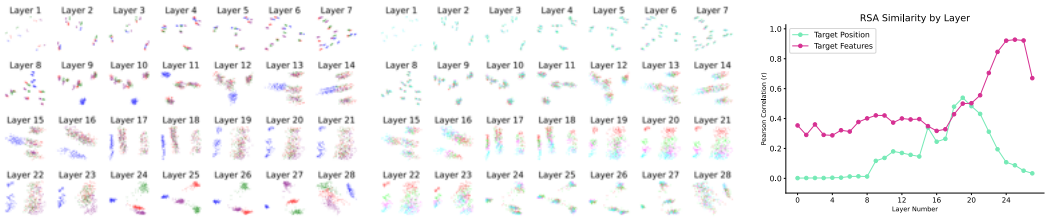
(a) Feature PCA (b) Position PCA (c) RSA

Figure 16: **Llava 1.5 Analysis.** (a) Feature PCA, (b) Position PCA, and (c) RSA analysis for LLava 1.5.



(a) Feature PCA (b) Position PCA (c) RSA

Figure 17: **Qwen 2.5 3B Analysis.** (a) Feature PCA, (b) Position PCA, and (c) RSA analysis for Qwen 2.5 3B.



(a) Feature PCA (b) Position PCA (c) RSA

Figure 18: **Qwen 2.5 7B Analysis.** (a) Feature PCA, (b) Position PCA, and (c) RSA analysis for Qwen 2.5 7B.